

SAS[®] 9.3 各国語サポート (NLS): リファレンスガイド

The correct bibliographic citation for this manual is as follows: SAS Institute Inc 2011. *SAS® SAS 9.3 National Language Support (NLS): Reference Guide*. Cary, NC: SAS Institute Inc.

SAS® SAS 9.3 各国語サポート(NLS): リファレンスガイド

Copyright © 2011, SAS Institute Inc., Cary, NC, USA

All rights reserved. Produced in the United States of America.

For a hardcopy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a Web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government Restricted Rights Notice: Use, duplication, or disclosure of this software and related documentation by the U.S. government is subject to the Agreement with SAS Institute and the restrictions set forth in FAR 52.227–19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

Electronic book 1, 2011 年 3 月

SAS® Publishing provides a complete selection of books and electronic products to help customers use SAS software to its fullest potential. For more information about our e-books, e-learning products, CDs, and hard-copy books, visit the SAS Publishing Web site at support.sas.com/publishing or call 1-800-727-3228.

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

目次

SAS 9.3 各国語サポート(NLS)の新機能	vii
推奨資料	xi

1部 各国語サポートの概念 1

1章・各国語サポート(NLS)	3
各国語サポートの概要	3
地域化と国際化の定義	4
2章・各国語サポート関連のロケール	5
各国語サポート関連のロケール概念の概要	5
ロケールの指定	6
3章・各国語サポート関連のエンコーディング	9
概要: 各国語サポート関連のエンコーディング	9
エンコーディングとトランスコーディングの違い	11
各国語サポートにおけるエンコーディングの文字セット	12
よく用いられるエンコーディング方法	12
各国語サポートエンコーディング関連の標準組織	15
EBCDIC エンコーディングの制約	15
照合順序	16
SAS セッションとデータセットのエンコーディング	20
デフォルトの SAS セッションエンコーディング	21
SAS セッションエンコーディングの設定	22
SAS セッションのエンコーディングの動作	24
4章・各国語サポート関連のトランスコーディング	27
トランスコーディングの概要	27
トランスコーディングが必要となる場合	28
トランスコーディングと変換テーブル	28
SAS データのトランスコーディングを指定する SAS オプション	30
動作環境間のトランスコーディング	30
トランスコーディングの留意点	31
互換性があるエンコーディングと互換性がないエンコーディング	32
トランスコーディングの防止	34
CVP エンジンを用いた文字データ切り捨てへの対応	34
5章・DBCS サポート	37
DBCS サポートの概要	37
DBCS でサポートされる言語	38
DBCS の指定	38
DBCS 文字セットの表示要件	38
DBCS の機能	39
汎用機版 SAS での DBCS の利用法	39
DBCS エンコーディングの SAS データ変換	40
DBCS 文字列の取り扱い方	40

2部	各国語サポート関連の自動呼び出しマクロ	41
6章	自動呼び出しマクロエントリ	43
	カテゴリ別の自動呼び出しマクロエントリ	43
	ディクショナリ	43
3部	各国語サポート関連のデータセットオプション	47
7章	データセットオプションエントリ	49
	カテゴリ別のデータセットオプション	49
	ディクショナリ	49
4部	各国語サポート関連の出力形式	53
8章	概要: 各国語サポート関連の出力形式	55
	国際化対応の日付、日時出力形式	55
	通貨表現	61
	ヨーロッパ通貨換算	68
9章	出力形式エントリ	71
	各国語サポート関連の出力形式のカテゴリ	74
	ディクショナリ	83
5部	各国語サポート関連の関数	241
10章	SAS 文字列関数の国際化の互換性	243
	SAS 文字列関数の国際化の互換性	243
11章	関数エントリ	255
	カテゴリ別の関数	256
	ディクショナリ	258
6部	各国語サポート関連の入力形式	317
12章	入力形式エントリ	319
	カテゴリ別の入力形式	321
	ディクショナリ	328
7部	各国語サポート関連のマクロ関数	441
13章	マクロ関数エントリ	443
	カテゴリ別のマクロ	443
	ディクショナリ	444

8部 各国語サポート関連のシステムオプション 455

14章・システムオプションエントリ	457
カテゴリ別のシステムオプションエントリ.....	457
ディクショナリ.....	459

9部 各国語サポート関連のコマンド、ステートメント、プロシジャオプション 479

15章・コマンド、ステートメント、プロシジャオプションのエントリ	481
カテゴリ別の各国語サポート関連のコマンド、ステートメント、およびプロシジャオプション.....	481
ディクショナリ.....	482

10部 各国語サポート関連のプロシジャ 513

16章・DBCSTAB プロシジャ	515
概要: DBCSTAB プロシジャ.....	515
構文: DBCSTAB プロシジャ.....	515
例: DBCSTAB プロシジャ.....	517
17章・TRANTAB プロシジャ	521
概要: TRANTAB プロシジャ.....	521
概念: TRANTAB プロシジャ.....	522
構文: TRANTAB プロシジャ.....	525
例: TRANTAB プロシジャ.....	531

11部 ロケール、エンコーディング、トランスコーディングの値 547

18章・LOCALE=システムオプションの値	549
ENCODING、PAPERSIZE、DFLANG、DATESTYLE オプションの LOCALE=値とデフォルト設定.....	549
19章・DBCS データ処理に関連する SAS システムオプション	561
DBCS の SAS セッションで使用されるシステムオプションの概要.....	561
SAS セッションの DBCS 値.....	561
20章・SAS 言語要素中のエンコーディング値	563
エンコーディング値を使用する SAS 言語要素の概要.....	563
データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値.....	563
21章・SAS セッションのエンコーディング値	573
UNIX エンコーディング値.....	573
Windows エンコーディング値.....	574
z/OS エンコーディング値.....	575

12部 付録 579

付録1・追加各国語サポートの言語要素	581
追加各国語サポートの言語要素.....	582
ディクショナリ.....	582
用語集	669
キーワード	673

SAS 9.3 各国語サポート(NLS)の新機能

概要

このリリースでは、各国語サポート(NLS)の対応範囲および機能が拡張されました。各国語サポートは、ソフトウェアを各国の市場向けに正しく動作させるために必要な機能の集合です。SAS には、地域の言語規則に準拠した SAS アプリケーションを確実に作成できるようにする、各国語サポート機能が組み込まれています。通常、英語で作成されるソフトウェアは、英語を使用し、米国内で使用される規則に従って出力形式が指定されたデータを使用するユーザーにとって使いやすくできています。ただし、これらのソフトウェアは、各国語サポートがなければ、世界の他の地域のユーザーにとっても同様に使いやすい製品とはなりません。SAS の各国語サポートでは、アジアやヨーロッパなどの地域のユーザーが、自国の言語および環境の下でデータを正常に処理できます。

全般的な拡張

SAS9.3 では、次の拡張が行われました。

- [LOCALE=システムオプションテーブル](#)のエイリアスが更新されました。詳細については、[LOCALE=テーブル \(549 ページ\)](#)を参照してください。
- [ユーロ通貨換算セクション](#)が更新され、ユーロの新規加盟国が追加されました。詳細については、[ヨーロッパ換算セクション \(68 ページ\)](#)を参照してください。

追加されたエンコーディング

次のエンコーディングが追加されました。

[Open Edition Katakana](#) (p. 563)

Open Edition Katakana エンコーディングを指定します。

[Open Edition Korean](#) (p. 563)

Open Edition Korean エンコーディングを指定します。

[Open Edition Simplified Chinese](#) (p. 563)

Open Edition Simplified Chinese エンコーディングを指定します。

[Open Edition Traditional Chinese](#) (p. 563)

Open Edition Traditional Chinese エンコーディングを指定します。

[Open Edition Japanese](#) (p. 563)

Open Edition Japanese エンコーディングを指定します。

[Open Edition Japanese-IBM-939E](#) (p. 563)

Open Edition Japanese-IBM-939E エンコーディングを指定します。

出力形式

次の出力形式が追加されました。

[NLDATMTZ](#) (p. 116)

ロケールの SAS 日時値の時間部分を、時刻とタイムゾーンに変換します。

[NLDATMWZ](#) (p. 118)

指定ロケールの SAS 日付値を、曜日、日時、タイムゾーンに変換します。

[NLDATMZ](#) (p. 123)

SAS 日時値を、タイムゾーンと日時などロケール依存の日時文字列に変換します。

次の出力形式が更新されました。

[YEN](#) (p. 235)

デフォルト値が 1 から 8 に変更されました。

関数

次の関数が追加されました。

[ENCODCOMPAT](#) (p. 258)

2 つのエンコーディング間のトランスコード互換性を検証します。

[ENCODISVALID](#) (p. 260)

有効なエンコーディング名を示します。

[SASMSG](#) (p. 292)

データセットのメッセージを示します。返されるメッセージは現在のロケールと指定されたキーに基づきます。

[SASMSG1](#) (p. 294)

指定されたデータセットのメッセージを示します。示されるメッセージは、指定されたロケール値と指定されたキー値に基づきます。

[SETLOCALE](#) (p. 299)

現在の SAS ロケールのロケールキーを示します。

システムオプション

次のシステムオプションが追加されました。

[URLENCODING](#) (p. 478)

URLENCODING 関数と URLDECODE 関数のパーセントエンコーディングを制御します。

[VALIDMEMNAME](#) (p. 478)

SAS データセット、ビュー、アイテムストアの命名規則を指定します。

[VALIDVARNAME](#) (p. 478)

SAS セッション中に作成され処理される有効な SAS 変数名の規則を指定します。

次のシステムオプションが拡張されました。

[DFLANG](#) (p. 464)

DFLANG システムオプションはロケールオプションをサポートします。

x SAS 各国語サポート

推奨資料

- Base SAS Procedures Guide
- SAS Companion for your operating environment
- SAS/CONNECT User's Guide
- SAS Data Set Options: Reference
- SAS Formats and Informats: Reference
- SAS Functions and CALL Routines: Reference
- SAS/GRAPH: Reference
- SAS Language Reference: Concepts
- SAS Output Delivery System: User's Guide
- SAS System Options: Reference
- SAS Statements: Reference

SAS の刊行物の総一覧については、support.sas.com/bookstore にてご確認ください。
必要な書籍についてのご質問は、下記までお寄せください。

SAS Publishing Sales
SAS Campus Drive
Cary, NC 27513-2414
電話: 1-800-727-3228
ファクシミリ: 1-919-677-8166
電子メール: sasbook@sas.com
Web アドレス: support.sas.com/bookstore

1 部

各国語サポートの概念

1 章	
各国語サポート(NLS)	3
2 章	
各国語サポート関連のロケール	5
3 章	
各国語サポート関連のエンコーディング	9
4 章	
各国語サポート関連のトランスコーディング	27
5 章	
DBCS サポート	37

1 章

各国語サポート(NLS)

各国語サポートの概要	3
地域化と国際化の定義	4

各国語サポートの概要

各国語サポート(NLS)は、ソフトウェアを各国の市場向けに正しく動作させるために必要な機能の集まりです。SAS System には、地域の言語規則に準拠した SAS アプリケーションを確実に作成できるようにする、各国語サポート機能が組み込まれています。通常、英語で作成されるソフトウェアは、英語を使用し、米国内で使用される規則に従って出力形式が指定されたデータを使用するユーザーにとって使いやすくできています。ただし、これらのソフトウェアは、各国語サポートがなければ、世界の他の地域のユーザーにとっては使いにくい製品となります。SAS の各国語サポートでは、アジアやヨーロッパなどの地域のユーザーが、自国の言語および環境の下でデータを正常に処理できます。

SAS では、データの各国語サポートに加えて、汎用機からパーソナルコンピュータに至るまで、あらゆる動作環境とハードウェアで使用されるコードの各国語サポートも提供します。このサポートは、クライアント/サーバー環境でアプリケーションを実行する、複数の言語を使用するユーザーにとって特に重要です。SAS では、以前のバージョンで開発されたアプリケーションとの整合性を維持しつつ、汎用機の各国語サポートを提供しています。

各国語サポートは、マシン間を移動するデータに適用されます。たとえば、各国語サポートにより、ターゲットマシンで使用される正しい出力形式にデータを確実に変換できます。

テキスト文字列の操作は、言語および地域の SAS 設定の影響を受けます。このアクションにより、文字の大文字化と小文字化、文字の分類、データのスキャンなどの操作による結果を修正できます。SAS には、特定の国または国のグループに固有の文字である各国語文字を適正に表示および出力するための機能が組み込まれています。

各国語サポートを組み込んだソフトウェアアプリケーションにより、ソフトウェアの機能に関して、次のような言語固有または文化固有の規則に依存することを回避できます。

- 文字の分類
- 文字の比較ルール
- コードセット

- 日時出力形式
- インターフェイス
- メッセージテキスト言語
- 数値および通貨の出力形式
- 並べ替え順序

地域化と国際化の定義

地域化は、特定の対象となる環境や市場の言語、文化、その他の要件に製品を適合させる処理で、これによりユーザーがその製品の使用時に現地の言語と規則を使用できます。ユーザーインターフェイス、システムメッセージ、ドキュメントの翻訳も地域化に含まれます。

国際化は、1つの言語またはロケールに基づくという前提なしで行われるソフトウェアアプリケーションを設計する処理です。国際化の目的の1つは、文字列の並べ替えと日付、時間、数、通貨の出力形式のルールを含む、国際的な規則を確実にサポートすることです。別の目的は、すべての異なる言語のエディションで一貫した外観、操作性、機能を提供する製品を設計することです。

アプリケーションのロジックが文化的規則(特定の地域の通貨や数値の出力形式など)をサポートするとしても、地域化されたバージョンのソフトウェアのみが現地語のユーザーインターフェイスとシステムメッセージを提供します。

SAS 各国語サポート(NLS)の機能を使用して、SAS アプリケーションを地域化や国際化することができます。

2 章

各国語サポート関連のロケール

各国語サポート関連のロケール概念の概要	5
ロケールの指定	6
SAS 起動時のロケールの指定法	6
SAS セッション中のロケールの指定法	7
言語の切り替え	7

各国語サポート関連のロケール概念の概要

ロケールには、各地域の言語、データ出力形式などの地域規則および文化が反映されています。この地域規則には、その国や地域の日付、時刻、数字の出力形式や通貨記号が含まれます。ページ順、用紙サイズ、住所、電話番号などもロケールに含まれます。

日付には、文化圏で受け入れられた地域規則に応じて、多くの表示形式があります。月は、数字または名前で表記される場合があります。名前は、完全に入力するか、略称が使用される場合があります。月、日および年の順序は、ロケールによって異なる可能性があります。

たとえば、"2002 年の 10 月の 3 日目"はロケールごとにさまざまな方法で表示されます。

ブルガリア
2002-X-3

カナダ
02-10-03

ドイツ
03-10-02

イタリア
3/10/02

米国
10/03/02

時間は、一部の英語圏の国または地域では 12 時間表記を使用して表示されるのに対し、他の英語圏では 24 時間表記を使用して時間値が表示されています。

言語はロケールの一部ですが、いずれか 1 つのロケールに固有ではありません。たとえばポルトガル語は、ポルトガルだけでなくブラジルでも話されていますが、それぞれの文化は異なります。ブラジルとポルトガルでは、データの表示形式に類似点があります。数字を表示するときに、整数部と小数部を区切るにはカンマ(,)を使用し、基数

文字の左側にある桁を区切るにはドット(.)を使用します。ただし、2つの異なるロケールで使用される通貨記号など、大きな違いもあります。ポルトガルではユーロを使用するため、ユーロ記号€が必要ですが、ブラジルではレアルが使用され、2文字の通貨記号 R\$ で表記されます。

また、1つの国で複数の公用語が使用されている場合もあります。カナダでは、英語とフランス語の2つが公用語として使用されています。LOCALE=システムオプションでは、English_Canada および French_Canada という2つの値を指定できます。

通貨などの数値にもさまざまな表示形式があります。たとえば、小数点区切り文字、つまり基数文字については、一部の地域ではドット(.)を使用し、他の地域ではカンマ(,)を使用しており、桁区切り文字については、ドットやカンマが使用され、スペースが使用されることもあります。貨幣の表記規則もロケールごとに異なります。たとえば、金額にドル記号や円記号を付けることもあります。

用紙サイズおよび測定単位もロケールの考慮事項に含まれます。標準の用紙サイズには、レター(8-1/2×11 インチ用紙)および A4 (210×297 ミリメートル用紙)があります。レター用紙サイズは、主に英語圏の国々で使用されています。A4 はその他のほとんどのロケールで使用されています。ほとんどのロケールではセンチメートルを使用していますが、一部のロケールではインチを使用します。

ロケールの指定

SAS 起動時のローケルの指定法

LOCALE=システムオプションを使用すると、SAS 起動時に SAS セッションのロケールを指定できます。LOCALE=によって、次の SAS システムオプションも暗黙的に設定されます。

- DATESTYLE=
- DFLANG=
- ENCODING=
- PAPERSIZE=
- TRANTAB=

Windows の例:

```
sas9 -locale English_UnitedStates
```

注: ロケールは、POSIX 命名規則を使用しても指定できます。たとえば、POSIX の en_US は、SAS 値の English_UnitedStates に相当します。

LOCALE=オプションのデフォルト値は、各動作環境で同じです。詳細については、“[ENCODING、PAPERSIZE、DFLANG、DATESTYLE オプションの LOCALE=値とデフォルト設定](#)” (549 ページ)を参照してください。

LOCALE=に English_UnitedStates の値を設定すると、SAS 起動時に、次のオプションが指定されたデフォルト値に暗黙的に設定されます。

- DATESTYLE=MDY
- DFLANG=English
- ENCODING=wlatin1
- PAPERSIZE=Letter

- TRANTAB=(lat1lat1, lat1lat1,wlt1_ucs,wlt1_lcs,wlt1_ccl,,)

起動時に、明示的に設定されたシステムオプションは暗黙的に設定されたオプションより優先されます。

Windows の例:

```
sas9 -papersize=A4;
```

起動時に、明示的な設定された PAPERSIZE=A4 は、LOCALE=オプションで暗黙的に設定された PAPERSIZE=オプションより優先されます。詳細については、“[PAPERSIZE=システムオプション](#)” (474 ページ)を参照してください。

SAS セッション中のロケールの指定法

LOCALE=システムオプションを使用すると、SAS セッション中に SAS セッションのロケールを指定できます。ただし、変更された LOCALE=の値を反映して暗黙的に変更されるのは、次のシステムオプションの値のみです。

- DATESTYLE=
- DFLANG=
- PAPERSIZE=

次のシステムオプションの値が、変更された LOCALE=の値を反映して暗黙的に変更されることはありません。

- ENCODING=
- TRANTAB=

注: ENCODING=は、SAS セッション中は再設定できません。設定できるのは、起動時のみです。

注: LOCALE=オプションと ENCODING=オプションの違いの詳細については、“[SAS セッションエンコーディングの設定](#)” (22 ページ)を参照してください。

Windows の例:

```
options locale=Italian_Italy;
```

LOCALE=オプションに Italian_Italy という値を割り当てると、次のオプションが、変更された LOCALE=システムオプションの値を反映して、SAS セッション中に暗黙的に再設定されます。

- DATESTYLE=DMY
- DFLANG=Italian
- PAPERSIZE=A4

ENCODING=オプションと TRANTAB=オプションの値は、再設定されません。以前の値がそのまま保持されます。

これらのシステムオプションの詳細については、“[DATESTYLE=システムオプション](#)” (460 ページ)を参照してください。

言語の切り替え

SAS メッセージは、起動中に SAS 構成ファイルの設定で指定された言語で表示されます。Unicode サーバーでは、言語の切り替え機能を使用して別の言語で SAS メッセージを表示できます。言語の切り替え機能には、LOCALELANGCHG システムオプションを使用してアクセスできます。LOCALELANGCHG が有効な場合、LOCALE システムオプションの値によって、プロシジャの出力、ユーザーインターフェイス要素およ

8 2章 ・ 各国語サポート関連のロケール

び ODS フォントの言語が決まります。LOCALELANGCHG が無効な場合、メッセージは起動中に設定された言語で表示されます。この機能は、Unicode サーバーでサポートされます。詳細については、“[LOCALELANGCHG システムオプション](#)” (471 ページ)を参照してください。

3 章

各国語サポート関連のエンコーディング

概要: 各国語サポート関連のエンコーディング	9
エンコーディングとトランスコーディングの違い	11
各国語サポートにおけるエンコーディングの文字セット	12
よく用いられるエンコーディング方法	12
各国語サポートエンコーディング関連の標準組織	15
EBCDIC エンコーディングの制約	15
照合順序	16
照合順序の概要	16
要求の代替照合順序	17
変換テーブルの指定	18
エンコーディング値の指定	18
言語照合の指定	19
SAS セッションとデータセットのエンコーディング	20
SAS セッションのエンコーディング	20
SAS データセットのエンコーディング	20
デフォルトの SAS セッションエンコーディング	21
SAS セッションエンコーディングの設定	22
SAS セッションのエンコーディングの動作	24
SAS リリース別のエンコーディングサポート	24
z/OS: 以前の SAS リリースとの互換性の確保	24
出力処理	25
入力処理	25
外部ファイルの読み込みと書き込み	25

概要: 各国語サポート関連のエンコーディング

エンコーディングは、文字セットの各文字を一意的な数値表現にマッピングし、すべてのコードポイントからなるテーブルを構成します。コードページと呼ばれるこのテーブルは、順序に従って並べられた文字セットで、各文字に数値のインデックス(コードポイント値)が関連付けられます。コードページ上の文字の位置によって、2 桁の 16 進数が決まります。

たとえば、Windows Latin1 エンコーディングのコードページは次のようになります。次の例では、行によって1桁目が決まり、列によって2桁目が決まります。大文字 A の数値表現は 16 進数の 41、等記号(=)の数値表現は 16 進数の 3D です。

図 3.1 Windows Latin1 コードページ

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	Null 0000	Stx 0001	Sof 0002	EtX 0003	Eot 0004	Enq 0005	Ack 0006	Bel 0007	Bs 0008	Ht 0009	Lf 000A	Vt 000B	Ff 000C	Cr 000D	So 000E	Si 000F
10	Plc 0010	Dcl 0011	Pc2 0012	Pc3 0013	Pc4 0014	NaK 0015	Syn 0016	EtB 0017	Can 0018	Em 0019	Sub 001A	Esc 001B	Fx 001C	Gz 001D	Rs 001E	Us 001F
20	Sp 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	;	<	=	>	?
40	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\ 005C] 005D	^ 005E	_ 005F
60	` 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	DEL 007F
80	e 20AC		/	f	//	...	†	‡	ˆ	˜	Š	<	œ		ž	
90		\	/	ˆ	˜	*	-	-	ˆ	“	š	>	œ		ž	ÿ
00	NSP 00A0	i 00A1	ç 00A2	é 00A3	ı 00A4	ı 00A5	ı 00A6	ı 00A7	ı 00A8	@ 00A9	« 00AA	» 00AB	» 00AC	» 00AD	» 00AE	» 00AF
10	° 00B0	± 00B1	² 00B2	³ 00B3	´ 00B4	µ 00B5	¶ 00B6	· 00B7	¸ 00B8	¹ 00B9	º 00BA	» 00BB	¼ 00BC	½ 00BD	¾ 00BE	¿ 00BF
20	À 00C0	Á 00C1	Â 00C2	Ã 00C3	Ä 00C4	Å 00C5	Æ 00C6	Ç 00C7	È 00C8	É 00C9	Ê 00CA	Ë 00CB	Ì 00CC	Í 00CD	Î 00CE	Ï 00CF
30	Ð 00D0	Ñ 00D1	Ò 00D2	Ó 00D3	Ô 00D4	Õ 00D5	Ö 00D6	× 00D7	Ø 00D8	Ù 00D9	Ú 00DA	Û 00DB	Ü 00DC	Ý 00DD	Þ 00DE	ß 00DF
40	à 00E0	á 00E1	â 00E2	ã 00E3	ä 00E4	å 00E5	æ 00E6	ç 00E7	è 00E8	é 00E9	ê 00EA	ë 00EB	ì 00EC	í 00ED	î 00EE	ï 00EF
50	ð 00F0	ñ 00F1	ò 00F2	ó 00F3	ô 00F4	õ 00F5	ö 00F6	÷ 00F7	ø 00F8	ù 00F9	ú 00FA	û 00FB	ü 00FC	ý 00FD	þ 00FE	ÿ 00FF

文字セットは、文字と記号のセットで、言語または言語のグループで使用されます。文字セットには、各国語文字(特定の国または国のグループに固有の文字)、特殊文字(句読点など)、アクセント記号のないラテン文字 A から Z、数字 0 から 9、コンピュータに必要なコントロール文字が含まれます。

エンコーディング方法は、数値表現を文字のセットに割り当てる一連のルールです。これらのルールによって、エンコーディングのサイズ(文字の数値表現の保存に使用されるビット数)と文字が表示されるコードページの範囲が決まります。エンコーディング方法は、コンピュータ業界で開発された標準に従って定義されています。多くの場合、エンコーディング方法は、コンピュータハードウェアベンダに固有です。

エンコーディングは、エンコーディング方法を文字セットに適用した結果です。

個々の文字には、使用するコードページによって、コードページの異なる位置が割り当てられます。たとえば、ドイツ語の大文字 Ä は、次のようになります。

- Windows Latin1 コードページ(1252)では、16 進数の C4 として表されます。
- ドイツ語 EBCDIC コードページ(1141)では、16 進数の 4A として表されます。

次のコードページの例では、文字セットがドイツ語で、エンコーディング方法は EBCDIC です。

次の例では、列によって 1 桁目が決まり、行によって 2 桁目が決まります。

図 3.2 ドイツ語 EBCDIC コードページ

HEX DIGITS 1ST → 2ND ↓	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-
-0	(SP) SP010000	& SMO30000	- SP100000	ø LC610000	∅ LC620000	□ SM190000	μ SM170000	€ SC040000	ä LA170000	ü LU170000	Ö LO180000	0 ND100000
-1	(RSP) SP300000	é LE110000	/ SP120000	É LE120000	á LA010000	¡ LJ010000	ß LS610000	£ SC020000	À LA020000	Ĵ LJ020000	÷ SA060000	1 ND010000
-2	â LA150000	ê LE150000	Â LA160000	Ê LE160000	á LB010000	ķ LK010000	š LS010000	¥ SC060000	Ā LB020000	Ķ LK020000	Š LS020000	2 ND020000
-3	{ SM110000	ë LE170000	[SM050000	Ë LE180000	č LC010000	ł LL010000	ł LT010000	• SD630000	Ĉ LC020000	Ľ LJ020000	Ť LT020000	3 ND030000
-4	â LA130000	è LE130000	Ā LA140000	È LE140000	d LD010000	m LM010000	u LU010000	© SM020000	D LD020000	M LM020000	U LU020000	4 ND040000
-5	á LA110000	í LI100000	Á LA120000	Í LI120000	e LE010000	n LN010000	v LV010000	@ SM050000	E LE020000	N LN020000	V LV020000	5 ND050000
-6	ã LA190000	î LI150000	Ā LA200000	Î LI160000	f LF010000	o LO010000	w LW010000	¶ SM050000	F LF020000	O LO020000	W LW020000	6 ND060000
-7	â LA270000	ï LI170000	Ā LA280000	Ï LI180000	g LG010000	p LP010000	x LX010000	¼ NF040000	G LG020000	P LP020000	X LX020000	7 ND070000
-8	ç LC410000	ì LI130000	Ç LC420000	Ĭ LI140000	h LH010000	q LQ010000	y LY010000	½ NF010000	H LH020000	Q LQ020000	Y LY020000	8 ND080000
-9	ñ LN190000	˜ SD190000	Ñ LN200000	˘ SD130000	i LI010000	r LR010000	z LZ010000	¾ NF050000	I LI020000	R LR020000	Z LZ020000	9 ND090000
-A	Ä LA180000	Û LU180000	ö LO170000	: SP130000	« SP170000	• SM210000	ı SP030000	¬ SM060000	š SP200000	ND011000	ND021000	ND031000
-B	. SP110000	š SC030000	, SP080000	# SMO10000	» SP180000	² SM200000	ł SP160000	SM130000	ô LO150000	û LU150000	Ô LO160000	Û LU160000
-C	< SA030000	* SMO40000	% SMO20000	§ SMO40000	ð LC630000	æ LA510000	Ð LD620000	- SM150000	SM650000	} SM140000	\ SMO70000] SMO50000
-D	(SP060000) SP070000	_ SP090000	´ SP050000	ý LY110000	˙ SD410000	Ý LY120000	˝ SD170000	ò LO130000	ù LU130000	Ò LO140000	Û LU140000
-E	+ SA010000	; SP140000	> SA060000	= SA040000	þ LT630000	Æ LA620000	Þ LT640000	´ SD110000	ó LO110000	ú LU110000	Ó LO120000	Û LU120000
-F	ı SP020000	ˆ SD150000	? SP150000	" SP040000	± SA020000	□ SC010000	⊗ SM530000	x SA070000	õ LO190000	ÿ LY170000	Ö LO200000	Ⓢ

各 SAS セッションは、デフォルトのエンコーディングに設定されます。これは、さまざまな SAS 言語要素を使用して指定できます。

エンコーディングとトランスコーディングの違い

エンコーディングは、SAS セッションのデフォルトの作業環境を確立します。たとえば、Windows Latin1 エンコーディングは、ドイツでのドイツ語の de_DE ロケールなど、西ヨーロッパロケールの Windows 環境での、SAS セッションのデフォルトエンコーディング

です。例に示すように、大文字 Ä の Windows Latin1 コードポイントは 16 進数の C4 です。

注: デフォルトのエンコーディングは、動作環境とロケールにより異なります。

ただし、国際的な環境で作業している場合(ドイツ語 EBCDIC でエンコードされた SAS データにアクセスする場合など)、大文字 Ä のドイツ語 EBCDIC コードポイントは 16 進数の 4A になります。通常は Windows Latin1 を使用するバージョンの SAS がドイツ語 EBCDIC でエンコードされたデータセットを適切に解釈するために、データをトランスコードする必要があります。トランスコーディングとは、1 つのエンコーディングから別のエンコーディングにデータを変換する処理のことです。SAS により Windows Latin1 の大文字の Ä がドイツ語 EBCDIC の大文字の Ä にトランスコードされると、文字の 16 進表現は C4 から 4A に変換されます。概念的な情報については、“[各国語サポート関連のトランスコーディング](#)” (27 ページ)を参照してください。

各国語サポートにおけるエンコーディングの文字セット

エンコーディングは、文字セットの要件に対処するために提供されています(英語の A から Z と同じ 26 文字を使用する言語はほとんどありません)。すべての言語は、次のクラスの文字セットのいずれかを使用して表記されます。

SBCS (Single-Byte Character Set, 1 バイト文字セット)

各文字を 1 バイトで表します。1 バイト文字セットは、7 ビット(最大 128 文字を提供)または 8 ビット(最大 256 文字を提供)に設定できます。一例として、8 ビット SBCS には ISO 8859-5 (キリル語)文字セット(ロシア語文字を表現)があります。

SAS での SBCS エンコーディングの使用法の詳細については、“[SAS 言語要素中のエンコーディング値](#)” (563 ページ)を参照してください。

DBCS (Double-Byte Character Set, 2 バイト文字セット)

東アジア文字セット(日本語、韓国語、簡体字中国語、繁体字中国語)を示します。ほとんどの文字は複数バイトで構成されるため、さまざま幅を持つエンコーディングが必要です。用語として DBCS (Double-Byte Character Set)が MBCS (Multi-Byte Character Set, 複数バイト文字セット)よりも一般的に使用されていますが、MBCS の方が適切です。複数バイトが必要なのは、東アジア文字セットの一部の(すべてではない)文字です。

SAS での DBCS エンコーディングの使用法の詳細については、“[DBCS データ処理に関連する SAS システムオプション](#)” (561 ページ)を参照してください。

MBCS (Multi-Byte Character Set, 複数バイト文字セット)

DBCS と同じ意味で使用されます。

よく用いられるエンコーディング方法

エンコーディング方法は、さまざまなコンピュータハードウェアメーカーおよび標準組織によって開発された標準に基づいています。詳細については、“[各国語サポートエンコーディング関連の標準組織](#)” (15 ページ)を参照してください。次に、一般的なエンコーディング方法の一覧を示します。

ASCII (American Standard Code for Information Interchange, ASCII コード、情報交換用米国標準コード)

128 文字の組み合わせを提供する米国の 7 ビットエンコーディングです。このエンコーディングには、大文字と小文字の英語、アメリカ英語の句読点、基数が 10 の

数値、少数のコントロール文字が含まれています。この 128 文字のセットは、他のほとんどのエンコーディングで共通です。ASCII はパーソナルコンピュータで使用されます。

EBCDIC (Extended Binary Coded Decimal Interchange Code)ファミリ

256 文字の組み合わせを提供する 8 ビットエンコーディングです。複数の EBCDIC ベースのエンコーディングが存在します。EBCDIC は、IBM メインフレームとほとんどの IBM ミッドレンジコンピュータで使用されています。EBCDIC では、ISO 646 規則に従って、EBCDIC エンコーディングと 7 ビット(および 8 ビット)ASCII ベースのエンコーディングの間の変換を行います。95 EBCDIC グラフィカル文字には、ほとんどの EBCDIC 1 バイトコードページにおいて同じコード位置を占める 82 のインバリエント文字(空白を含む)と、ほとんどの EBCDIC 1 バイトコードページにおいてさまざまなコード位置を占める 13 のバリエントグラフィカル文字が含まれます。バリエント文字の詳細については、“[EBCDIC エンコーディングの制約](#)” (15 ページ)を参照してください。

ISO (国際標準化機構) 646 ファミリ

128 文字の組み合わせを提供する国際標準の 7 ビットエンコーディングです。ISO 646 エンコーディングファミリは、各国バリエントとして 12 コードポイントを使用する点を除いて ASCII に似ています。12 の各国バリエントは、特定の言語で必要とされる特殊文字を表します。

ISO 8859 ファミリと Windows ファミリ

すべての ASCII コードポイントをサポートし、12 コードポイントを追加して 256 文字の組み合わせを提供する ASCII の 8 ビット拡張です。公式には ISO-8859-1 という名前を持つ Latin1 は、ISO 8859 エンコーディングファミリの中で最も頻繁に使用されるメンバです。Latin1 には、ASCII 文字に加えて、アクセント記号付き文字、西ヨーロッパの言語で必要とされる他の文字、およびいくつかの特殊文字が含まれています。HTTP および HTML プロトコルは、Unicode に基づいています。

Unicode

最大で 107,361 文字の組み合わせを提供します。基本的に Unicode は、世界中のすべての言語に対応できます。

Unicode には、次の 3 つのエンコーディング形式があります。

UTF-8

ラテン系言語、ギリシャ語、キリル語、アラビア語、ヘブライ語、および日本語、中国語、韓国語などの東アジア言語を含む MBCS エンコーディングです。UTF-8 の文字には、1 バイトから 4 バイトまでのさまざまな幅があります。UTF-8 では、1 から 128 のコード位置に ASCII 文字を残すことによって、ASCII との互換性を維持しています。

UTF-16

現代のすべての記述体系に含まれる共通文字のほとんどを包含する 16 ビット形式です。文字のほとんどは 2 バイトで均一に表記されます。ただし、4 バイトを必要とする追加文字には、代用符号スペースと呼ばれる拡張スペースがあります。

UTF-32

各文字が 4 バイトを占める 32 ビット形式です。

その他のエンコーディング

ISO 8859 ファミリには、他の言語用に設計されたメンバが他にもあります。次の表に、ISO によって承認された他のエンコーディングの説明を示します。

表 3.1 ISO によって承認されたその他のエンコーディング

ISO 標準	エンコーディング名	説明
ISO 8859-1	ラテン諸語 1	米国および西ヨーロッパの言語
ISO 8859-2	ラテン諸語 2	中央および東ヨーロッパの言語
ISO 8859-3	ラテン諸語 3	南ヨーロッパの言語、マルタ語、エスペラント語
ISO 8859-4	バルト諸語	北ヨーロッパの言語
ISO 8859-5	キリル語	スラブ系言語
ISO 8859-6	アラビア語	アラビア語
ISO 8859-7	ギリシャ語	現代ギリシャ語
ISO 8859-8	ヘブライ語	ヘブライ語およびイディッシュ語
ISO 8859-9	トルコ語	トルコ語
ISO 8859-10	ラテン諸語 6	北欧(イヌイット語、サーミ語、アイスランド語)
ISO 8859-11	ラテン/タイ語	タイ語
ISO 8859-13	ラテン諸語 7	バルト語
ISO 8859-14	ラテン諸語 8	ケルト語
ISO 8859-15	ラテン諸語 9	西ヨーロッパの言語とアルバニア語

さらに、東アジア言語用に多数のエンコーディング標準が開発されています。その一部を次の表に示します。

表 3.2 ISO によって承認された一部の東アジア言語のエンコーディング

標準	エンコーディング名	説明
GB 2312-80	簡体字中国語	中華人民共和国
CNS 11643	繁体字中国語	台湾
Big-5	繁体字中国語	台湾
KS C 5601	韓国産業標準	韓国
JIS	日本工業規格	日本
Shift-JIS	日本工業規格マルチバイトエンコーディング	日本

さまざまな言語とロケールをサポートする EBCDIC および Windows の標準には、他のエンコーディングがあります。

各国語サポートエンコーディング関連の標準組織

SAS によってサポートされるエンコーディングは、次の標準組織によって定義されません。

国際標準化機構(ISO)

標準の策定と関連活動を推進して、国家間での物資およびサービスの自由な流れを促進し、知的、科学的、技術的情報の交換を提唱します。ISO では、エンコーディングの標準も策定します。

米国規格協会(ANSI)

米国の標準規格の調整を計り、それらの標準に承認を与えます。国際規格を作成するために、ISO (国際標準化機構)と協力して活動します。

Unicode コンソーシアム

すべての文字に一意的な番号を付与する Unicode 標準を作成および推進します。

EBCDIC エンコーディングの制約

EBCDIC エンコーディング方法によっては、選択した文字にコードマップ内で割り当てられるコードポイント位置が異なる場合があります。たとえば、次の文字は、英語 (U.S.)とドイツ語の EBCDIC コードマップでは別のコードポイント位置に割り当てられません。

表 3.3 選択した言語における EBCDIC コードポイントの差異

EBCDIC コードポイント	英語(米国)	フィンランド語	スペイン語	ドイツ語(オーストリア)
4A	¢	§	[Ä
4F		!		!
5A	!	¤]	Ü
5B	\$	Å	\$	\$
5F	¬	^	¬	^
6A	!	ö	ñ	ö
79	'	é	'	'
7B	#	Ä	Ñ	#
7C	@	ö	@	§

EBCDIC コード ポイント	英語(米国)	フィンランド語	スペイン語	ドイツ語(オースト リア)
A1	~	ü	ˆ	ß
C0	{	ä	{	ä
D0	}	å	}	ü
E0	\	É	\	ö

プログラミング言語でよく使用される文字の例は、{や\$です。

これらの文字は、バリエーション文字と呼ばれます。たとえば、ドイツ語のメインフレームユーザーが ä (コードポイント C0) を入力すると、アメリカのコンパイラはコードポイント C0 を { と解釈します。

照合順序

照合順序の概要

照合順序は、文字を並べ替える順序です。たとえば、SORT プロシジャを実行すると、照合順序により他の文字に対する特定の文字の並べ替え順序(高い、低い、等しい)が決まります。

デフォルトの照合順序はバイナリ照合で、セッションエンコーディングのコードページにおける各文字の場所に従って文字を並べ替えます (セッションエンコーディングは SAS セッションのデフォルトエンコーディングです。デフォルトエンコーディングは、さまざまな SAS 言語要素を使用して指定できます)。並べ替え順序は、コードページ内のコードポイントの配置に直接対応します。データ処理で最も一般的に使用される 2 つの 1 バイト文字エンコーディング方法は、ASCII と EBCDIC です。OpenVMS、UNIX、Windows の動作環境では、ASCII エンコーディングが使用されます。IBM メインフレームコンピュータでは、EBCDIC エンコーディングが使用されます。

バイナリ照合はコンピュータにとって最も効率が高く、最速の種類 of 照合です。ただし、バイナリ照合のレポート内の文字の検索は、この方法がよくわからない場合には多少複雑です。たとえば、バイナリ照合のレポートは、大文字で始まる単語を小文字で始まる単語と区別して表示し、アクセント記号付きの文字で始まる単語はアクセント記号のない文字で始まる単語の後に表示されます。そのため、ASCII ベースのエンコーディングでは、大文字の z は小文字の a より前に表示されます。同じように、EBCDIC ベースのエンコーディングでは、小文字の z は大文字の A より前に表示されます。

バイナリ照合より優先される代替照合順序を要求できます。代替照合順序を要求するには、次の順序のいずれかを指定します。

- 変換テーブル名
- エンコーディング値
- 言語照合

表 3.4 (17 ページ) は、異なる照合順序を使用して単語のリストを並べ替えた結果を示します。

表 3.4 異なる照合順序の結果

バイナリ	変換テーブル	エンコーディング値	言語
Aaron	aardvark	Aaron	aardvark
Aztec	azimuth	Aztec	Aaron
Zeus	Aaron	Zeus	azimuth
aardvark	Aztec	aardvark	Aztec
azimuth	cote	azimuth	cote
cote	coté	cote	côte
coté	côte	coté	coté
côte	côte	côte	côte
côte	zebra	côte	zebra
zebra	zèbre	zebra	zèbre
zèbre	Zeus	zèbre	Zeus

最初の列は、ASCII ベースのエンコーディングで表示される文字のバイナリ照合の結果です。大文字と小文字で始まる単語は異なるグループになるため、一貫したアルファベット順ではありません。たとえば、単語 Zeus は、ASCII ベースのエンコーディング内で文字に割り当てられたコードポイントでの判断により、aardvark より前に表示されます。

2 番目の列は、小文字と大文字の並べ替えを交互にする変換テーブルを指定した結果です。変換テーブルを使用した場合、単語 aardvark は Zeus より前に表示されます。ただし、変換テーブルにより小文字 a は大文字 A よりも低い重みの値が割り当てられるため、単語 azimuth は Aaron より前に表示されます。さらに、アクセント記号は左から右の順序で比較して並べ替えられます。たとえば、coté は côte より前に表示されます。

3 番目の列は、ASCII ベースの 2 バイト latin1 エンコーディングを指定した結果です。

最後の列は、セッションロケール fr_FR (French_France) の言語照合の結果で、単語をアルファベット順にする照合アルゴリズムを使用します。このアルゴリズムでは、小文字で始まる単語が大文字で始まる単語より前に表示されます。さらに、言語照合によりフランス語のロケール仕様に基づいて、アクセントが右から左の順序で比較して並べ替えられます。

SAS では、言語照合の実装に International Components for Unicode (ICU) を採用しています。ICU とその Unicode 照合アルゴリズム (UCA) の実装は標準となっています。照合順序は、指定したロケールの ICU により提供されたデフォルトが使用されます。

要求の代替照合順序

代替照合順序を要求するには、次の SAS 言語要素を使用します。

- PROC SORT ステートメントの SORTSEQ=オプション。詳細については、“[照合順序オプション](#)” (483 ページ) を参照してください。

- SORTSEQ=システムオプション。詳細については、“SORTSEQ=システムオプション: UNIX、Windows および z/OS” (474 ページ)を参照してください。

どちらの方法も照合順序のすべてをサポートしていないことに注意してください。たとえば、PROC SORT ステートメントの SORTSEQ=オプションのみが言語照合をサポートします。ただし、PROC SORT ステートメントの SORTSEQ=オプションと SORTSEQ=システムオプションの両方が、変換テーブル照合順序をサポートします。

SAS/SHARE の BASE (V9) Engine と REMOTE Engine は、すべての代替照合順序をサポートします。V9TAPE 順序エンジンはデータの並べ替えに変換テーブルとエンコーディング値の使用をサポートしますが、V9TAPE Engine は言語照合をサポートしません。

変換テーブルの指定

変換テーブルは、1 バイトエンコーディングから別の 1 バイトエンコーディングにデータをトランスコードする SAS カタログエントリです。また、変換テーブルにより文字が並べ替えられます。ASCII、EBCDIC、DANISH などの標準照合順序のような、SAS が提供する変換テーブルを使用できます。または、ユーザー定義の変換テーブルを使用できます。

代替照合順序に変換テーブルを指定する場合、各文字のコードポイントを 0 から 255 の範囲の整数の重みにマッピングすることで、文字が並べ替えられます。次に、バイナリ照合が実行されます。

照合目的で、小文字と大文字を交互に並べ替える変換テーブルを作成できます。たとえば、ASCII ベースのエンコーディングで z が a より前に来る状態を修正する変換テーブルを作成できます。(ただし、変換テーブルの重みの割り当てにかかわらず、大文字、小文字を含めた完全なアルファベット順の並べ替えは困難です)。また、希望する順序で特定の言語の英字を並べ替える変換テーブルも作成できます。

TRANTAB プロシジャは、変換テーブルを作成、編集、表示します。たとえば、文字の重みの値を参照するために変換テーブルを表示できます。SAS で提供される変換テーブルは、SASHELP.HOST カタログに格納されます。作成またはカスタマイズした変換テーブルは、SASUSER.PROFILE カタログに格納されます。変換テーブルには TRANTAB のエントリタイプがあります。変換テーブルの詳細については、17 章、“TRANTAB プロシジャ,” (521 ページ)を参照してください。

PROC SORT ステートメントの SORTSEQ=オプションまたは SORTSEQ=システムオプションを使用して、変換テーブルを指定できます。たとえば、動作環境のデフォルトが ASCII ベースの Wlatin1 エンコーディングによる並べ替えで、大文字と小文字を交互にする変換テーブルで並べ替える場合、SAS 変換テーブル FRSOLAT1 を指定するために次のステートメントを発行します。

```
proc sort data=myfiles.test sortseq=FRSOLAT1;
  by name;
run;
```

変換テーブルで並べ替えられた SAS データセットには、指定した変換テーブル名を CONTENTS プロシジャ出力の照合順序として表示する並べ替え指標が含まれます。

エンコーディング値の指定

エンコーディングは、コンピュータが使用するコードポイントと呼ばれる 16 進数値にマッピングされた文字(通常の文字、表語文字、数字、句読点、記号、コントロール文字)のセットです。代替照合順序にエンコーディング値を指定する場合、文字は SAS セッションエンコーディングから指定したエンコーディングにトランスコードされ、バイナリ照合が実行されます。マルチバイトエンコーディングを含め、ENCODING=オプションでサポートされるすべてのエンコーディング値を指定できます。変換テーブルを指定して

データをトランスコードできますが、変換テーブルは 1 バイトエンコーディングのみに制限されることに注意してください。

PROC SORT ステートメントの SORTSEQ=オプションでエンコーディング値を指定できますが、SORTSEQ=システムオプションではエンコーディング値を指定できません。たとえば、SAS データセットを並べ替えて日本語の Windows 環境に移動させるとします。セッションエンコーディングが ASCII ベースでバイナリ照合が実行されている場合、ASCII ベースの 2 バイトエンコーディング SHIFT-JIS を指定するために次のステートメントを発行できます。

```
proc sort data=myfiles.test sortseq='shift-jis';
by name;
run;
```

SAS では、同名の変換テーブルのエンコーディング値があるかどうか確認されることに注意してください。変換テーブル名が存在する場合、その変換テーブルが使用されます。

エンコーディング値で並べ替えられた SAS データセットには、指定したエンコーディング値を CONTENTS プロシジャ出力の照合順序として表示する並べ替え指標が含まれます。

言語照合の指定

言語照合は、言語のルールに従って文字を並べ替え、直感的で文化的に受け入れられる結果を作成します。結果は、辞書、電話帳、本の索引などの紙媒体で使用される照合と似ています。言語照合は、レポートやその他のデータ表示の作成と、システム間の互換性の実現に役立ちます。

SAS では International Components for Unicode (ICU)を採用しています。ICU は、Unicode 照合アルゴリズム(UCA)と互換性のある言語照合のルーチンを提供するオープンソースライブラリです。UCA は、Unicode 文字列を比較して並べ替えができる標準です。

SORTSEQ=システムオプションは言語照合をサポートしないため、言語照合を要求するには、PROC SORT ステートメントの SORTSEQ=オプションを使用する必要があります。たとえば、次のステートメントでは、French_France のロケールに従って言語的に照合する SORT プロシジャを実行します。

```
options locale=fr_FR;

proc sort data=myfiles.test sortseq=linguistic;
by name;
run;
```

言語照合を要求すると、SAS セッションロケール用に ICU で提供されたデフォルトの言語照合アルゴリズムが使用されます。このアルゴリズムには、各地域の言語、データ出力形式などの規則、文化が反映されています。アルゴリズムを変更するには、LINGUISTIC キーワードの後にオプションをカッコで囲んで指定します。たとえば、異なるロケールを指定できます。また、小文字を大文字より前にしたり、その逆の順序にする CASE_FIRST=オプションも指定できます。その他いろいろな指定が可能です。通常は、ICU によりさまざまな言語とロケールがデフォルトで関連付けられるため、オプションを指定する必要はありません。言語オプションの詳細については、“[照合順序オプション](#)” (483 ページ) の SORTSEQ=オプション、または *Base SAS* プロシジャガイム内の PROC SORT ステートメントの SORTSEQ=オプションを参照してください。

言語的に並べ替えられた SAS データセットには、CONTENTS プロシジャ出力の照合順序 LINGUISTIC を表示する並べ替え指標が含まれます。この並べ替え指標とともに、データセットにより言語照合順序の完全な説明がファイルのディスクリプタ情報に記録されます。この情報は CONTENTS プロシジャ出力にも表示されます。

SAS セッションとデータセットのエンコーディング

SAS セッションのエンコーディング

現在の SAS セッションエンコーディングは ENCODING=システムオプションに割り当てられた値であるため、判別するには OPTIONS プロシジャまたは **OPTIONS** ウィンドウを使用します。たとえば、次の PROC OPTIONS ステートメントではセッションエンコーディング値を表示します。

```
proc options option=encoding;  
run;
```

SAS ログには次の情報が表示されます。

```
ENCODING=WLATIN1 Specifies default encoding for processing external data.
```

SAS セッションのエンコーディングは次のコマンドを使っても判別できます。

```
%PUT %SYSFUNC(getOption(ENCODING));
```

SAS データセットのエンコーディング

特定の SAS データセットのエンコーディングを判別するには、次の操作を実行します。

1. SAS Explorer を使用してデータセットを検索します。
2. データセットを右クリックします。
3. メニューからプロパティを選択します。
4. 詳細タブをクリックします。

データセットのエンコーディングが、他の情報と一緒に表示されます。

エンコーディングは次のコマンドを使っても判別できます。

```
%LET DSID=%SYSFUNC(open(sashelp.class,i));  
%PUT %SYSFUNC(ATTRC(&DSID,ENCODING));
```

SAS 9 データセットのエンコーディングを表示するには、CONTENTS プロシジャまたは SAS ウィンドウ環境の**プロパティ**ウィンドウを使用します。

次の例は、CONTENTS プロシジャから SAS ログにレポートされた出力です。エンコーディングは Western latin1 です。

アウトプット 3.1 SAS ログにレポートされたエンコーディング

```

The SAS System 10:15 Friday, June 06, 2003 1

The CONTENTS Procedure

Data Set Name WORK.GRADES Observations
1
Member Type DATA Variables
4
Engine V9 Indexes
0
Created 11:03 Friday, June 06 2003 Observation Length 32
Last Modified 11:03 Friday, June 06, 2003 Deleted Observations 0
Protection Compressed
NO
Data Set Type Sorted
NO
Label
Data Representation HP_UX_64, RS_6000_AIX_64, SOLARIS_64, HP_IA64
Encoding latin1
Western (ISO)

Engine/Host Dependent Information

Data Set Page Size 4096
Number of Data Set Pages 1
First Data Page 1
Max Obs per Page 126
Obs in First Data Page 1
Number of Data Set Repairs 0
File Name C:\TEMP\SAS Temporary
Files\TD228\grades.sas7bdat
Release Created 9.0000M0
Host Created WIN_NT

Alphabetic List of Variables and Attributes

# Variable Type Len

4 final Num 8
1 student Char 8
2 test1 Num 8
3 test2 Num 8

```

デフォルトの SAS セッションエンコーディング

ENCODING=オプションは、SAS セッションエンコーディングの指定に使用されます。SAS セッションエンコーディングによって、SAS 構文と SAS データセットの処理、および外部ファイルの読み込みと書き出しを行うための環境が設定されます。LOCALE=オプションと ENCODING=オプションのいずれも設定されていない場合は、デフォルト値が設定されます。

表 3.5 デフォルトの SAS セッションエンコーディング値

動作環境	デフォルトの ENCODING=値	説明
z/OS	OPEN_ED-1047	OpenEdition EBCDIC cp1047-Latin1
UNIX	Latin1	西ヨーロッパ諸語(ISO)
Windows	WLatin1	西ヨーロッパ諸語(Windows)

SAS セッションでサポートされる全エンコーディング値のリストについては、“[SAS セッションのエンコーディング値](#)” (573 ページ)を参照してください。

SAS セッションエンコーディングの設定

セッションエンコーディングを設定するには、ENCODING=システムオプション、DBCS オプションまたは LOCALE=システムオプションを使用します。

注: ENCODING=システムオプションの値は、動作環境によって異なります。

エンコーディングは、次の優先順位に従って設定されます。

1. ENCODING=システムオプション

SAS セッションエンコーディングは、DBCS オプションまたは LOCALE=オプションが指定されたかどうかに関係なく、ENCODING=オプションによって決まります。ENCODING=オプションが指定された場合、ユーザーが DBCS オプションを指定したかどうかに関係なく、有効な DBCS オプション一式が設定されます。また、ENCODING=オプションが指定された場合、LOCALE=オプションは、ユーザーによって値が指定されていない場合は、適切な値に設定されます。

注: ENCODING=オプションが指定された場合、TRANTAB=オプションは暗黙的に設定されます。

2. DBCS オプション

ほとんどの南北アメリカとヨーロッパのユーザーは SAS SBCS 環境を使用しており、DBCS 環境は使用しません。

ENCODING=オプションが指定されていない場合、SAS セッションエンコーディングは、LOCALE=オプションが指定されたかどうかに関係なく、DBCS オプションによって決まります。LOCALE=オプションは、ユーザーによって値が指定されていない場合は、適切な値に設定されます。

日本語、韓国語、簡体字中国語、繁体字中国語などの DBCS 言語の場合、エンコーディングは DBCSLANG および DBCSTYPE オプションの値によって決まります。

DBCS オプションは、DBCS 拡張ディレクトリがパスオプションリストに含まれる場合にのみ有効です。DBCS 拡張 DLL (ダイナミックリンクライブラリ)のパスは、DBCS SAS セッションの起動時に、DBCS 言語用パスオプションのパス名リストの先頭に置く必要があります。デフォルトでは、DBCS 拡張 DLL はディレクトリ: `SASROOT/dbcs/sasexe` にあります。

また、SAS セッションエンコーディングが DBCS 言語(ポーランド語、ドイツ語、フランス語など)でなくても、地域化されたリソースを使用するには、resourcesloc、msg および sashelp オプションの指定が必要になる場合があります。地域化されたリソ

ースは、!SASROOT/nls/<language identifier>/<sasmsg, sashelp, sasmacro, resource>の下にあります。言語識別子の値は、cs、de、en、es、fr、hu、it、ja、ko、pl、ru、sv、zh、zt です。

!SASROOT/nls/<language identifier>などの地域化されたディレクトリにある sasv9.cfg ファイルを指定すれば、path, resourcesloc, sasmsg、および sashelp オプションの使用を検討する必要はありません。

DBCS (SAS が DBCS エンコーディングを処理することを指定)が指定された場合、DBCSLANG=および DBCSTYPE=オプションは暗黙的に設定されます。DBCSTYPE=および DBCSLANG=のデフォルト値は、ホストの DBCS 環境(たとえば日本語や韓国語や中国語)の対応する値と一致する必要があります。

3. LOCALE=システムオプション

ENCODING=と DBCS オプションのいずれも指定されていない場合、SAS セッションエンコーディングは、LOCALE=オプションとプラットフォームによって決まります。

次の例では、エンコーディングは、Spanish_Spain ロケールのデフォルトに明示的に設定されます。

```
sas9 -locale Spanish_Spain
```

Spanish_Spain ロケールのデフォルトのエンコーディングは、wlatin1 エンコーディングです。

次の例では、SAS 起動時に wlatin2 エンコーディングが明示的に設定されます。

```
sas9 -encoding wlatin2
```

注: DBCS 拡張が使用できない場合、DBCS エンコーディング、DBCS オプションまたは CJK (中国語、日本語、韓国語)ロケールを SAS に設定すると、SAS の起動が失敗します。

注: SAS セッションのエンコーディングを変更しても、SAS キーワードや SAS ログ出力は英語のまま、影響はありません。

表 3.6 (23 ページ) では、CJK ロケールの次の値は、ロケールとプラットフォームに基づきます。

表 3.6 LOCALE=オプションに基づいたデフォルトのエンコーディング値

ロケール	WIN	MVS	UNIX
zh_TW	MS-950 (ywin)	IBM-937 (yibm)	X64 の Solaris、 SPARC の Solaris、 EUC-TW (yeuc)
zh_HK			
zh_MO			その他: MS-950 (ywin)
zh_CN	EUC-CN (zeuc)	IBM-935 (zibm)	EUC-CN (zeuc)
zh_SG			
ja_JP	SHIFT-JIS (sjis)	IBM-939 (jibm) IBM-930(j930)	h64、h6i、Power の AIX、SHIFT-JIS (sjis) その他: EUC-JP (jeuc)

ロケール	WIN	MVS	UNIX
ko_KR	EUC-KR (keuc)	IBM-933 (kibm)	EUC-KR (keuc)

SAS セッションのエンコーディングの動作

SAS リリース別のエンコーディングサポート

Base SAS ファイルには、ファイルを作成した SAS のバージョンに基づいてエンコーディングをサポートする 3 つのカテゴリがあります。

- SAS 9 で作成されたデータセットには、ファイルのディスクリプタ部分を使用してエンコーディング属性が自動的に指定されます。SAS 9 では、デフォルトで、DBCS が DBCSTYPE 値を認識してエンコーディング値に変換し、その値をフィールドのディスクリプタ部分に指定します。
- SAS 7 および SAS 8 で作成されるデータセットには、ファイルで指定されるエンコーディング値がありません。SAS 7 および SAS 8 のデータセットは、動作環境の SAS セッションエンコーディングで作成されたものと見なされます。ただし、ファイルのディスクリプタ部分はエンコーディング値をサポートします。SAS 9 セッションで SAS 7 または SAS 8 のファイルを置き換えたり更新したりすると、SAS はデフォルトで、現在のセッションエンコーディングをファイルのディスクリプタ部分に指定します。SAS 8 の場合、DBCS には、エンコーディングフィールドではなく DBCSTYPE フィールドがあります。
- SAS 6 で作成されたデータセットにはファイルに関連付けられたエンコーディング値がなく、ファイルでエンコーディング値を指定できません。

z/OS: 以前の SAS リリースとの互換性の確保

NLSCOMPATMODE システムオプションを設定することにより、以前のリリースの SAS との互換性を確保します。

注: NLSCOMPATMODE は、z/OS 動作環境でのみサポートされています。

以前のリリースの SAS で実行されたプログラムは、NLSCOMPATMODE を指定しても動作し続けます。

NONLSCOMPATMODE システムオプションにより、外部データの読み込みと書き込みや、SAS 構文とユーザーデータの処理など、ENCODING=オプションまたは LOCALE=オプションで設定されたエンコーディングでデータが処理されるように指定します。

以前のリリースの SAS で実行できた既存のプログラムの一部は、NONLSCOMPATMODE が有効な場合には実行できなくなります。SAS 構文ステートメントで文字代替を行った場合は、各国語の文字を使用するようにプログラムを変更する必要があります。たとえば、既存の SAS 構文で文字 Å を \$文字で代替したフィンランド語の顧客は、フィンランド語環境で \$を使用するようにプログラムを更新する必要があります。

詳細については、“NLSCOMPATMODE システムオプション: z/OS” (473 ページ)を参照してください。

出力処理

SAS 9 でデータセットを作成する場合、エンコーディングは次のように決定されます。

- 新しい出力ファイルを作成する場合、データは、現在のセッションエンコーディングを使用してそのファイルに書き込まれます。
- 現在のセッションとは異なるデータ表記を指定する OUTREP=オプションを使用して新しい出力ファイルを作成する場合、データは、OUTREP=値によって指定されるオペレーティングシステムでのデフォルトセッションエンコーディングを使用してそのファイルに書き込まれます。詳細については、“[OUTREP=データセットオプション](#)” (52 ページ)を参照してください。
- 新しい出力ファイルで既存のファイルを置き換える場合、新しいファイルは既存のファイルのエンコーディングを継承します。別の動作環境にある既存のファイルを置き換える出力処理の場合、または既存のファイルにエンコーディングが指定されていない場合は、現在のセッションエンコーディングが使用されます。

入力処理

SAS 9 での入力(読み込み)処理におけるエンコーディング動作を次に示します。

- ほとんどのユーザーは、入力ファイルにエンコーディングは指定しない、デフォルトの動作を選びます。
- セッションエンコーディングとファイルで指定されたエンコーディングの間に互換性がない場合、データはセッションエンコーディングにトランスコードされます。たとえば、現在のセッションエンコーディングが ASCII で、ファイルで指定されたエンコーディングが EBCDIC の場合、SAS は EBCDIC から ASCII にデータをトランスコードします。
- ファイルでエンコーディングが指定されていない場合、SAS は、ファイルのデータ表記が現在のセッションと異なる場合にのみ、データをトランスコードします。

外部ファイルの読み込みと書き込み

SAS は、現在のセッションエンコーディングを使用して外部ファイルの読み込みと書き込みを行います。SAS では、外部ファイルのエンコーディングがセッションエンコーディングと同じであると想定します。たとえば、外部ファイルを読み込んで新しい SAS データセットを作成する場合、SAS は、外部ファイルのエンコーディングと現在のセッションのエンコーディングが同じであると想定します。エンコーディングが同じでない場合、外部データは、新しい SAS データセットに誤って書き込まれる可能性があります。入出力処理を実行する SAS ステートメントの構文の詳細については、“[SAS データのトランスコーディングを指定する SAS オプション](#)” (30 ページ)を参照してください。

4 章

各国語サポート関連のトランスコーディング

トランスコーディングの概要	27
トランスコーディングが必要となる場合	28
トランスコーディングと変換テーブル	28
SAS データのトランスコーディングを指定する SAS オプション	30
動作環境間のトランスコーディング	30
トランスコーディングの留意点	31
互換性があるエンコーディングと互換性がないエンコーディング	32
互換性があるエンコーディングと互換性がないエンコーディングの概要	32
改行文字と EBCDIC と ASCII 間のデータ転送	32
互換性のある EBCDIC と OpenEdition エンコーディング	33
互換性のある東アジア言語の MBCS エンコーディング	33
トランスコーディングの防止	34
CVP エンジンを用いた文字データ切り捨てへの対応	34

トランスコーディングの概要

トランスコーディングとは、1つのエンコーディングから別のエンコーディングにデータを変換する処理のことです。トランスコーディングは、SAS セッションエンコーディングとデータのエンコーディングが異なる場合に必要になります。多くの場合、異なるロケールとエンコーディングを使用する動作環境間でデータを移動するときにトランスコーディングが必要になります。

たとえば、Latin1 エンコーディングを使用する UNIX 動作環境で作成されたファイルを、ドイツ語 EBCDIC エンコーディングを使用する IBM メインフレームに移動する場合などが考えられます。IBM メインフレームでファイルが処理される時、データは、Latin1 エンコーディングからドイツ語 EBCDIC エンコーディングに再マッピングされます。データに大文字の Å が含まれる場合、16 進数は C4 から 4A に変換されます。

トランスコーディングでは、言語間の翻訳は行いません。トランスコーディングで行われるのは文字の再マッピングです。

異なるエンコーディングを使用する動作環境間でデータを動的にトランスコードするには、明確なエンコーディング値を指定する必要があります。詳細については、“SAS 言語要素中のエンコーディング値” (563 ページ)を参照してください。

トランスコーディングが必要となる場合

データがトランスコードされる頻度が高いのは、次のような状況です。

- 異なるロケールまたは異なる動作環境で実行している、2つの異なる SAS セッション間でデータを共有する場合
- 大文字または小文字への変換などの、テキスト文字列の操作を実行する場合
- 別の言語の文字を表示または出力する場合
- 異なるロケールで実行している SAS セッション間でデータをコピーして貼り付ける場合

トランスコーディングと変換テーブル

LOCALE=または ENCODING=を間接的に指定すると、TRANTAB=オプションに適切な translation-table の値が設定されます。変換テーブルは、SBCS エンコーディングを別の SBCS エンコーディングにトランスコードしたり元に戻したりするために使用します。たとえば、Windows Latin2 を ISO Latin2 にマッピングする特定の変換テーブルがあります。

次の図に、変換テーブルを示します。Windows Latin 2 (wlt2)から ISO Latin 2 (lat2)へのマッピングの変換テーブル領域名は"table 1"で、ISO Latin 2 から Windows Latin 2 へのマッピングの領域名は"table 2"です。

図 4.1 SAS Windows Latin 2 から ISO Latin 2 への変換テーブル

MLT2LAT2 table 1:	
	0 1 2 3 4 5 6 7 8 9 A B C D E F
00	'000102030405060708090A0B0C0D0E0F'x
10	'101112131415161718191A1B1C1D1E1F'x
20	'202122232425262728292A2B2C2D2E2F'x
30	'303132333435363738393A3B3C3D3E3F'x
40	'404142434445464748494A4B4C4D4E4F'x
50	'505152535455565758595A5B5C5D5E5F'x
60	'606162636465666768696A6B6C6D6E6F'x
70	'707172737475767778797A7B7C7D7E7F'x
80	'80818283848586878889A98AA6ABAEAC'x
90	'908B8C8D8E8F91929893B994B6BBEBC'x
A0	'A0B7A2A3A4A195A7A896AA9799AD9AAF'x
B0	'B09BB2B3B49C9D9EB8B1BA9FA5BDB5BF'x
C0	'C0C1C2C3C4C5C6C7C8C9CACBCCCDCECF'x
D0	'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0	'E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEF'x
F0	'F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF'x

MLT2LAT2 table 2:	
	0 1 2 3 4 5 6 7 8 9 A B C D E F
00	'000102030405060708090A0B0C0D0E0F'x
10	'101112131415161718191A1B1C1D1E1F'x
20	'202122232425262728292A2B2C2D2E2F'x
30	'303132333435363738393A3B3C3D3E3F'x
40	'404142434445464748494A4B4C4D4E4F'x
50	'505152535455565758595A5B5C5D5E5F'x
60	'606162636465666768696A6B6C6D6E6F'x
70	'707172737475767778797A7B7C7D7E7F'x
80	'808182838485868788898B9192939495'x
90	'909697999BA6A9AB98ACAEB1B5B6B7BB'x
A0	'A0A5A2A3A4BC8CA7A88AAA8D8FAD8EAF'x
B0	'B0B9B2B3B4BE9CA1B89ABA9D9FBD9EBF'x
C0	'C0C1C2C3C4C5C6C7C8C9CACBCCCDCECF'x
D0	'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0	'E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEF'x
F0	'F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF'x

LOCALE=または ENCODING=システムオプションとその他のエンコーディングオプション(ステートメント、コマンドまたはプロシジャ)を使用すると、変換テーブルを直接作成または管理する必要がなくなります。

注意:

利用方法がよくわからない場合は変換テーブルを変更しないでください。変換テーブルは SAS のシステムにより各国語サポートを提供するために内部的に利用されます。変換テーブルの利用方法がよくわからない場合や、技術的な支援がない場合は設定を変更しないでください。

TRANTAB=オプションでは、SAS セッションで使用される変換テーブルを指定します。詳細については、“TRANTAB=システムオプション”(476 ページ)を参照してください。TRANTAB プロシジャは、カスタマイズした変換テーブルの作成、編集、表示に使用されます。詳細については、17 章、“TRANTAB プロシジャ,”(521 ページ)を参照してください。

SAS データのトランスコーディングを指定する SAS オプション

さまざまな言語要素に次の SAS オプションを使用すると、トランスコーディングを実行したりデフォルトのエンコーディング動作を無効にして別の動作を指定したりできます。これらの要素によって、SAS ファイルまたは SAS アプリケーションに異なるエンコーディングを指定したり、トランスコーディングを抑止したりできます。

表 4.1 SAS データのトランスコーディングを指定する SAS オプション

オプション	使用箇所
CHARSET=	ODS MARKUP ステートメント
CORRECTENCODING=	DATASETS プロシジャの MODIFY ステートメント
ENCODING=	%INCLUDE、FILE、FILENAME、INFILE、ODS の各ステートメント、FILE および INCLUDE コマンド
ENCODING=	DATA ステップ
INENCODING=	LIBNAME ステートメント
ODSCHARSET=	XML の LIBNAME ステートメント
ODSTRANTAB=	XML の LIBNAME ステートメント
OUTENCODING=	LIBNAME ステートメント
XMLENCODING=	XML の LIBNAME ステートメント

これらのオプションに使用するサポート対象エンコーディング値のリストについては、“データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値” (563 ページ)を参照してください。

動作環境間のトランスコーディング

動作環境を越えて SAS ファイルを移動したり、SAS ファイルにアクセスしたりすると、トランスコーディングが自動的に実行されます。一般的な SAS トランスコーディングの動作は次のとおりです。

CPORT プロシジャと CIMPORT プロシジャ

トランスポートファイルを作成するには、変換テーブルを使用して 1 つのエンコーディングから別のエンコーディングに、またその逆方向に、トランスコーディングを自動的に実行します。データは、まず入力元エンコーディングからトランスポート出力形式に変換され、次にトランスポート出力形式から出力先エンコーディングに変換されます。詳細については、*Base SAS プロシジャガイド*を参照してください。

SAS の CEDA (クロス環境データアクセス)機能

現在のセッションエンコーディングとは異なるエンコーディングを含む SAS データセットを処理すると、CEDA ソフトウェアを使用してデータが自動的にトランスコードされます (CEDA は、動作環境から別の動作環境へファイルを移動すると、SAS ファイルを正しいデータ表記に変換する処理も行います)。詳細については、*SAS 言語リファレンス: 解説編*を参照してください。

SAS/CONNECT データ転送サービス(UPLOAD および DOWNLOAD プロシジャ)

詳細については、*SAS/CONNECT User's Guide* を参照してください。

SAS/CONNECT 計算サービス(RSUBMIT ステートメント)

クライアントセッションが処理のためにサーバーセッションに送信するステートメントのブロックを識別します。詳細については、*SAS/CONNECT User's Guide* を参照してください。

SAS/CONNECT および SAS/SHARE リモートライブラリサービス(LIBNAME)

クライアントアクセスでリモートマシンのライブラリを参照します。詳細については、*SAS/CONNECT User's Guide* および *SAS/SHARE User's Guide* を参照してください。

トランスコーディングの留意点

通常、トランスコーディングは問題なく実行されますが、データに影響を与え、不満足な結果をもたらす場合があります。次に、例を示します。

- エンコーディング同士が競合する可能性があります。つまり、2 種類のエンコーディングにおいて、同じ文字に対して異なるコードポイントが使用されていたり、2 つの異なる文字に対して同じコードポイントが使用されていたりする場合があります。
- 1 つのエンコーディングの文字が、別のエンコーディングに存在しない可能性があります。たとえば、特定のエンコーディングにはドル記号(\$)の文字がない場合があります。ドル記号をサポートしていないエンコーディングにデータをトランスコーディングすると、その文字が出力されなかったり、表示されなかったりします。
- 1 つのエンコーディングでの 1 文字に対応するバイト数が、別のエンコーディングでの同じ文字に対応するバイト数と異なる可能性があります。たとえば、DBCS から SBCS にトランスコーディングする場合があります。したがって、トランスコーディングにより文字値が切り捨てられる可能性があります。
- トランスコーディング中、データを元のエンコーディングにトランスコーディングして戻すことができないなどのエラーが発生した場合は、データが失われる可能性があります。つまり、更新処理のためにデータセットを開いても、オブザベーションが更新されない場合があります。ただし、入力(読み込み)処理のデータセットを開き、出力データセットを何も開かない場合、SAS は出力可能な警告を発行します。処理が進行し、PRINT プロシジャまたは他の読み込み操作で、トランスコードしないデータを表示できます。
- CEDA には、いくつかの処理制限があります。たとえば、CEDA は更新処理をサポートしません。
- 不正なエンコーディングについては、SAS 9 セッションでコピーされたり、データの別のセッションエンコーディングと置き換えられたりする場合、SAS 7 または SAS 8 データセットにスタンプできます。不正なエンコーディングスタンプは、PROC DATASETS の MODIFY ステートメントの CORRECTENCODING=オプションによって訂正できます。文字変数にバイナリデータが含まれると、トランスコーディングによってデータが破損する可能性があります。

互換性があるエンコーディングと互換性がないエンコーディング

互換性があるエンコーディングと互換性がないエンコーディングの概要

ASCII はほとんどのエンコーディングの基礎で、多くのパーソナルコンピュータ、ミニコンピュータおよびワークステーションで使用されます。ただし、IBM メインフレームでは EBCDIC エンコーディングが使用されます。そのため、ASCII と EBCDIC のマシンとデータには互換性がありません。あるエンコーディングの一部またはすべての文字が別のエンコーディングの文字と異なる場合、トランスコーディングが必要になります。

ただし、トランスコーディングを避けるため、データセットを作成して SAS がトランスコードしないエンコーディング値を指定することもできます。たとえば、ENCODING=データセットオプション、LIBNAME ステートメントの INENCODING=オプション/OUTENCODING=オプションのいずれかで次の値を使用した場合、トランスコーディングは実行されません。

- ANY を使用して、EBCDIC エンコーディングと ASCII エンコーディング間でもトランスコーディングを実行しないように指定します。

注: ANY は BINARY の指定と同等です。データはバイナリであるため、実際のエンコーディングには意味がありません。

- ASCHIANY を使用して、すべての ASCII ベースのエンコーディングと互換性があるデータセットを作成できます。
- EBCDICANY を使用して、すべての EBCDIC ベースのエンコーディングと互換性があるデータセットを作成できます。

混在したエンコーディングを含む SAS データセットを作成できます。たとえば、Latin1 と Latin2 の両方を含めることができます。入力処理または出力処理でのデータのトランスコードは避けず、デフォルトでは、データは現在のセッションエンコーディングにトランスコードされます。

SAS ファイルと SAS セッションが互換性がないエンコーディングを使用する場合、データをトランスコードする必要があります。たとえば、ASCII と EBCDIC には互換性がありません。

SAS ファイルと SAS セッションに互換性があるエンコーディングがある場合は、トランスコーディングは必要ありません。

動作環境別のエンコーディングのリストについては、“[SAS セッションのエンコーディング値](#)” (573 ページ)を参照してください。

改行文字と EBCDIC と ASCII 間のデータ転送

ASCII 動作環境で実行するソフトウェアでは、改行文字で行末を指定する必要があります。データを z/OS から ASCII エンコーディングをサポートするマシンに転送する場合、EBCDIC 復帰改行文字が認識されないため、特に HTML 出力で出力形式の問題が発生する可能性があります。z/OS では、次の 2 セットの EBCDIC ベースのエンコーディングをサポートします。

- 名前に EBCDIC が含まれるエンコーディングは、ASCII 改行文字に対して EBCDIC 改行の従来のマッピングを使用します。そのため、データが 1 列で表示される可能性があります。

- 名前に Open Edition が含まれるエンコーディングは、改行文字を行末文字として使用します。ASCII を使用する動作環境にデータを転送する場合、EBCDIC 復帰改行文字を ASCII 改行文字にマッピングします。このマッピングにより、ASCII アプリケーションが行末を適切に解釈できるため、出力形式がより正確になります。

動作環境別のエンコーディングのリストについては、“[SAS セッションのエンコーディング値](#)” (573 ページ)を参照してください。

互換性のある EBCDIC と OpenEdition エンコーディング

EBCDIC と OpenEdition は、互換性があるエンコーディングです。

名前に EBCDIC が含まれるエンコーディングは、EBCDIC 改行(0x25)と復帰改行(0x15)文字の従来のマッピングを使用します。

名前に OPEN_ED、説明に OpenEdition が含まれるエンコーディングは、復帰改行と改行文字のマッピングを切り替えます。そのため、改行文字を行末文字として使用します。

2 つのエンコーディングが同じコードページ番号を使用し、1 つが EBCDIC でもう 1 つが Open Edition の場合、トランスコーディングは必要ありません。

例:

データが EBCDIC1143 でエンコードされ、SAS セッションが OPEN_ED-1143 でエンコードされる場合は、同じ 1143 コードページが使用されるためトランスコーディングは必要ありません。

ASCII と EBCDIC 間でデータを転送するために、互換性があるエンコーディングのリストから Open Edition エンコーディングを指定できます。

注: Open Edition エンコーディングは、デフォルトで、NONLSCOMPATMOD で使用されます。

互換性のある東アジア言語の MBCS エンコーディング

一部の東アジアの 2 バイト(DBCS)は、互換性があるエンコーディングです。次のリストの各行に、互換性があるエンコーディングを示します。

- SHIFT-JIS、MS-932、IBM-942、MACOS-1
- MS-949、MACOS-3、EUC-KR
- EUC-CN、MS-936、MACOS-25、DEC-CN
- EUC-TW、DEC-TW
- MS-950、MACOS-2、BIG5

SAS セッションがグループ内の 1 つのエンコーディングでエンコードされ、データセットが同グループ内の別のエンコーディングでエンコードされる場合、トランスコーディングは発生しません。

例:

セッションエンコーディングが SHIFT-JIS でデータセットエンコーディングが IBM-942 の場合、トランスコーディングは発生しません。

トランスコーディングの防止

エンコーディング値によっては、SAS がトランスコードしないデータセットを作成できるものがあります。また、入力または出力処理のデータをトランスコードするのではなく、エンコーディングが混在するデータを含む SAS ライブラリを作成することもできます。たとえば、Latin1 と Latin2 の両方を含めることができます。

たとえば、ENCODING=データセットオプション、LIBNAME ステートメントの NENCODING=オプション/OUTENCODING=オプションのいずれかで次の値を使用する場合は、トランスコーディングを回避できます。

- ANY を使用して、EBCDIC エンコーディングと ASCII エンコーディング間でもトランスコーディングを実行しないように指定します。
注: ANY は BINARY の指定と同等です。データはバイナリであるため、実際のエンコーディングには意味がありません。
- ASCHIANY を使用すると、ASCII ベースのエンコーディング間ではトランスコーディングが不要と指定されます。
- EBCDICANY を使用すると、EBCDIC ベースのエンコーディング間ではトランスコーディングが不要と指定されます。

詳細については、“ENCODING=データセットオプション” (49 ページ) および “INENCODING=/OUTENCODING=オプション” (499 ページ)を参照してください。

TRANSCODE=オプションを使って、特定の列のデータに対してトランスコーディングを防止しつつ、残りのデータセットの文字データをトランスコードすることができます。詳細については、“PROC SQL の TRANSCODE=列修飾子” (502 ページ)を参照してください。

CVP エンジンを用いた文字データ切り捨てへの対応

ENCODING=データセットオプションを指定する場合、出力データセットのエンコーディングでは、元のデータセットより多くのスペースが必要になります。たとえば、UTF8 エンコーディングを使用して Windows 環境で DBCS データを書き出す場合は、各 DBCS 文字に 3 バイトが必要です。データの切り捨てを避けるには、各変数で元のデータの幅より 1.5 倍広い幅が必要になります。

トランスコーディングを必要とする SAS データファイルを処理する場合は、CVP (文字変数埋め込み)エンジンに文字変数の長さの拡張を要求して、文字データの切り捨てが発生しないようにできます (変数の長さは、各変数の値を保存するために使用するバイト数です)。

文字データの切り捨ては、1 つのエンコーディングにおける 1 文字のバイト数が、別のエンコーディングにおける同じ文字のバイト数と異なる場合に発生する可能性があります。たとえば、1 バイト文字セット(SBCS)を、2 バイト文字セット(DBCS)やマルチバイト文字セット(MBCS)にトランスコードする場合などです。SBCS では各文字を 1 バイトで表すのに対し、DBCS では各文字を 2 バイトで表します。MBCS は、1 から 4 バイトのさまざまな長さで文字を表します。たとえば、Wlatin2 から UTF-8 などの Unicode エンコーディングにトランスコーディングする場合、変数の長さ(バイト単位)は値を保持するのに不十分であり、結果として文字データが切り捨てられることとなります。

CVP エンジンを使用して、トランスコーディングの前に変数の長さが拡張されるように拡張量を指定し、それからデータを処理します。CVP エンジンは、トランスコーディン

グに向けてデータの準備を行うために使用される中間エンジンと考えてください。長さが増えた後は、デフォルトのベースエンジンなどの主エンジンを使用して実際のファイル処理が行われます。

CVP エンジンは、SAS データファイル限定の読み取り専用エンジンです。文字変数の拡張要求(たとえば、LIBNAME ステートメントを使用)は、次のいずれかの方法で行えます。

- CVP エンジンを明示的に指定し、変数の長さの 1.5 倍のデフォルト拡張を使用します。
- LIBNAME の CVPBYTES=オプションまたは CVPMULTIPLIER=オプションにより、CVP エンジンを暗黙的に指定します。これらのオプションにより、拡張量を指定します。さらに、CVPENGINE=オプションを使用して、SAS ファイルの処理で使用する主エンジンを指定できます。デフォルトは、デフォルトの SAS エンジンです。

たとえば、次の LIBNAME ステートメントでは、CVP エンジンを明示的に割り当てています。文字変数の長さは、長さを 1.5 倍にするデフォルト拡張を使用して増やします。たとえば、長さが 10 の文字変数の新しい長さは 15 になり、長さが 100 の文字変数の新しい長さは 150 になります。

```
libname expand cvp ' SAS data-library' ;
```

注: 拡張量は、どのような拡張にでも対応できるよう十分大きくする必要があります。大きさが不十分な場合は切り捨てが発生します。

注: WHERE 式を使用してオブザベーションのサブセットを条件付きで選択する処理の場合は、CVP エンジンを使用するとパフォーマンスに影響することがあります。CVP エンジンを使用しない場合のファイルの処理は、CVP エンジンを使用する場合のファイルの処理より高速です。たとえば、CVP エンジンを使用する場合は、データセットにインデックスがあっても、WHERE 式を最適化するためにそのインデックスは使用されません。

詳細および例については、“LIBNAME Statement” in *SAS Statements: Reference* の CVP オプションを参照してください。

5 章

DBCS サポート

DBCS サポートの概要	37
DBCS でサポートされる言語	38
DBCS の指定	38
DBCS 文字セットの表示要件	38
DBCS の機能	39
汎用機版 SAS での DBCS の利用法	39
DBCS エンコーディングの SAS データ変換	40
DBCS 文字列の取り扱い方	40

DBCS サポートの概要

東アジア言語には数千の文字があるため、それぞれの文字を表すには 2 バイトの情報が必要です。

東アジアの各言語では、コンピュータのメーカー間での標準化が行われていないため、通常は複数の DBCS エンコーディングシステムがあります。SAS は、主要な東アジア言語を使用する各メーカーに固有の DBCS エンコーディング情報を処理します。

ソフトウェアを適切に拡張することにより、SAS を使用して次の機能を実行できます。

- 主要な東アジア言語のいずれかを SAS System の DBCS バージョンで表示します。
- 東アジア言語のコンピュータからデータをインポートし、アプリケーションまたは動作環境から別のアプリケーションまたは動作環境にデータを移動します(この場合は、SAS ACCESS や他の SAS 製品が必要になります)。
- 東アジアでの標準的な日付と時間の表記を、SAS 日付値、SAS 時間値および SAS 日時値に変換します。
- 東アジア言語の文字を含むデータセットおよびさまざまなタイプの出力(レポートやグラフなど)を作成します。

DBCS でサポートされる言語

次の言語が DBCS でサポートされます。

- 簡体字中国語(中華人民共和国とシンガポールで使用)
- 繁体字中国語(香港特別行政区、マカオ特別行政区、台湾で使用)
- 日本語
- 韓国語

DBCS の指定

DBCS を指定するには、次の SAS システムオプションを使用します。

DBCS

DBCS 文字を認識します。

DBCSSLANG=

言語を指定します。

DBCSTYPE=

DBCS エンコーディング方法の種類を指定します。

Windows の SAS 構成ファイルの例:

```
/*basic DBCS options */
```

```
-dbcs /*Recognizes DBCS*/
```

```
-dbcstype PCMS /*Specifies the PCMS encoding method*/
```

```
-dbcslang JAPANESE; /*specifies the Japanese language */
```

DBCSTYPE=と DBCSSLANG=は、バージョン 6.12 で導入されました。代替方法として、ENCODING=を設定すると DBCSTYPE=および DBCSSLANG=オプションが暗黙的に設定されます。詳細については、“[ENCODING システムオプション: UNIX、Windows、および z/OS](#)” (465 ページ)を参照してください。

DBCS 文字セットの表示要件

DBCS 文字を含むデータセットを表示するには、次のリソースが必要です。

- システムによる複数コードページのサポート
- 使用する言語に対応する DBCS フォント

ユーザー定義文字を作成して SAS ソフトウェアで使用する必要がある場合は、使用するコンピュータが DBCS をサポートしている必要があります。これらのコンピュータは、米国およびヨーロッパでは数が限られています。東アジア言語のコンピュータシステムでは、さまざまな方法を使用して文字を作成します。一般的な方法の 1 つでは、多くの場合にラテン語の文字を使用して、ユーザーが文字の音声発音を入力します。

コンピュータは、入力された音声発音に似た音を持つ文字のメニューを表示し、ユーザーに1つ選択するよう求めます。

DBCS の機能

特定の DBCS 言語と動作環境を認識する SAS セッションをセットアップした後で、次の一般領域で指定した言語を操作できます。

- DATA ステップおよびバッチ指向のプロシジャ
- ウィンドウ機能と対話型の機能
- システム間の接続性と互換性
- データベースへのアクセス
- グラフィック

DATA ステップおよびバッチ指向のプロシジャでは、引用符内の文字列として許容される場所であればどこでも DBCS を使用できます。変数値、変数ラベルおよびデータセットラベルは、すべて DBCS にできます。DBCS は、入力データとして、また FORMAT プロシジャでの範囲指定とラベル指定で使用することもできます。WHERE 式の処理では、埋め込み DBCS テキストを検索できます。

汎用機版 SAS での DBCS の利用法

汎用機システムで使用される別のタイプの DBCS エンコーディングでは、DBCS サポートと 3270 型データストリームを組み合わせています。それぞれの DBCS 文字列は、シフトアウト/シフトイン(または SO/SI)と呼ばれるエスケープコードで囲まれています。これらのコードは、旧型のプリンタを EBCDIC 文字セットから DBCS 文字セットにシフトアウトする必要があったために採用されました。主要なメーカーでは、さまざまな方法で SO/SI のエンコーディングを行います。一部のメーカーは DBCS コードに 1 バイトのシフトコード情報を埋め込むのに対し、他のメーカーは 2 バイトのシフトコード情報を DBCS コードに埋め込みます。これらの相違により、汎用機に関する DBCS 情報を読み込むときに問題が発生します。

PC、ミニコンピュータおよびワークステーションに SO/SI はありませんが、メーカーによって異なる独自の DBCS エンコーディングが使用されています。SAS では、SO/SI システムで DBCS を読み込める数種類の出力形式と入力形式を備えています。

表 5.1 SO/SI システムでの DBCS をサポートする SAS の出力形式と入力形式

キーワード	言語要素	説明
\$KANJI	入力形式	日本語の漢字 DBCS から SO/SI を削除します
\$KANJI	出力形式	日本語の漢字 DBCS に SO/SI を追加します
\$KANJI	出力形式	日本語の漢字 DBCS に SO/SI を追加します

キーワード	言語要素	説明
\$KANJI	出力形式	日本語の漢字 DBCS から SO/SI を削除します

DBCS エンコーディングの SAS データ変換

通常、1つのコンピュータシステムで生成された DBCS データは、別のコンピュータシステムで生成されたデータと互換性がありません。SAS には、次の表に示すように、1つの DBCS ソースから別の DBCS ソースへの変換を可能にする機能があります。

表 5.2 DBCS 変換

言語要素	タイプ	使用法	参照先
KCVT	関数	1つの動作環境から別の動作環境へ DBCS データを変換します	“KCVT 関数” (p. 266)
CPORT	プロシジャ	1つの環境から別の環境へファイルを移動します	Base SAS プロシジャガイド
CIMPORT	プロシジャ	CPORT によって作成されたトランスポートファイルをインポートします	Base SAS プロシジャガイド

DBCS 文字列の取り扱い方

- DBCS 文字を操作する場合は、データを見直して、DATA または PROC ステップでデータをインポート、変換または使用するときに SAS が文字列全体を認識することを確認してください。
- シフトアウト/シフトインエスケープコードを使用する汎用機システムでは、動作環境間での変換中に DBCS 文字列が切り捨てられる可能性があります。
- PRINT、REPORT、TABULATE、FREQ プロシジャを使用して処理する場合には、DBCS 文字列が分割される可能性があります。不適切な分割が発生した場合は、DBCS 文字列のどちらかの側にスペースを追加して、強制的に適切な位置で分割されるようにする必要があります。PROC REPORT および PROC PRINT で SPLIT=オプションを使用し、強制的に適切な位置で文字列を分割させることもできます。

2 部

各国語サポート関連の自動呼び出しマクロ

6 章	自動呼び出しマクロエントリ.....	43
-----	--------------------	----

6 章

自動呼び出しマクロエントリ

カテゴリ別の自動呼び出しマクロエントリ	43
ディクショナリ	43
%KLOWCASE と%QKLOWCAS 自動呼び出しマクロ	43
%KTRIM と%QKTRIM 自動呼び出しマクロ	44
%KVERIFY 自動呼び出しマクロ	44

カテゴリ別の自動呼び出しマクロエントリ

次の表に、SAS 各国語サポート自動呼び出しマクロの概要を示します。詳細については、それぞれの各国語サポート関連のマクロに関する情報を参照してください。

カテゴリ	言語要素	説明
DBCS	%KLOWCASE と%QKLOWCAS 自動呼び出しマクロ (p. 43)	大文字を小文字に変更します。
	%KTRIM と%QKTRIM 自動呼び出しマクロ (p. 44)	末尾の空白を取り除きます。
	%KVERIFY 自動呼び出しマクロ (p. 44)	式に固有の最初の文字の位置を返します。

ディクショナリ

%KLOWCASE と%QKLOWCAS 自動呼び出しマクロ

大文字を小文字に変更します。

カテゴリ: DBCS

要件 MAUTOSOURCE システムオプション

構文

`%KLOWCASE` (text | text expression)

`%QKLOWCAS` (text | text expression)

詳細

`%KLOWCASE` と `%QKLOWCAS` マクロは、大文字の英字を小文字に変更します。次のリストにある特殊文字またはニーモニック演算子が引数に含まれている可能性がある場合は、`%QKLOWCAS` を使用します。

`%KLOWCASE` は、引数に引用符が含まれる場合でも、引用符なしの結果を返します。`%QKLOWCAS` は、次の特殊文字とニーモニック演算子がマスクされた結果を生成します。そのためマクロプロセッサは、それらの結果を、マクロ言語の要素ではなくテキストとして解釈します。

`& % ' " () + - * / < > = ~ ^ ~ ; , blank AND OR NOT EQ NE LE LT GE GT IN`

自動呼び出しマクロは SAS ライブラリに含まれます。このライブラリはサイトにインストールされていないか、またはサイト固有のバージョンの可能性があります。このマクロにアクセスできない場合、またはライブラリがサイト固有のバージョンかどうかを確認する場合は、オンサイトの SAS サポート担当者にお問い合わせください。

%KTRIM と %QKTRIM 自動呼び出しマクロ

末尾の空白を取り除きます。

カテゴリ: DBCS

要件 MAUTOSOURCE システムオプション

構文

`%KTRIM` (text | text expression)

`%QKTRIM` (text | text expression)

詳細

`KTRIM` マクロと `QKTRIM` マクロは、末尾の空白を取り除きます。引数が次に示す特殊文字やニーモニック演算子を含む場合は、`%QKTRIM` を使用します。

`QKTRIMS` は、次の特殊文字とニーモニック演算子がマスクされた結果を生成します。そのためマクロプロセッサは、それらの結果を、マクロ言語の要素ではなくテキストとして解釈します。

`& % ' " () + - * / < > = ~ ? ~ ; , # blank AND OR NOT EQ NE LE LT GE GT IN`

自動呼び出しマクロは SAS ライブラリに含まれます。このライブラリはサイトにインストールされていないか、またはサイト固有のバージョンの可能性があります。このマクロにアクセスできない場合、またはライブラリがサイト固有のバージョンかどうかを確認する場合は、オンサイトの SAS サポート担当者にお問い合わせください。

%KVERIFY 自動呼び出しマクロ

式に固有の最初の文字の位置を返します。

カテゴリ: DBCS
要件 MAUTOSOURCE システムオプション

構文

%KVERIFY (source, excerpt)

構文

source

excerpt に存在しない文字を確認する、テキストまたはテキスト式です。

excerpt

%KVERIFY で source の確認に使用される文字のセットを定義する、テキストまたはテキスト式です。

詳細

%KVERIFY は、excerpt に存在しない source 内の最初の文字の位置を返します。source 内のすべての文字が excerpt に存在する場合、%KVERIFY は 0 を返します。

自動呼び出しマクロは SAS ライブラリに含まれます。このライブラリはサイトにインストールされていないか、またはサイト固有のバージョンの可能性があります。このマクロにアクセスできない場合、またはライブラリがサイト固有のバージョンかどうか確認する場合は、オンサイトの SAS サポート担当者にお問い合わせください。

3 部

各国語サポート関連のデータセット オプション

7 章	
データセットオプションエントリ	49

7 章

データセットオプションエントリ

カテゴリ別のデータセットオプション	49
ディクショナリ	49
ENCODING=データセットオプション	49
OUTREP=データセットオプション	52

カテゴリ別のデータセットオプション

各国語サポートは、選択したデータセットオプションのデータセットコントロールカテゴリに影響を与えます。次の表に、各データセットオプションの概要を示します。詳細については、各データセットオプションの辞書エントリを参照してください。

カテゴリ	言語要素	説明
データセットコントロール	ENCODING=データセットオプション (p. 49)	SAS データセットの読み取りまたは書き込みに使用するエンコーディングを手動で指定します。
	OUTREP=データセットオプション (p. 52)	出力 SAS データセットのデータ表記を指定します。

ディクショナリ

ENCODING=データセットオプション

SAS データセットの読み取りまたは書き込みに使用するエンコーディングを手動で指定します。

該当要素: DATA ステップおよび PROC ステップ

カテゴリ: データセットコントロール

構文

ENCODING= ANY | ASCIIANY | EBCDICANY | *encoding-value*

説明**ANY**

トランスコーディングを行わないように指定します。

注: ANY は BINARY の指定と同等です。データはバイナリであるため、実際のエンコーディングには意味がありません。

ASCHANY

混在したエンコーディングが ASCII エンコーディングの場合にトランスコーディングを行わないように指定します。

EBCDICANY

混在したエンコーディングが EBCDIC エンコーディングの場合にトランスコーディングを行わないように指定します。

encoding-value

エンコーディング値を指定します。

参照項目: “[各国語サポート関連のエンコーディング](#)” (9 ページ)

詳細

ENCODING=の値は、SAS データセットに現在のセッションエンコーディングとは異なるエンコーディングがあることを示します。データは、データセットから読み取られるときに、指定したエンコーディングからセッションエンコーディングにトランスコードされます。また、データセットに書き込まれるときに、セッションエンコーディングから指定したエンコーディングにトランスコードされます。

入力処理

デフォルトでは、入力処理のエンコーディングは次のように決まります。

- セッションエンコーディングとファイルに指定されたエンコーディングが異なる場合、データはセッションエンコーディングにトランスコードされます。
- ファイルにエンコーディングが指定されておらず、ファイルのデータ表記が現在のセッションのエンコーディングとは異なる場合、データは現在のセッションのエンコーディングにトランスコードされます。

出力処理

デフォルトでは、出力処理のエンコーディングは次のように決まります。

- データは現在のセッションのエンコーディングを使用してファイルに書き込まれます。ただし、OUTREP=データセットオプション、LIBNAME ステートメントの OUTENCODING=オプション、または ENCODING=データセットオプションを使用して別の出力表記が指定された場合を除きます。
- 既存のファイルが新しいファイルで置き換えられる場合、新しいファイルは既存のファイルのエンコーディングを継承します。
- 既存のファイルが、別の動作環境で作成されたか、エンコーディングが指定されていない新しいファイルで置き換えられる場合、新しいファイルは現在のセッションのエンコーディングを使用します。

注: 文字メタデータとデータ出力は、データセットの作成時に使用されたエンコーディングと異なるエンコーディングを指定すると文字化けします。この例では、出力するデータセットは内部的に ASCII としてエンコードされますが、データセットオプションでは EBCDIC エンコーディングが指定されています。SAS は、データを EBCDIC から ASCII にトランスコードしようと試みますが、データはすでに ASCII です。その結果、データが文字化けします。

```
data a;
x=1;
```

```
abc='abc';
run;
proc print data=a (encoding="ebcdic");
run;
```

注: 次の ENCODING=の値は無効です。

- UCS2
- UCS4
- UTF16
- UTF32

比較

- セッションエンコーディングは、ENCODING=システムオプションまたは LOCALE=システムオプションを使用して指定されます。それぞれの動作環境にはデフォルトのエンコーディングがあります。
- LIBNAME ステートメントの INENCODING=オプション(入力ファイル用)および OUTENCODING=オプション(出力ファイル用)を使用して SAS ライブラリ用のエンコーディングを指定できます。LIBNAME ステートメントオプションと ENCODING=データセットオプションを両方とも指定すると、データセットオプションが使用されません。

例

例 1: エンコーディングが混在し、トランスコーディングが抑制された SAS データセットの作成

データセットオプション ENCODING=ANY を指定すると、混在したエンコーディングが含まれる SAS データセットを作成し、入力処理または出力処理でトランスコーディングを抑制できます。

この例では、新しいデータセット MYFILES.MIXED に、Latin1 エンコーディングを使用するデータと Latin2 エンコーディングを使用するデータが含まれます。データセットの処理時に、トランスコーディングは行われません。たとえば、Latin1 セッションエンコーディングの正しい Latin1 文字と Latin2 セッションエンコーディングの正しい Latin2 文字が表示されます。

```
libname myfiles 'SAS data-library';
data myfiles.mixed (encoding=any);
set work.latin1;
set work.latin2;
run;
```

例 2: 特定のエンコーディングを使用した SAS データセットの作成

出力処理では、現在のセッションエンコーディングを手動で指定できます。このアクションは、通常のファイルアクセスに別のセッションエンコーディングを使用する場合などに必要になることがあります。

たとえば、現在のセッションエンコーディングが Wlatin1 の場合、ENCODING=WLATIN2 を指定してエンコーディング Wlatin2 を使用するデータセットを作成できます。次のステートメントでは、セッションエンコーディングのかわりに Wlatin2 エンコーディングを使用して新しいデータセットにデータを書き込むように指示しています。エンコーディングはまた、ファイルのディスクリプタ部分にも指定されません。

```
libname myfiles 'SAS data-library';
data myfiles.difencoding (encoding=wlatin2);
.
.
.
run;
```

例 3: 入力処理のエンコーディングより優先する

入力処理では、ファイル内に指定されたエンコーディングを無効にして別のエンコーディングを指定できます。

この例では、現在のセッションエンコーディングが EBCDIC-870 ですが、ファイルのディスクリプタ情報にはエンコーディング値 EBCDIC-1047 が含まれています。ENCODING=EBCDIC-870 と指定すると、データのトランスコーディングは行われずに、データは EBCDIC-870 エンコーディングを使用して表示されます。

```
proc print data=myfiles.mixed (encoding=ebcdic870);
run;
```

関連項目:

- “[各国語サポート関連のエンコーディング](#)” (9 ページ) の概念の説明

ステートメントとコマンドのオプション:

- “[ENCODING=オプション](#)” (495 ページ)
- “[INENCODING=/OUTENCODING=オプション](#)” (499 ページ)

システムオプション:

- “[ENCODING システムオプション: UNIX、Windows、および z/OS](#)” (465 ページ)
- “[LOCALE システムオプション](#)” (470 ページ)

OUTREP=データセットオプション

出力 SAS データセットのデータ表記を指定します。

該当要素: DATA ステップおよび PROC ステップ

カテゴリ: データセットコントロール

参照項目: “OUTREP= Data Set Option” in *SAS Data Set Options: Reference*

4 部

各国語サポート関連の出力形式

8 章		
	概要: 各国語サポート関連の出力形式	55
9 章		
	出力形式エントリ	71

8 章

概要: 各国語サポート関連の出力形式

国際化対応の日付、日時出力形式	55
通貨表現	61
通貨の概要	61
米国ドル	61
地域化されたユーロ	62
カスタマイズした通貨表現	62
地域化された各国語および国際化対応通貨表現	63
固有の各国語および国際化対応通貨表現	65
例: 各国語および国際化対応出力形式での通貨の表現	66
ヨーロッパ通貨換算	68
ヨーロッパ通貨換算の概要	68
固定レートによるユーロ換算	68
変動レートによるユーロ換算	69
例: ヨーロッパ通貨とユーロ間の換算	69
ヨーロッパ通貨間の直接換算	70

国際化対応の日付、日時出力形式

SAS では、最も一般的に使用される英語の日付出力形式に相当する、国際化対応出力形式がサポートされています。いずれの場合でも、出力形式は対応する英語の出力形式と同様に機能します。最大幅、最小幅、デフォルトの幅のみが異なります。

表 8.1 国際化対応の日付、日時出力形式

言語	英語出力形式	国際化対応出力形式	最小	最大	デフォルト
アフリカーンス語(AFR)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEW.	10	200	20

言語	英語出力形式	国際化対応出力形式	最小	最大	デフォルト
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
カタロニア語(CAT)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEW.	10	200	20
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
クロアチア語(CRO)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	40	27
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
チェコ語(CSY)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	2	40	25
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
デンマーク語(DAN)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10

言語	英語出力形式	国際化対応出力形式	最小	最大	デフォルト
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	2	31	31
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
オランダ語(NLD)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	2	38	28
	WORDDATX.	NLDATE.	10	200	20
	WEEKDAY.	NLDATEWN.	4	200	10
フィンランド語(FIN)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	2	37	37
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
フランス語(FRA)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	27	27
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20

言語	英語出力形式	国際化対応出力形式	最小	最大	デフォルト
ドイツ語(DEU)	DATE.	NLDATE.	105	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	30	30
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
ハンガリー語(HUN)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	40	28
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
イタリア語(ITA)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	28	28
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
マケドニア語(MAC)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10

言語	英語出力形式	国際化対応出力形式	最小	最大	デフォルト
	WEEKDATX.	NLDATEWX.	3	40	29
	WEEKDDATX.	EURDFWDX.	1	32	1
	WORDDATX.	NLDATEWN.	4	200	10
ノルウェー語(NOR)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	26	26
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
ポーランド語(POL)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	20	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	2	40	34
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
ポルトガル語(PTG)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	38	38
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATEWX.	10	200	20
ロシア語(RUS)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30

言語	英語出力形式	国際化対応出力形式	最小	最大	デフォルト
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	2	40	29
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
スペイン語(ESP)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	1	35	35
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
スロベニア語(SLO)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	40	29
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
スウェーデン語(SVE)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	26	26
	WEEKDAY.	NLDATEWN.	4	200	10

言語	英語出力形式	国際化対応出力形式	最小	最大	デフォルト
	WORDDATX.	NLDATE.	10	200	20
フランス語(スイス) (FRS)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	26	26
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20
ドイツ語(スイス) (DES)	DATE.	NLDATE.	10	200	20
	DATETIME.	NLDATM.	10	200	30
	DOWNAME.	NLDATEWN.	4	200	10
	MONNAME.	NLDATEMN.	4	200	10
	MONYY.	NLDATEYM.	6	200	10
	WEEKDATX.	NLDATEWX.	3	30	30
	WEEKDAY.	NLDATEWN.	4	200	10
	WORDDATX.	NLDATE.	10	200	20

通貨表現

通貨の概要

通貨は、その国特有の交換手段です。SAS では、通貨の書き出しと読み込みに使用する出力形式と入力形式を提供します。

米国ドル

DOLLARw.d 出力形式と入力形式は、当初、米国通貨の読み込みと書き出しを行うために導入されました。DOLLARw.d では次の記号を使用します。

- 米国通貨の先頭に通貨記号としてドル記号(\$)を使用します。
- カンマ(,)を桁区切り文字として、ピリオド(.)を小数点区切り文字として使用します。

例:

```
$12,345.00
```

DOLLARXw.d はまた、通貨の先頭にドル記号(\$)を付け、桁区切り文字としてピリオド(.)、小数点区切り文字としてカンマ(,)を使用して書き出します。ピリオドとカンマが逆の通貨出力形式は、ヨーロッパの多くの国で使用される規則です。

例:

```
$12.345,00
```

DOLLAR 出力形式および入力形式には次の制限があります。

- サポートされない通貨記号があります。
- ヨーロッパ諸国には、ピリオドとカンマが逆の通貨出力形式を使用しない国があります。
- 通貨記号の見え方はコンピュータによって違います(EBCDIC ベースのコンピュータと ASCII ベースのコンピュータでは文字の表示が違います)。

地域化されたユーロ

EUROw.d 出力形式および入力形式は、1999年に設立された欧州通貨同盟(EMU)によって制定されたユーロ通貨をサポートするために導入されました。EUROw.d では次の記号を使用します。

- ユーロ通貨データの先頭にユーロ通貨記号(e)を使用します。
- カンマ(,)を桁区切り文字として、ピリオド(.)を小数点区切り文字として使用します。

例:

```
options locale=English_UnitedKingdom;
x=12345;
put x euro10.2;
run;
```

出力:

```
e12.345,00
```

EURO 出力形式および入力形式には次の制限があります。

- ヨーロッパ諸国には、ピリオドとカンマが逆の通貨出力形式を使用しない国があります。
- ユーロは EMU のメンバーのみに制限されます。
- ロケールの具体的な値が必要となります。

カスタマイズした通貨表現

カスタマイズした通貨表現を作成するには、FORMAT プロシジャが使用できます。次の例では、オーストラリアドル、スイスフランおよびイギリスポンドに固有の出力形式を作成します。FORMAT プロシジャの詳細については、*Base SAS プロシジャガイド*を参照してください。

例のコード 8.1 通貨表現をカスタマイズする SAS コード

```
proc format;

picture aud low-<0='0,000,000,009.00'
```

```

(prefix='-AU$' mult=100)
0-high='0,000,00,009.00 '
(prefix='AU$' mult=100);

picture sfr low-<0='0,000,000,009.00'
(prefix='-SFr.' mult=100)
0-high='0,000,00,009.00 '
(prefix='-SFr.' mult=100);

picture bpd low-<0='0,000,000,009.00'
(prefix='-BPd.' mult=100)
0-high='0,000,00,009.00 '
(prefix='BPd.' mult=100);

run;
data currency;
input aud sfr bpd 12.2;
datalines;
12345 12345 12345
0 0 0
-12345 -12345 -12345
;

proc print data=currency noobs;
var aud sfr bpd;
format aud aud. sfr sfr. bpd bpd.;
title 'Unique Currency Formats';
run;

```

aud	sfr	bpd
AU\$12,345.00	SFr.12,345.00	BPd.123.45
AU\$0.00	SFr.0.00	BPd.0.00
-AU\$12,345.00	-SFr.12,345.00	-BPd.123.45

通貨表現をカスタマイズすると柔軟性は向上しますが、プログラムでの対応が必要です。

地域化された各国語および国際化対応通貨表現

NLMNYw.d および NLMNYIw.d の出力形式と入力形式は、地域化された通貨を次の 2 つの形式で表現するために導入されました。

地域化された各国語通貨表現

ロケールの地域規則と慣習が反映されます。各国語出力形式は、NLMNYw.d 出力形式および入力形式を使用して指定します。NLMNYw.d 出力形式および入力形式を使用する場合、さらに LOCALE=オプションも使用してロケールを指定する必要があります。

```

例: options locale=english_UnitedStates; data _null_;
x=12345; put x nlmny15.2; run;

```

出力:

\$12,345.00 選択した各国語通貨表現は、次の表現に従います。

表 8.2 地域化された各国語通貨表現

LOCALE=	通貨	各国語表現
English_UnitedStates	米国ドル	\$12,345.00
French_Canada	カナダドル	12 345,00 \$
French_France	フランスユーロ	12 345,00 e
French_Switzerland	スイスフラン	SFr. 12'345.00
German_Germany	ドイツユーロ	12.345,00 e
German_Luxembourg	ルクセンブルクユーロ	12.345 e
Spanish_Spain	スペインユーロ	12.345,00 e
Spanish_Venezuela	Venezuelan euros	Bs12.345,00

地域化された表示には、通貨表示に関する各国固有の地域規則が反映されます。たとえば、これらの EMU 諸国の中には、同じユーロ通貨を使用しているも通貨の表記が異なる場合があります。French_France は、桁区切り文字は使用せず、小数点区切り文字としてカンマを使用しますが、German_Germany および Spanish_Spain は桁区切り文字としてピリオドを、小数点区切り文字としてカンマを使用します。

地域化された国際化対応通貨表現

ISO 標準 4217 に準拠します。国際化対応形式は、NLMNYIw.d 出力形式および入力形式を使用して指定します。国際化対応形式は一般に、世界各国の通貨の対比を示すために使用されます。たとえば、航空券、貿易、株価などで使用されます。NLMNYIw.d 出力形式および入力形式を使用する場合、さらに LOCALE= オプションも使用してロケールを指定する必要があります。"International"を示す文字"I"が、出力形式名および入力形式名に付加されます。

例: `options locale=english_UnitedStates; data _null_;`
`x=12345; put x nlmnyi15.2; run;`出力: USD12,345.00

選択した国際化対応通貨表現は、次の表現に従います。

表 8.3 ロケール別国際化対応通貨表現(ISO 標準 4217)

LOCALE=	通貨	国際化対応表現
English_UnitedStates	米国ドル	USD12,345.00
French_Canada	カナダドル	12,345.00 CAD
French_France	フランスユーロ	12 345,00 EUR
French_Luxembourg	ルクセンブルクユーロ	12,345.00 EUR
German_Germany	ドイツユーロ	12,345.00 EUR
German_Switzerland	スイスフラン	CHF 12,345.00
Spanish_Spain	スペインユーロ	12,345.00 EUR

LOCALE=	通貨	国際化対応表現
Spanish_Venezuela	ベネズエラボリバル	VEB12,345.00

国際化対応の表示には、通貨表示に関する各国固有の地域規則も反映されません。たとえば、すべてのロケールがカンマ(,)を桁区切り文字として、ピリオド(.)を小数点として使用していても、ISO 通貨コードの位置が異なることがあります。EMU 諸国が通貨の後に通貨コードを付けるのに対し、English_UnitedStates、German_Switzerland および Spanish_Venezuela では通貨の前に ISO コードが付けられます。

ISO 標準 4217 通貨コードの全リストについては、www.bsi-global.com/Technical%2BInformation/Publications/_Publications/tig90x.doc を参照してください。

地域化された各国語および国際化対応の通貨表現を使用する場合、LOCALE=システムオプションの値に依存することが最大の制限となります。

固有の各国語および国際化対応通貨表現

NLMNLISOw.d および NLMNIISOw.d の出力形式と入力形式は、LOCALE=オプションを併用してロケールを指定しなくても各通貨を一意に表現できるように導入されました。各通貨は、固有の ISO 標準 4217 通貨コードで指定します。

固有の各国語通貨表現

固有の ISO 通貨コードで指定します。各国語出力形式は、NLMNLISOw.d 出力形式および入力形式を使用して指定します。次の例では、USD が米国ドルの ISO 通貨コードです。

注: NLMNLISOw.d 出力形式および入力形式を使用する場合、LOCALE=オプションを使用してロケールを指定しないでください。

例: `data _null_ ; x=12345 ; put x nlmnlusd15.2 ; run ;`

出力:

`us$12,345.00` 選択した固有の各国語通貨表現は、次の表現に従います。

表 8.4 ISO 通貨コード別固有の通貨表現

ISO 通貨コード	通貨	各国語表現
USD	米国ドル	USD\$12,345.00
CAD	カナダドル	CAS\$12,345.00
EUR	フランスユーロ	e12,345.00
CHF	スイスフラン	SFr.12,345.00
EUR	ドイツユーロ	e12,345.00
EUR	ルクセンブルクユーロ	e12,345.00
EUR	スペインユーロ	e12,345.00
VEB	ベネズエラボリバル	不明

通貨記号または通貨コードは、多くの通貨で先頭に付けられます。また、桁区切り文字にはカンマ、小数点区切り文字にはピリオドが使用されます。ローカル通貨の通貨記号が現在の SAS セッションのエンコーディングでサポートされていない場合、NLMNLxxxw.d 出力形式が 3 文字の ISO 通貨コードを使用して値をフォーマットします。

固有の国際化対応通貨表現

固有の ISO 通貨コードで指定します。国際化対応出力形式は、NLMNIISOw.d 出力形式および入力形式を使用して指定します。国際化対応形式は一般に、世界各国の通貨の対比を示すために使用されます。たとえば、航空券、貿易、株価などで使用されます。"International"を示す文字"I"が、出力形式名および入力形式名に付加されます。次の例では、USD が米国ドルの ISO 通貨コードです。

注: NLMNIISOw.d 出力形式および入力形式を使用する場合、LOCALE=オプションを使用してロケールを指定しないでください。

例: data _null_; x=12345; put x nlmni15.2; run;出力:
USD12,345.00

選択した国際化対応通貨表現は、次の表現に従います。

表 8.5 ISO 通貨コード別国際化対応通貨表現

ISO 通貨コード	通貨	国際化対応表現
USD	米国ドル	USD12,345.00
CAD	カナダドル	CAD12,345.00
EUR	フランスユーロ	EUR12,345.00
CHF	スイスフラン	CHF12,234.00
EUR	ドイツユーロ	EUR12,345.00
EUR	ルクセンブルクユーロ	EUR12,345.00
EUR	スペインユーロ	EUR12,345.00
VEB	ベネズエラボリバル	不明

国際化対応表示では、該当する ISO コードが通貨の先頭に付けられます。また、桁区切り文字にはカンマ、小数点区切り文字にはピリオドが使用されます。

例: 各国語および国際化対応出力形式での通貨の表現

この SAS プログラムでは、選択したアジア太平洋諸国について米国ドルに対する換算レートを使用します。出力では、各国の通貨が各国語および国際化対応出力形式を使用して表記されます。

例のコード 8.2 各国語および国際化対応出力形式で通貨を表記する SAS コード

```
data curr;

input ex_date mmddyy. usd aud hkd jpy sgd 12.2;
```

```

datalines;

061704 1.00000 1.45349 7.79930 110.110 1.71900 1

;

proc print data=curr noobs label;

var ex_date usd aud hkd jpy sgd;

format ex_date mmddyy. usd nlmnlusd15.2 aud nlmnlaud15.2 hkd
nlmnlhkd15.2
jpy nlmnljpy15.2 sgd nlmnlskd15.2;
2

label ex_date='Date' usd="US" aud='Australia' hkd='Hong Kong'

jpy='Japan' sgd='Singapore';

title 'Exchange Rates for Selected Asian-Pacific Countries
(Localized Currency Codes)';

proc print data=curr noobs label;

var ex_date usd aud hkd jpy sgd;

format ex_date mmddyy. usd nlmniusd15.2 aud nlmniaud15.2 hkd
nlmnihkd15.2
jpy nlmnijpy15.2 sgd nlmnisgd15.2;
3

label ex_date='Date' usd="US" aud='Australia' hkd='Hong Kong'

jpy='Japan' sgd='Singapore';

title 'Exchange Rates for Selected Asian-Pacific Countries
(International Currency Codes)';

run;

```

1. 2004年6月17日の時点の各換算レートが、データとしてSASプログラムに指定されます。
2. これらのNLMNLIISO出力形式は、INPUTステートメントで指定された数値データ項目にそれぞれ適用されます。これらの出力形式は、該当する各国語出力形式で通貨を表示します。
3. これらのNLMNIISO出力形式は、INPUTステートメントで指定された数値データ項目にそれぞれ適用されます。これらの出力形式は、該当する国際化対応出力形式で通貨を表示します。

画面 8.1 各国語および国際化対応出力形式の出力

Exchange Rates for Selected Asian-Pacific Countries (Localized Currency Codes)					
13:30 Monday, June 21, 2004					
Date	US	Australia	Hong Kong	Japan	Singapore
06/17/04	US\$1.00	AUS\$1.45	HK\$7.80	JPY110.11	SG\$1.72

Exchange Rates for Selected Asian-Pacific Countries (International Currency Codes)					
13:30 Monday, June 21, 2004					
Date	US	Australia	Hong Kong	Japan	Singapore
06/17/04	USD1.00	AUD1.45	HKD7.80	JPY110.11	SGD1.72

ヨーロッパ通貨換算

ヨーロッパ通貨換算の概要

SAS では、ヨーロッパ通貨を、ある国の通貨から別の国の通貨の相当する金額に換算できます。ある国の通貨をユーロに換算したり、ユーロを特定の国の通貨に換算したりすることもできます。

SAS では、通貨換算に使用する一連の出力形式、入力形式および関数が提供されます。EURFRISO の一連の出力形式を使用すると、特定のヨーロッパ通貨をユーロの金額に換算できます。ISO は、ISO 標準 4214 通貨コードを表します。ISO 標準 4217 通貨コードの全リストについては、www.bsi-global.com/Technical%2BInformation/Publications/_Publications/tig90x.doc を参照してください。

固定レートによるユーロ換算

EMU (欧州通貨同盟) は、ヨーロッパ 27 か国によって構成されます。17 か国の換算レートは固定で、EURFRISO および EURTOISO 出力形式と EUROCURR 関数に組み込まれます。次の表に、固定レートを使用する特定の国の通貨コードと換算レートを示します。

表 8.6 固定レートによるユーロ換算

ISO 通貨コード	換算レート	通貨
ATS	13.7603	オーストリアシリング
BEF	40.3399	ベルギーフラン
CYP	0.585274	キプロスポンド
DEM	1.95583	ドイツマルク
ESP	166.386	スペインペセタ
EEK	15.6466	エストニアクローン
EUR	1	ユーロ
FIM	5.94573	フィンランドマルッカ
FRF	6.55957	フランスフラン

ISO 通貨コード	換算レート	通貨
GRD	340.750	ギリシャドラクマ
IEP	0.787564	アイルランドポンド
ITL	1936.27	イタリアリラ
LUF	40.3399	ルクセンブルクフラン
MTL	0.429300	マルタリラ
NLG	2.20371	オランダギルダー
PTE	200.482	ポルトガルエスクード
SIT	239.640	スロヴェニアトラー
SKK	30.1260	スロバキアコルナ

変動レートによるユーロ換算

EMU の 13 か国については、通貨換算レートが変動します。これらの国の換算レートは、ASCII テキストファイルで保存されており、EURFR*BL ファイル参照名を使用して参照します。例えば、変動レートを variableRates.txt という名前のファイルにも保存して、そのファイルを `Filename EURFR*BL "variables.txt";` ステートメントで参照することができます。variableRates.txt の内容は次のようになっています。

```
EURFRCHF=1.5260
EURFRPLZ=1.3650
```

次のコードを使用して、ポーランドズウォティからユーロへの換算ができます。

```
data _null;
x=12345;
put x eurfrplz15.2;
run;
```

出力:
€2.939,29

例: ヨーロッパ通貨とユーロ間の換算

次の例では、ベルギーフランをユーロに換算します。EURFRBEF 出力形式は、ベルギー通貨の金額を換算レートで除算します。

```
CurrencyAmount / ExchangeRate
12345 / 40.3399
```

例のコード 8.3 コード例: ベルギーフランをユーロに換算

```
data _null_
x=12345 /*convert from Belgian francs to euros*/
put x eurfrbef15.2;
run;
```

出力:

```
e306,02
```

次の例では、ユーロをベルギーフランに換算します。EURFRBEF 出力形式は、ユーロの金額をターゲット通貨の換算レートで乗算します。

```
EurosAmount * ExchangeRate
12345 * 40.3399
```

```
data _null_
x=12345; /*convert from euros to Belgian francs*/
put x eurtobef15.2;
run;
```

出力:

```
497996.07
```

ヨーロッパ通貨間の直接換算

EUROCURR 関数は、換算レートテーブルを使用して通貨間の換算を実行します。2か国の通貨間の換算の場合、SAS では次の処理を実行します。

1. 金額をユーロに換算します。

注: SAS では、動作環境で可能な限り高い精度で、その中間値を丸めずに保存します。

2. ユーロの金額をターゲット通貨の金額に換算します。

```
SourceCurrencyAmount → EurosAmount → TargetCurrencyAmount
```

```
BelgianFrancs → euros
12345 / 40.3399 = 306.02456 euros
```

```
Euros → FrenchFrancs
306.02456 * 6.55957 = 2007.3895 French francs
```

```
data _null_;
x=eurocurr(12345,'bef','frf'); /*convert from Belgian francs to French francs*/
put x=;
run;
```

出力:

```
x=2007.389499
```

ベルギーフランをユーロに換算してから、ユーロをフランスフランに換算します。

9 章

出力形式エントリ

各国語サポート関連の出力形式のカテゴリ	74
ディクショナリ	83
\$BIDIw. 出力形式	83
\$CPTDWw. 出力形式	85
\$CPTWDw. 出力形式	85
EUROw.d 出力形式	86
EUROXw.d 出力形式	89
HDATEw. 出力形式	91
HEBDATEw. 出力形式	92
\$KANJIw. 出力形式	93
\$KANJIXw. 出力形式	94
\$LOGVSw. 出力形式	95
\$LOGVSRw. 出力形式	96
MINGUOw. 出力形式	97
NENGOw. 出力形式	99
NLBESTw. 出力形式	100
NLDATEw. 出力形式	102
NLDATEMDw. 出力形式	103
NLDATEMNw. 出力形式	104
NLDATEWw. 出力形式	105
NLDATEWNw. 出力形式	106
NLDATEYMw. 出力形式	107
NLDATEYQw. 出力形式	108
NLDATEYRw. 出力形式	109
NLDATEYWw. 出力形式	110
NLDATMw. 出力形式	111
NLDATMAPw. 出力形式	112
NLDATMDTw. 出力形式	113
NLDATMMDw. 出力形式	114
NLDATMMNw. 出力形式	114
NLDATMTMw. 出力形式	115
NLDATMTZw. 出力形式	116
NLDATMWw. 出力形式	117
NLDATMWNw. 出力形式	118
NLDATMWZw. 出力形式	118
NLDATMYMw. 出力形式	119
NLDATMYQw. 出力形式	120
NLDATMYRw. 出力形式	121
NLDATMYWw. 出力形式	122
NLDATMZw. 出力形式	123
NLMNIAEDw.d 出力形式	124

NLMNIAUDw.d 出力形式	124
NLMNIBGNw.d 出力形式	125
NLMNIBRLw.d 出力形式	126
NLMNICADw.d 出力形式	127
NLMNICHFw.d 出力形式	128
NLMNICNYw.d 出力形式	129
NLMNICZKw.d 出力形式	130
NLMNIDKKw.d 出力形式	131
NLMNIEEKw.d 出力形式	132
NLMNIEGPw.d 出力形式	133
NLMNIEURw.d 出力形式	133
NLMNIGBPw.d 出力形式	134
NLMNIHKDw.d 出力形式	135
NLMNIHRKw.d 出力形式	136
NLMNIHUFw.d 出力形式	137
NLMNIIDRW.d 出力形式	138
NLMNIILSw.d 出力形式	139
NLMNIINRW.d 出力形式	140
NLMNIJPYw.d 出力形式	141
NLMNIKRWw.d 出力形式	142
NLMNILTLw.d 出力形式	142
NLMNILVLw.d 出力形式	143
NLMNIMOPw.d 出力形式	144
NLMNIMXNw.d 出力形式	145
NLMNIMYRW.d 出力形式	146
NLMNINOKw.d 出力形式	147
NLMNINZDw.d 出力形式	148
NLMNIPLNw.d 出力形式	149
NLMNIRUBw.d 出力形式	150
NLMNISEKw.d 出力形式	151
NLMNISGDw.d 出力形式	151
NLMNITHBw.d 出力形式	152
NLMNITRYw.d 出力形式	153
NLMNITWDw.d 出力形式	154
NLMNIUSDw.d 出力形式	155
NLMNIZARw.d 出力形式	156
NLMNLAEDx.d 出力形式	157
NLMNLAUDw.d 出力形式	158
NLMNLBGNw.d 出力形式	159
NLMNLBRLw.d 出力形式	160
NLMNLCADw.d 出力形式	160
NLMNLCHFw.d 出力形式	161
NLMNLCNYw.d 出力形式	162
NLMNLCZKw.d 出力形式	163
NLMNLDDKKw.d 出力形式	164
NLMNLEEKw.d 出力形式	165
NLMNLEGPw.d 出力形式	166
NLMNLEURw.d 出力形式	167
NLMNLGBPw.d 出力形式	168
NLMNLHKDw.d 出力形式	169
NLMNLHRKw.d 出力形式	169
NLMNLHUFw.d 出力形式	170
NLMNLIDRW.d 出力形式	171
NLMNLILSw.d 出力形式	172
NLMNLINRW.d 出力形式	173
NLMNLJPYw.d 出力形式	174

NLMNLKRWw.d 出力形式	175
NLMNLLTLw.d 出力形式	176
NLMNLLVLw.d 出力形式	177
NLMNLMOPw.d 出力形式	178
NLMNLMXNw.d 出力形式	178
NLMNLMYRw.d 出力形式	179
NLMNLNOKw.d 出力形式	180
NLMNLNZDw.d 出力形式	181
NLMNLPLNw.d 出力形式	182
NLMNLRUBw.d 出力形式	183
NLMNLSEKw.d 出力形式	184
NLMNLSGDw.d 出力形式	185
NLMNLTHBw.d 出力形式	186
NLMNLTRYw.d 出力形式	187
NLMNLTWDw.d 出力形式	187
NLMNLUSDw.d 出力形式	188
NLMNLZARw.d 出力形式	189
NLMNYw.d 出力形式	190
NLMNYIw.d 出力形式	191
NLNUMw.d 出力形式	193
NLNUMIw.d 出力形式	194
NLPCTw.d 出力形式	195
NLPCTIw.d 出力形式	197
NLPCTNw.d 出力形式	198
NLPCTPw.d 出力形式	199
NLPVALUEw.d 出力形式	200
NLSTRMONw.d 出力形式	201
NLSTRQTRw.d 出力形式	202
NLSTRWKw.d 出力形式	203
NLTIMAPw. 出力形式	204
NLTIMEw. 出力形式	205
\$UCS2Bw. 出力形式	206
\$UCS2BEw. 出力形式	207
\$UCS2Lw. 出力形式	208
\$UCS2LEw. 出力形式	209
\$UCS2Xw. 出力形式	211
\$UCS2XEw. 出力形式	212
\$UCS4Bw. 出力形式	213
\$UCS4BEw. 出力形式	214
\$UCS4Lw. 出力形式	215
\$UCS4LEw. 出力形式	217
\$UCS4Xw. 出力形式	218
\$UCS4XEw. 出力形式	219
\$UESCw. 出力形式	220
\$UESCEw. 出力形式	221
\$UNCRw. 出力形式	222
\$UNCREw. 出力形式	223
\$UPARENw. 出力形式	224
\$UPARENEw. 出力形式	225
\$UTF8Xw. 出力形式	226
\$UTF8XEw. 出力形式	227
\$VSLOGw. 出力形式	228
\$VSLOGRw. 出力形式	229
WEEKUw. 出力形式	231
WEEKVw. 出力形式	232
WEEKWw. 出力形式	234

YENw.d 出力形式	235
YYWEEKUw. 出力形式	236
YYWEEKVw. 出力形式	237
YYWEEKWw. 出力形式	239

各国語サポート関連の出力形式のカテゴリ

各国語サポートに関するカテゴリ:

カテゴリ	説明
BIDI テキストの操作	データ変数から双方向のデータ値を書き出すように指示します。
文字	文字変数から文字データ値を書き出すように指示します。
通貨の換算	1つの通貨を別の通貨に変換するように指示します。
DBCS	アジア言語で使用される2バイト文字セットを変換するように指示します。
ヘブライ語のテキストの操作	データ変数からヘブライ語のデータを読み込むように指示します。
国際化対応日時	日付、時刻および日時を表す変数からデータ値を書き出すように指示します。
数値	数値変数から数値データ値を書き出すように指示します。

カテゴリ	言語要素	説明
BIDI テキストの操作	\$BIDIw. 出力形式 (p. 83)	ラテン語の単語と数字の順序を維持しつつ、ヘブライ語とアラビア語の文字の順序を逆にすることにより、論理的な順序の文字列と視覚的な順序の文字列を双方向に変換します。
	\$LOGVSw. 出力形式 (p. 95)	左から右に論理的順序で並ぶ文字列を処理し、視覚的順序でその文字列を書き出します。
	\$LOGVSRw. 出力形式 (p. 96)	右から左に論理的順序で並ぶ文字列を処理し、視覚的順序でその文字列を書き出します。
	\$VSLOGw. 出力形式 (p. 228)	視覚的順序で並ぶ文字列を処理し、その文字列を左から右の論理的順序で書き出します。
	\$VSLOGRw. 出力形式 (p. 229)	視覚的順序で並ぶ文字列を処理し、その文字列を右から左の論理的順序で書き出します。
DBCS	\$KANJIw. 出力形式 (p. 93)	シフトコードデータを DBCS データに追加します。

カテゴリ	言語要素	説明
	\$KANJIXw. 出力形式 (p. 94)	DBCS データからシフトコードデータを削除します。
数値	EUROW.d 出力形式 (p. 86)	先頭にユーロ(E)記号を付け、3桁ごとにカンマを入れ、小数部分との区切りにはピリオドを使用して数値を出力します。
	EUROXw.d 出力形式 (p. 89)	先頭にユーロ(E)記号を付け、3桁ごとにピリオドを入れ、小数部分との区切りにはカンマを使用して数値を出力します。
	NLBESTw. 出力形式 (p. 100)	ロケールに基づいて最適な数値表記を書き出します。
	NLMNIAEDw.d 出力形式 (p. 124)	アラブ首長国連邦の国際表示形式で通貨データを書き出します。
	NLMNIAUDw.d 出力形式 (p. 124)	オーストラリアの国際表示形式で通貨データを書き出します。
	NLMNIBGNw.d 出力形式 (p. 125)	ブルガリアの国際表示形式で通貨データを書き出します。
	NLMNIBRLw.d 出力形式 (p. 126)	ブラジルの国際表示形式で通貨データを書き出します。
	NLMNICADw.d 出力形式 (p. 127)	カナダの国際表示形式で通貨データを書き出します。
	NLMNICHFW.d 出力形式 (p. 128)	リヒテンシュタインとスイスの国際表示形式で通貨データを書き出します。
	NLMNICNYw.d 出力形式 (p. 129)	中国の国際表示形式で通貨データを書き出します。
	NLMNICZKw.d 出力形式 (p. 130)	チェコ共和国の国際表示形式で通貨データを書き出します。
	NLMNIDKKw.d 出力形式 (p. 131)	デンマーク、フェロー諸島およびグリーンランドの国際表示形式で通貨データを書き出します。
	NLMNIEEKw.d 出力形式 (p. 132)	エストニアの国際表示形式で通貨データを書き出します。
	NLMNIEGPw.d 出力形式 (p. 133)	エジプトの国際表示形式で通貨データを書き出します。
	NLMNIEURw.d 出力形式 (p. 133)	ベルギー、フィンランド、フランス、ドイツ、ギリシャ、アイルランド、イタリア、ルクセンブルク、マルタ、オランダ、ポルトガル、スロベニア、スペインの国際表示形式で通貨データを書き出します。
	NLMNIGBPw.d 出力形式 (p. 134)	英国の国際表示形式で通貨データを書き出します。
	NLMNIHKDw.d 出力形式 (p. 135)	香港の国際表示形式で通貨データを書き出します。

カテゴリ	言語要素	説明
	NLMNIHRKw.d 出力形式 (p. 136)	クロアチアの国際表示形式で通貨データを書き出します。
	NLMNIHUFw.d 出力形式 (p. 137)	ハンガリーの国際表示形式で通貨データを書き出します。
	NLMNIIDRW.d 出力形式 (p. 138)	インドネシアの国際表示形式で通貨データを書き出します。
	NLMNIILSw.d 出力形式 (p. 139)	イスラエルの国際表示形式で通貨データを書き出します。
	NLMNIINRW.d 出力形式 (p. 140)	インドの国際表示形式で通貨データを書き出します。
	NLMNIJPYw.d 出力形式 (p. 141)	日本の国際表示形式で通貨データを書き出します。
	NLMNIKRWw.d 出力形式 (p. 142)	韓国の国際表示形式で通貨データを書き出します。
	NLMNITLw.d 出力形式 (p. 142)	リトアニアの国際表示形式で通貨データを書き出します。
	NLMNIVLw.d 出力形式 (p. 143)	ラトビアの国際表示形式で通貨データを書き出します。
	NLMNIMOPw.d 出力形式 (p. 144)	マカオの国際表示形式で通貨データを書き出します。
	NLMNIMXNw.d 出力形式 (p. 145)	メキシコの国際表示形式で通貨データを書き出します。
	NLMNIMYRw.d 出力形式 (p. 146)	マレーシアの国際表示形式で通貨データを書き出します。
	NLMNINOKw.d 出力形式 (p. 147)	ノルウェーの国際表示形式で通貨データを書き出します。
	NLMNINZDw.d 出力形式 (p. 148)	ニュージーランドの国際表示形式で通貨データを書き出します。
	NLMNIPLNw.d 出力形式 (p. 149)	ポーランドの国際表示形式で通貨データを書き出します。
	NLMNIRUBw.d 出力形式 (p. 150)	ロシアの国際表示形式で通貨データを書き出します。
	NLMNISEKw.d 出力形式 (p. 151)	スウェーデンの国際表示形式で通貨データを書き出します。
	NLMNISGDw.d 出力形式 (p. 151)	シンガポールの国際表示形式で通貨データを書き出します。

カテゴリ	言語要素	説明
	NLMNITHBw.d 出力形式 (p. 152)	タイの国際表示形式で通貨データを書き出します。
	NLMNITRYw.d 出力形式 (p. 153)	トルコの国際表示形式で通貨データを書き出します。
	NLMNITWDw.d 出力形式 (p. 154)	台湾の国際表示形式で通貨データを書き出します。
	NLMNIUSDw.d 出力形式 (p. 155)	プエルトリコと米国の国際表示形式で通貨データを書き出します。
	NLMNIZARw.d 出力形式 (p. 156)	南アフリカの国際表示形式で通貨データを書き出します。
	NLMNLAEDx.d 出力形式 (p. 157)	アラブ首長国連邦のローカル表示形式で通貨データを書き出します。
	NLMNLAUDw.d 出力形式 (p. 158)	オーストラリアのローカル表示形式で通貨データを書き出します。
	NLMNLBGNw.d 出力形式 (p. 159)	ブルガリアのローカル表示形式で通貨データを書き出します。
	NLMNLBRLw.d 出力形式 (p. 160)	ブラジルのローカル表示形式で通貨データを書き出します。
	NLMNLCADw.d 出力形式 (p. 160)	カナダのローカル表示形式で通貨データを書き出します。
	NLMNLCHFw.d 出力形式 (p. 161)	リヒテンシュタインとスイスのローカル表示形式で通貨データを書き出します。
	NLMNLCNYw.d 出力形式 (p. 162)	中国のローカル表示形式で通貨データを書き出します。
	NLMNLCZKw.d 出力形式 (p. 163)	チェコ共和国のローカル表示形式で通貨データを書き出します。
	NLMNLDKKw.d 出力形式 (p. 164)	デンマーク、フェロー諸島およびグリーンランドのローカル表示形式で通貨データを書き出します。
	NLMNLEEKw.d 出力形式 (p. 165)	エストニアのローカル表示形式で通貨データを書き出します。
	NLMNLEGPw.d 出力形式 (p. 166)	エジプトのローカル表示形式で通貨データを書き出します。
	NLMNLEURw.d 出力形式 (p. 167)	オーストリア、ベルギー、フィンランド、フランス、ドイツ、ギリシャ、アイルランド、イタリア、ルクセンブルク、マルタ、オランダ、ポルトガル、スロベニア、スペインのローカル表示形式で通貨データを書き出します。

カテゴリ	言語要素	説明
	NLMNLGBPw.d 出力形式 (p. 168)	英国のローカル表示形式で通貨データを書き出します。
	NLMNLHKDw.d 出力形式 (p. 169)	香港のローカル表示形式で通貨データを書き出します。
	NLMNLHRKw.d 出力形式 (p. 169)	クロアチアのローカル表示形式で通貨データを書き出します。
	NLMNLHUFw.d 出力形式 (p. 170)	ハンガリーのローカル表示形式で通貨データを書き出します。
	NLMNLIDRw.d 出力形式 (p. 171)	インドネシアのローカル表示形式で通貨データを書き出します。
	NLMNLILSw.d 出力形式 (p. 172)	イスラエルのローカル表示形式で通貨データを書き出します。
	NLMNLINRw.d 出力形式 (p. 173)	インドのローカル表示形式で通貨データを書き出します。
	NLMNLJPYw.d 出力形式 (p. 174)	日本の国際表示形式で通貨データを書き出します。
	NLMNLKRWw.d 出力形式 (p. 175)	韓国のローカル表示形式で通貨データを書き出します。
	NLMNLLTLw.d 出力形式 (p. 176)	リトアニアのローカル表示形式で通貨データを書き出します。
	NLMNLLVLw.d 出力形式 (p. 177)	ラトビアのローカル表示形式で通貨データを書き出します。
	NLMNLMOPw.d 出力形式 (p. 178)	マカオのローカル表示形式で通貨データを書き出します。
	NLMNLMXNw.d 出力形式 (p. 178)	メキシコのローカル表示形式で通貨データを書き出します。
	NLMNLMYRw.d 出力形式 (p. 179)	マレーシアのローカル表示形式で通貨データを書き出します。
	NLMNLNOKw.d 出力形式 (p. 180)	ノルウェーのローカル表示形式で通貨データを書き出します。
	NLMNLNZDw.d 出力形式 (p. 181)	ニュージーランドのローカル表示形式で通貨データを書き出します。
	NLMNLPLNw.d 出力形式 (p. 182)	ポーランドのローカル表示形式で通貨データを書き出します。
	NLMNLRUBw.d 出力形式 (p. 183)	ロシアのローカル表示形式で通貨データを書き出します。

カテゴリ	言語要素	説明
	NLMNLSEKw.d 出力形式 (p. 184)	スウェーデンのローカル表示形式で通貨データを書き出します。
	NLMNLSGDw.d 出力形式 (p. 185)	シンガポールのローカル表示形式で通貨データを書き出します。
	NLMNLTHBw.d 出力形式 (p. 186)	タイのローカル表示形式で通貨データを書き出します。
	NLMNLTRYw.d 出力形式 (p. 187)	トルコのローカル表示形式で通貨データを書き出します。
	NLMNLTWDw.d 出力形式 (p. 187)	台湾のローカル表示形式で通貨データを書き出します。
	NLMNLUSDw.d 出力形式 (p. 188)	プエルトリコと米国のローカル表示形式で通貨データを書き出します。
	NLMNLZARw.d 出力形式 (p. 189)	南アフリカのローカル表示形式で通貨データを書き出します。
	NLMNYw.d 出力形式 (p. 190)	ローカル通貨を使用して、指定されたロケールに定義されたローカル表示形式で通貨データを書き出します。
	NLMNYIw.d 出力形式 (p. 191)	指定されたロケールに定義された国際表示形式を使用して通貨データを書き出します。
	NLNUMw.d 出力形式 (p. 193)	指定されたロケールに定義されたローカル表示形式で数値を書き出します。
	NLNUMIw.d 出力形式 (p. 194)	指定されたロケールに定義された国際表示形式で数値を書き出します。
	NLPCTw.d 出力形式 (p. 195)	指定されたロケールに定義されたローカル表示形式で百分率のデータを書き出します。
	NLPCTIw.d 出力形式 (p. 197)	指定されたロケールに定義された国際表示形式で百分率のデータを書き出します。
	NLPCTNw.d 出力形式 (p. 198)	百分率を書き出します。負の値にはマイナス記号を使用します。
	NLPCTPw.d 出力形式 (p. 199)	ロケール固有の数値を百分率として書き出します。
	NLPVALUEw.d 出力形式 (p. 200)	指定されたロケールのローカル表示形式で p 値を書き出します。
	NLSTRMONw.d 出力形式 (p. 201)	指定されたロケールの月名を書き出します。
	NLSTRQTRw.d 出力形式 (p. 202)	指定されたロケールの四半期の形式で数値を書き出します。

カテゴリ	言語要素	説明
	NLSTRWKw.d 出力形式 (p. 203)	指定されたロケールの曜日の形式で数値を書き出します。
	YENw.d 出力形式 (p. 235)	円記号、桁区切り、小数点付きの数値を出力します。
日付と時間	HDATEw. 出力形式 (p. 91)	yyyy mmmmm dd 形式で日付値を書き出します。dd は月の日を、mmmmm はヘブライ語の月名を、yyyy は年を表します。
	HEBDATEw. 出力形式 (p. 92)	日付値をユダヤ暦で書き出します。
	MINGUOw. 出力形式 (p. 97)	SAS 日付値を台湾の日付として yyyyymmdd 形式で書き出します。
	NENGOw. 出力形式 (p. 99)	SAS 日付値を日本の日付として e.yymmdd 形式で書き出します。
	NLDATEw. 出力形式 (p. 102)	SAS 日付値を、指定されたロケールの日付値に変換し、その日付値から日付を書き出します。
	NLDATEMDw. 出力形式 (p. 103)	SAS 日付値を、指定されたロケールの日付値に変換し、その値から月の名前と月の日を書き出します。
	NLDATEMNw. 出力形式 (p. 104)	SAS 日付値を、指定されたロケールの日付値に変換し、その日付値から月の名前を書き出します。
	NLDATEWw. 出力形式 (p. 105)	SAS 日付値を、指定されたロケールの日付値に変換し、その日付値から日付と曜日を書き出します。
	NLDATEWNw. 出力形式 (p. 106)	SAS 日付値を、指定されたロケールの日付値に変換し、その日付値から曜日を書き出します。
	NLDATEYMw. 出力形式 (p. 107)	SAS 日付値を、指定されたロケールの日付値に変換し、その値から年と月の名前を書き出します。
	NLDATEYQw. 出力形式 (p. 108)	SAS 日付値を、指定されたロケールの日付値に変換し、その値から年と四半期を書き出します。
	NLDATEYRw. 出力形式 (p. 109)	SAS 日付値を、指定されたロケールの日付値に変換し、その値から年を書き出します。
	NLDATEYWw. 出力形式 (p. 110)	SAS 日付値を、指定されたロケールの日付値に変換し、その値から年と週を書き出します。
	NLDATMw. 出力形式 (p. 111)	SAS 日時値を、指定されたロケールの日時値に変換し、その日時値から日時を書き出します。
	NLDATMAPw. 出力形式 (p. 112)	SAS 日時値を、指定されたロケールの日時値に変換し、その日時値から午前または午後付きの日時を書き出します。
	NLDATMDTw. 出力形式 (p. 113)	SAS 日時値を、指定されたロケールの日時値に変換し、その値から月の名前、月の日および年を書き出します。

カテゴリ	言語要素	説明
	NLDATMMDw. 出力形式 (p. 114)	SAS 日時値を、指定されたロケールの日時値に変換し、その値から月の名前と月の日を書き出します。
	NLDATMMNw. 出力形式 (p. 114)	SAS 日時値を、指定されたロケールの日時値に変換し、その値から月の名前を書き出します。
	NLDATMTMw. 出力形式 (p. 115)	SAS 日時値の時間部分を、指定されたロケールの時間値に変換し、その時間値から時刻を書き出します。
	NLDATMTZw. 出力形式 (p. 116)	ロケールの SAS 日時値の時間部分を、時刻とタイムゾーンに変換します。
	NLDATMWw. 出力形式 (p. 117)	SAS 日時値を、曜日および日時形式の、ロケール依存の日時文字列に変換します。
	NLDATMWNw. 出力形式 (p. 118)	SAS 日時値を、指定されたロケールの日時値に変換し、その値から曜日を書き出します。
	NLDATMWZw. 出力形式 (p. 118)	指定されたロケールの SAS 日時値を、曜日、日時、およびタイムゾーン値に変換します。
	NLDATMYMw. 出力形式 (p. 119)	SAS 日時値を、指定されたロケールの日時値に変換し、その値から年と月の名前を書き出します。
	NLDATMYQw. 出力形式 (p. 120)	SAS 日時値を、指定されたロケールの日時値に変換し、その値から年と四半期を書き出します。
	NLDATMYRw. 出力形式 (p. 121)	SAS 日時値を、指定されたロケールの日時値に変換し、その値から年を書き出します。
	NLDATMYWw. 出力形式 (p. 122)	SAS 日時値を、指定されたロケールの日時値に変換し、その値から年と週の名前を書き出します。
	NLDATMZw. 出力形式 (p. 123)	SAS 日時値を、タイムゾーンおよび日時形式の、ロケール依存の日時文字列に変換します。
	NLTIMAPw. 出力形式 (p. 204)	SAS 時間値を、指定されたロケールの時間値に変換し、その時間値から午前または午後付きの時間値を書き出します。NLTIMAP は、SAS 日時値も変換します。
	NLTIMEw. 出力形式 (p. 205)	SAS 時間値を、指定されたロケールの時間値に変換し、その時間値から時刻を書き出します。NLTIME は、SAS 日時値も変換します。
	WEEKUw. 出力形式 (p. 231)	U アルゴリズムを使用して、10 進数の週番号を書き出します。
	WEEKVw. 出力形式 (p. 232)	V アルゴリズムを使用して、10 進数の週番号を書き出します。
	WEEKWw. 出力形式 (p. 234)	W アルゴリズムを使用して、10 進数の週番号を書き出します。
	YYWEEKUw. 出力形式 (p. 236)	曜日情報を除外し、U アルゴリズムを使用して、10 進数の週番号を書き出します。

カテゴリ	言語要素	説明
	YYWEEKVw. 出力形式 (p. 237)	曜日情報を除外し、V アルゴリズムを使用して、10 進数の週番号を書き出します。
	YYWEEKWw. 出力形式 (p. 239)	曜日情報を除外し、W アルゴリズムを使用して、10 進数の週番号を書き出します。
ヘブライ語のテキストの操作	\$CPTDWw. 出力形式 (p. 85)	IBM-PC (cp862)でエンコードされたヘブライ語テキスト中の文字列を処理し、その文字列を Windows のヘブライ語エンコーディング(cp 1255)で書き出します。
	\$CPTWDw. 出力形式 (p. 85)	Windows でエンコード(cp1255)された文字列を処理し、その文字列をヘブライ語 DOS (cp862)エンコーディングで書き出します。
文字	\$UCS2Bw. 出力形式 (p. 206)	現在の SAS セッションのエンコーディングを使用する文字列を処理し、ビッグエンディアン、16 ビット、UCS2 形式の UNICODE (ユニコード)文字列を書き出します。
	\$UCS2BEw. 出力形式 (p. 207)	ビッグエンディアン、16 ビット、UCS2 形式の UNICODE (ユニコード)文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。
	\$UCS2Lw. 出力形式 (p. 208)	現在の SAS セッションのエンコーディングを使用する文字列を処理し、リトルエンディアン、16 ビット、UCS2 形式の UNICODE (ユニコード)文字列を書き出します。
	\$UCS2LEw. 出力形式 (p. 209)	リトルエンディアン、16 ビット、UCS2 形式の UNICODE (ユニコード)文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。
	\$UCS2Xw. 出力形式 (p. 211)	現在の SAS セッションのエンコーディングを使用する文字列を処理し、ネイティブエンディアン、16 ビット、UCS2 形式の UNICODE (ユニコード)文字列を書き出します。
	\$UCS2XEw. 出力形式 (p. 212)	ネイティブエンディアン、16 ビット、UCS2 形式の UNICODE (ユニコード)文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。
	\$UCS4Bw. 出力形式 (p. 213)	現在の SAS セッションのエンコーディングを使用する文字列を処理し、ビッグエンディアン、32 ビット、UCS4 形式の UNICODE (ユニコード)文字列を書き出します。
	\$UCS4BEw. 出力形式 (p. 214)	ビッグエンディアン、32 ビット、UCS4 形式の UNICODE (ユニコード)文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。
	\$UCS4Lw. 出力形式 (p. 215)	現在の SAS セッションのエンコーディングを使用する文字列を処理し、リトルエンディアン、32 ビット、UCS4 形式の UNICODE (ユニコード)文字列を書き出します。
	\$UCS4LEw. 出力形式 (p. 217)	リトルエンディアン、32 ビット、UCS4 形式の UNICODE (ユニコード)文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。

カテゴリ	言語要素	説明
	\$UCS4Xw. 出力形式 (p. 218)	現在の SAS セッションのエンコーディングを使用する文字列を処理し、ネイティブエンディアン、32 ビット、UCS4 形式の UNICODE (ユニコード)文字列を書き出します。
	\$UCS4XEw. 出力形式 (p. 219)	ネイティブエンディアン、32 ビット、UCS4 形式の UNICODE (ユニコード)で文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。
	\$UESCw. 出力形式 (p. 220)	現在の SAS セッションでエンコードされる文字列を処理し、その文字列を Unicode エスケープ(UESC)表記で書き出します。
	\$UESCEw. 出力形式 (p. 221)	Unicode エスケープ(UESC)表記の文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。
	\$UNCRw. 出力形式 (p. 222)	現在の SAS セッションでエンコードされる文字列を処理し、その文字列を数値文字表現(NCR)で書き出します。
	\$UNCREw. 出力形式 (p. 223)	数値文字表現(NCR)の文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。
	\$UPARENw. 出力形式 (p. 224)	現在の SAS セッションでエンコードされる文字列を処理し、その文字列を Unicode かつこ(UPAREN)表記で書き出します。
	\$UPARENEw. 出力形式 (p. 225)	Unicode かつこ(UPAREN)内の文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。
	\$UTF8Xw. 出力形式 (p. 226)	現在の SAS セッションのエンコーディングを使用する文字列を処理し、UTF-8 (Universal Transformation Format)エンコーディングの文字列を書き出します。
	\$UTF8XEw. 出力形式 (p. 227)	UTF-8 (Universal Transformation Format)の文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。

ディクショナリ

\$BIDIw. 出力形式

ラテン語の単語と数字の順序を維持しつつ、ヘブライ語とアラビア語の文字の順序を逆にすることにより、論理的な順序の文字列と視覚的な順序の文字列を双方向に変換します。

カテゴリ: BIDI テキストの操作

配置: 左

構文

\$BIDIw.

説明

w

出力フィールドの幅を指定します。

デフォルト: *w* が指定されていない場合は 1

範囲: 1–32767

詳細

Windows の動作環境では、ヘブライ語とアラビア語のテキストが論理的な順序で保存されます。テキストは書き込まれるときの順序で保存され、表示されるときの順序で保存されるときとは限りません。ただし、その他の動作環境でのヘブライ語のテキストは、表示されるときと同じ順序で保存されます。SAS ユーザーは、逆順のヘブライ語とアラビア語のテキストを検出できます。このような状況は、SAS/CONNECT または他のソフトウェアを使用して、ヘブライ語およびアラビア語のテキストによる SAS データセットまたはレポートを表示動作環境から論理動作環境に転送するときが発生する可能性があります。\$BIDIw 出力形式は、数字および Latin-1 単語の順序を維持しながらヘブライ語とアラビア語のテキストを逆順にする出力形式です。

動作環境の情報

汎用機動作環境でのこの出力形式は、NewCode Hebrew および Arabic を使用できるように設計されています。一部の汎用機動作環境では、OldCode Hebrew または Arabic エンコーディングを使用するため、不満足な結果になる場合があります。このエンコーディングの Hotfix は、[SAS Institute の Web サイト](#)にあります。

比較

\$BIDIw 出力形式では、文字データを逆順に書き出して空白を保持する、\$REVERJw 出力形式と似た逆転処理が実行されます。\$BIDIw は、次のように動作します。

- \$BIDIw より、空白を保持したまま、指定した文字列の単語と数字の順序が逆になります。Latin-1 の単語と数字自体は逆順にならず、文字列での順序のみが逆になります。
- テキスト文字列にヘブライ語またはアラビア語で構成される単語が登場すると、\$BIDI により、ヘブライ語またはアラビア語の単語の文字が逆順になり、文字列中のヘブライ語またはアラビア語の単語の位置が逆になります。

例

この例は、\$BIDIw によってヘブライ語文字が逆順になる方法を示しています。文字列内のヘブライ語は逆順になります。単語内のヘブライ語文字も逆になります。

```
data;
  a='שלום כי תהיה נ abc 123';
  b1 = put (a,$bidi20.);
  put b=;
  b2 = put (b,$bidi20.);
  put b=;
run;
```

次の行が SAS ログに書き込まれます。

```
b1=123 abc נ תהיה ושלום
b2=שלום כי תהיה נ abc 123
```

\$CPTDWw. 出力形式

IBM-PC (cp862)でエンコードされたヘブライ語テキスト中の文字列を処理し、その文字列を Windows のヘブライ語エンコーディング(cp 1255)で書き出します。

カテゴリ: ヘブライ語のテキストの操作

配置: 左

構文

\$CPTDWw.

説明

w
出力フィールドの幅を指定します。

デフォルト: 200

範囲: 1-32767

比較

\$CPTDWw. 出力形式では、\$CPTWDw.出力形式とは逆の処理を行います。

例

次の例では、入力値として"808182x"を使用します。

ステートメント	結果
	----+-----1-----+
put text \$cptdw3.;	אבא

関連項目:

出力形式:

- “\$CPTWDw. 出力形式” (85 ページ)

入力形式:

- “\$CPTDWw. 入力形式” (328 ページ)
- “\$CPTWDw. 入力形式” (329 ページ)

\$CPTWDw. 出力形式

Windows でエンコード(cp1255)された文字列を処理し、その文字列をヘブライ語 DOS (cp862)エンコーディングで書き出します。

カテゴリ: ヘブライ語のテキストの操作

配置: 左

構文

`$CPTDWw.`

説明

`w`
出力フィールドの幅を指定します。

デフォルト: 200

範囲: 1-32767

比較

`$CPTDWw.`出力形式では、`$CPTDWw.`出力形式とは逆の処理を行います。

例

次の例では、入力値として“1234”を使用します。

ステートメント	結果
	----+-----1-----+-----2-----+
<code>put text \$cptwd3.;</code>	€□,

関連項目:

出力形式:

- “[\\$CPTDW_w. 出力形式](#)” (85 ページ)

入力形式:

- “[\\$CPTDW_w. 入力形式](#)” (328 ページ)
- “[\\$CPTDW_w. 入力形式](#)” (329 ページ)

EUROW.d 出力形式

先頭にユーロ(E)記号を付け、3桁ごとにカンマを入れ、小数部分との区切りにはピリオドを使用して数値を出力します。

カテゴリ: 数値

配置: 右

構文

EUROw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

範囲: 1-32

ヒント: 出力にユーロ記号を含める場合、適切な幅を選択していることを確認してください。

d

数値の小数点以下の桁数を指定します。

デフォルト: 0

範囲: 0-31

要件 *w* より小さい数にします。

比較

- EUROw.d 出力形式は EUROXw.d 出力形式に似ていますが、EUROXw.d 出力形式では小数点とカンマの役割が逆になっています。この規則はヨーロッパの国で共通です。
- EUROw.d 出力形式は DOLLARw.d 出力形式に似ていますが、DOLLARw.d 出力形式は先頭にユーロ記号のかわりにドル記号を付けます。

例

この例では、金額の値に 1254.71 を使用します。

ステートメント	結果
	----+-----1----+-----2----+-----3
put amount euro10.2;	E1,254.71
put amount euro5.;	1,255
put amount euro9.2;	E1,254.71
put amount euro15.3;	E1,254.710

```
data _null_;
input x;
put x euro10.2;
put x euro5. ;
put x euro9.2;
put x euro15.3;
datalines;
1254.71
;
run;
SAS Log:
```

```

E1,254.71
1,255
E1,254.71
E1,254.710

/* This code determines the default length. */
data _null_;
input x;
put x euro.;
datalines;
1
22
333
4444
55555
666666
7777777
88888888
999999999
1234561234
;run;
SAS Log:
datalines;
E1
E22
E333
E4,444
55,555
666666
7.78E6
8.89E7
1E9
1.23E9
NOTE: At least one W.D format was too small for the number to be printed.
The decimal may be shifted by the "BEST" format.

/* This code determines the range. */
data _null_;
input x;
put x euro5.;
put x euro6.;
put x euro7.;
put x euro8.;
put x euro9.;
put x euro9.2;
put x euro10.;
put x euro10.2;
put x euro10.4;
put x euro11.;
put x euro11.3;
put x euro12.;
put x euro12.2;
put x euro13.;
put x euro13.2;
datalines;
333
4444

```

```
55555
666666
7777777
88888888
999999999
1234561234
;run;
```

関連項目:

出力形式:

- “EUROXw.d 出力形式” (89 ページ)

入力形式:

- “EUROw.d 入力形式” (330 ページ)
- “EUROXw.d 入力形式” (332 ページ)

EUROXw.d 出力形式

先頭にユーロ(E)記号を付け、3 桁ごとにピリオドを入れ、小数部分との区切りにはカンマを使用して数値を出力します。

カテゴリ: 数値

配置: 右

構文

EUROXw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

範囲: 1-32

ヒント: 出力にユーロ記号を含める場合、適切な幅を選択していることを確認してください。

d

数値の小数点以下の桁数を指定します。

デフォルト: 0

範囲: 0-31

要件 *w* より小さい数にします。

比較

- EUROXw.d 出力形式は EUROw.d 出力形式に似ていますが、EUROw.d 出力形式では小数点とカンマの役割が逆になっています。この規則は英語圏の国で共通です。

- EUROX $w.d$ 出力形式は DOLLARX $w.d$ 出力形式に似ていますが、DOLLARX $w.d$ 出力形式は先頭にユーロ記号のかわりにドル記号を付けます。

例

この例では、金額の値に 1254.71 を使用します。

ステートメント	結果
	----+----1-----+----2-----+----3
put amount eurox10.2;	E1.254,71
put amount eurox5.;	1.255
put amount eurox9.2;	E1.254,71
put amount eurox15.3;	E1.254,710

```

data _null_;
input x;
put x eurox10.2;
put x eurox5.;
put x eurox9.2;
put x eurox15.3;
datalines;
1254.71
; run;
SAS Log:
E1.254,71
1.255
E1.254,71
E1.254,710

/* This code determines the default length. */
data _null_;
input x;
put x eurox.;
datalines;
1
22
333
4444
55555
666666
7777777
88888888
999999999
1234561234
;run;
SAS Log:
E1
E22
E333
E4.444

```

55.555
 666666
 7.78E6
 8.89E7
 1E9
 1.23E9

注: 1つ以上の W.D 出力形式において、数値を出力するには幅の長さが十分ではありませんでした。"BEST"出力形式が使用されて、10 進数のシフトが行われました。

関連項目:

出力形式:

- “EUROw.d 出力形式” (86 ページ)

入力形式:

- “EUROw.d 入力形式” (330 ページ)
- “EUROXw.d 入力形式” (332 ページ)

HDATEw. 出力形式

yyyy mmmmm dd 形式で日付値を書き出します。dd は月の日を、mmmmm はヘブライ語の月名を、yyyy は年を表します。

カテゴリ: 日付と時間

配置: 右

構文

HDATEw.

説明

w

出力フィールドの幅を指定します。

注: 最善の表示状態にするため、幅には 9、11、15、17 を使用してください。

デフォルト: 17

範囲: 9–17

詳細

HDATEw. 出力形式では、SAS 日付値を yyyy mmmmm dd 形式で書き出します。

yyyy

年です。

mmmmm

ヘブライ語での月名です。

dd

月の日です。

例

次の例では、15780 を入力値として使用します。この値は 2003 年 3 月 16 日に相当する SAS 日付値です。

ステートメント	結果
	-----1-----2-----+
put day hdate9.;	03 16 16
put day hdate11.;	2003 16 16
put day hdate17.;	2003 16 16

関連項目:

出力形式:

- “HEBDATEw. 出力形式” (92 ページ)

HEBDATEw. 出力形式

日付値をユダヤ暦で書き出します。

カテゴリ: 日付と時間

配置: 右

構文

HEBDATEw.

説明

w

出力フィールドの幅を指定します。

デフォルト: 16

範囲: 7–24

詳細

ユダヤ暦は、太陽暦と太陰暦を組み合わせた暦です。年は天地創造以来の年数として計算されます。ユダヤの歴史によれば、天地創造はキリスト教の始まりより 3760 年と 3 か月前です。ユダヤ暦の年数を計算するには、グレゴリオ暦の指定された年の秋を起点として 3761 年を追加する必要があります。

HEBDATEw. 出力形式では、SAS 日付値をユダヤ暦に従って書き出します。日付は次のいずれかの出力形式で書き出されます。

long

ראשון י' אדר ה'תשס"ג

デフォルト
 'א אדר תשס"ג

short
 'א/ב'תשס"ג

例

次の例では、15780 を入力値として使用します。この値は 2003 年 3 月 16 日に相当する SAS 日付値です。

ステートメント	結果
	----+-----1-----+
put day hebdate13.;	א תשס"ג / '21 / ב"א
put day hebdate16.;	א תשס"ג 'ב-אדר אב"א
put day hebdate24.;	א תשס"ג 'ב-אדר אב"א 1701

関連項目:

入力形式:

- [“HDATEw. 出力形式” \(91 ページ\)](#)

\$KANJIw. 出力形式

シフトコードデータを DBCS データに追加します。

カテゴリ: DBCS

配置: 左

構文

\$KANJIw.

説明

w

出力フィールドの幅を指定します。

範囲: 出力形式の最小幅は、2 + (現在の DBCSTYPE=設定で使用されるシフトコードの長さ)*2 です。

制限事項: 幅は偶数幅(2 バイト単位)で指定します。奇数幅を指定すると幅が切り捨てられます。この幅は、シフトコードデータの長さ以上にする必要があります。

詳細

\$KANJI 出力形式では、シフトコードデータが、シフトコードデータのない DBCS データに追加されます。入力データが空白の場合、シフトコードデータは追加されません。

\$KANJI 出力形式で処理されるのはホストメインフレームデータですが、\$KANJI は他のプラットフォームで使用できます。\$KANJI 出力形式を非 EBCDIC (モーダルエンコーディングではない)ホストで使用する場合、データは変化しません。

関連項目:

出力形式:

- “\$KANJIw. 出力形式” (94 ページ)

入力形式:

- “\$KANJIw. 入力形式” (334 ページ)
- “\$KANJIw. 入力形式” (334 ページ)

システムオプション:

- “DBCSTYPE システムオプション: UNIX、Windows および z/OS” (462 ページ)

\$KANJIw. 出力形式

DBCS データからシフトコードデータを削除します。

カテゴリ: DBCS

配置: 左

構文

\$KANJIw.

説明

w

出力フィールドの幅を指定します。

範囲: 出力形式の最小幅は 2 です。

制限事項: 幅は偶数幅(2 バイト単位)で指定します。奇数幅を指定すると幅が切り捨てられます。この幅は、シフトコードデータの長さ以上にする必要があります。

詳細

\$KANJIX 出力形式では、DBCS データからシフトコードデータが削除されます。入力データ長は、 $2 + (SO/SI \text{ 長}) * 2$ である必要があります。SBCS データが返された場合を除き、データは SO で開始し、SI で終了する必要があります。

\$KANJIX 出力形式で処理されるのはホストメインフレームデータですが、\$KANJIX は他のプラットフォームで使用できます。\$KANJIX 出力形式を非 EBCDIC (モーダルエンコーディングではない)ホストで使用する場合、データは変化しません。

関連項目:**出力形式:**

- “\$KANJIw. 出力形式” (93 ページ)

入力形式:

- “\$KANJIw. 入力形式” (334 ページ)
- “\$KANJIXw. 入力形式” (334 ページ)

システムオプション:

- “DBCSTYPE システムオプション: UNIX、Windows および z/OS” (462 ページ)

\$LOGVSw. 出力形式

左から右に論理的順序で並ぶ文字列を処理し、視覚的順序でその文字列を書き出します。

カテゴリ: BIDI テキストの操作

配置: 左

構文

\$LOGVSw.

説明

w

出力フィールドの幅を指定します。

デフォルト: 200

範囲: 1–32767

詳細

\$LOGVSw.出力形式は、論理的順序のテキストを視覚サーバーに保存するときに使用します。

注: If the \$LOGVSw.出力形式を使用できない場合、データのヘブライ語またはアラビア語の部分は逆順になります。

比較

\$LOGVSw.出力形式では、\$LOGVSRw.出力形式とは逆の処理を行います。

例

次の例では、“flight”のヘブライ語の入力値 תִּירוּס を使用します。

ステートメント	結果
	----+----1----+----2----+
put text \$logvs12.;	ﻁﻰ ﻁﻰ ﻁﻰ flight

次の例では、“computer”のアラビア語の入力値 **ﻁﺎﺗ** を使用します。

ステートメント	結果
	----+----1----+----2----+
put text \$logvs12.;	ﻁﺎﺗ computer

関連項目:

出力形式:

- “\$LOGVSRw. 出力形式” (96 ページ)

入力形式:

- “\$LOGVSRw. 入力形式” (336 ページ)
- “\$LOGVSw. 入力形式” (335 ページ)

\$LOGVSRw. 出力形式

右から左に論理的順序で並ぶ文字列を処理し、視覚的順序でその文字列を書き出します。

カテゴリ: BIDI テキストの操作

配置: 左

構文

\$LOGVSRw.

説明

w 出力フィールドの幅を指定します。

デフォルト: 200

範囲: 1–32767

詳細

\$LOGVSRw.出力形式は、論理的順序のテキストを視覚サーバーに保存するときに使用します。テキストのヘブライ語とアラビア語の部分は、\$テキストのヘブライ語とアラ

ビア語の部分は、\$LOGVSw. 出力形式がサーバーにない場合には、逆順になります。

比較

\$LOGVSRw.出力形式では、\$LOGVSw.出力形式とは逆の処理を行います。

例

次の例では、“flight”のヘブライ語の入力値 **תִּירוּם** を使用します。

ステートメント	結果
	----+-----1-----+
put text \$logvsr12.;	flight תִּירוּם

次の例では、“computer”のアラビア語の入力値 **كأذ** を使用します。

ステートメント	結果
	----+-----1-----+
put text \$logvsr12.;	كأذ computer

関連項目:

出力形式:

- “\$LOGVSw. 出力形式” (95 ページ)

入力形式:

- “\$LOGVSw. 入力形式” (335 ページ)
- “\$LOGVSRw. 入力形式” (336 ページ)

MINGUOw. 出力形式

SAS 日付値を台湾の日付として `yyyymmdd` 形式で書き出します。

カテゴリ: 日付と時間

配置: 左

構文

MINGUOw.

説明**w**

出力フィールドの幅を指定します。

デフォルト: 8**範囲:** 1-10**詳細**MINGUOw.出力形式では、SAS 日付値を *yyyymmdd* 形式で書き出します。*yyyy*

年を表す整数です。

mm

月を表す整数です。

dd

月の日を表す整数です。

台湾のカレンダーでは、1912 年を基準年(1912 年 1 月 1 日が 01/01/01)として使用します。1912 年より前の日付は、一連のアスタリスクで表示されます。年の値は 100 年周期で繰り返されることはありません。年の値は、継続して増加していきます。

例

例の表では、次の入力値を使用します。

- 12054 は、1993 年 1 月 1 日に対応する SAS 日付値です。
- 18993 は、2012 年 1 月 1 日に対応する SAS 日付値です。
- -20088 は、1905 年 1 月 1 日に対応する SAS 日付値です。

ステートメント	結果
<code>x=put(12054,minguo7.); put x=;</code>	x=820101
<code>x=put(12054,minguo9.); put x=;</code>	x=82/01/01
<code>x=put(12054,minguo10.); put x=;</code>	x=0082/01/01
<code>x=put(18993,minguo7.); put x=;</code>	x=1000101
<code>x=put(18993,minguo9.); put x=;</code>	x=100/01/01
<code>x=put(18993,minguo10.); put x=;</code>	x=0101/01/01
<code>x=put(-20088,minguo7.); put x=;</code>	*****
<code>x=put(-20088,minguo9.); put x=;</code>	*****

ステートメント	結果
x=put(-20088,minguo10.); put x=;	*****

関連項目:

入力形式:

- “MINGUOW. 入力形式” (337 ページ)

NENGOW. 出力形式

SAS 日付値を日本の日付として *e.yymmdd* 形式で書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NENGOW.

説明

w

出力フィールドの幅を指定します。

デフォルト: 10

範囲: 2–10

詳細

NENGOW.出力形式では、SAS 日付値を *e.yymmdd* 形式で書き出します。

e

元号(明治、大正、昭和、平成)の先頭の文字(M、T、S、H)です。

yy

年を表す整数です。

mm

月を表す整数です。

dd

月の日を表す整数です。

幅が小さすぎる場合は、ピリオドが省略されます。

例

例の表では、15342 を入力値として使用します。この値は 2002 年 1 月 2 日に相当する SAS 日付値です。

```

data _null_;
date=15342;
put date nengo3.;
put date nengo6.;
put date nengo8.;
put date nengo9.;
put date nengo10.;
run

```

ステートメント	結果
	----+-----1
put date nengo3.;	H14
put date nengo6.;	H14/01
put date nengo8.;	H.140102
put date nengo9.;	H14/01/02
put date nengo10.;	H.14/01/02

関連項目:

入力形式:

- [“NENGOW. 入力形式” \(339 ページ\)](#)

NLBESTw. 出力形式

ロケールに基づいて最適な数値表記を書き出します。

カテゴリ: 数値

配置: 右

構文

NLBESTw.

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 1-32

ヒント: 0 から .01 の数字(端点を除く)を出力する場合、過度に丸めないように 7 以上のフィールド幅を使用します。0 から -.01 の数字(端点を除く)を出力する場合は、8 以上のフィールド幅を使用します。

詳細

NLBEST 出力形式は、ロケールの小数点と記号の位置に基づいて、最も適切な数値を書き出します。NLBEST は BEST 出力形式に似ています。詳細については、*SAS 出力形式と入力形式: リファレンス*の BEST 出力形式を参照してください。

例

次に、コードのロケールに基づいた結果を示します。

```
x=-1257000
put x nlbest6.;
put x nlbest3.;
put "=====";
x=-0.1
put x nlbest6.;
put x nlbest3.;
put "=====";
x=0.1
put x nlbest6.;
put x nlbest3.;
put "=====";
x=1257000
put x nlbest6.;
put x nlbest3.;
```

ロケール	結果
locale=English_UnitedStates	-126E4

	=====
	-0.1
	-.1
	=====
	0.1
	0.1
	=====
	1.26E6
	1E6

ロケール	結果
locale=German_Germany	-126E4

	=====
	-0,1
	-,1
	=====
	0,1
	0,1
	=====
	1,26E6
1E6	
locale=ar_BH	126E4-

	=====
	0.1-
	.1-
	=====
	0.1
	0.1
	=====
	1.26E6
1E6	

NLDATE w . 出力形式

SAS 日付値を、指定されたロケールの日付値に変換し、その日付値から日付を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATE w .

説明

w

出力フィールドの幅を指定します。必要に応じて、出力形式の幅に合うように日付が短縮されます。

デフォルト: 20

範囲: 10–200

比較

NLDATMDw.出力形式は DATEw.出力形式と WORDDATEw.出力形式に似ていますが、NLDATMDw.出力形式がロケール固有である点が違います。

例

これらの例では、15760 を入力値として使用します。この値は、2003 年 2 月 24 日に相当する SAS 日付値です。

ステートメント	結果
	----+----1----+----2
options locale=English_UnitedStates; put day nldate.;	February 24, 2003
options locale=German_Germany; put day nldate.;	24. Februar 2003

関連項目:

出力形式:

- “NLDATMDNw. 出力形式” (104 ページ)
- “NLDATMDWw. 出力形式” (105 ページ)
- “NLDATMDWNw. 出力形式” (106 ページ)

NLDATMDw. 出力形式

SAS 日付値を、指定されたロケールの日付値に変換し、その値から月の名前と月の日を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMDw.

説明

w
出力フィールドの幅を指定します。

デフォルト: 16

範囲: 6-200

例

この例では、en_US ロケールオプションを使用します。

ステートメント	結果
put 1 nldatemd.;	January 02

関連項目:

出力形式:

- [“NLDATEYMw. 出力形式” \(107 ページ\)](#)

NLDATEMNw. 出力形式

SAS 日付値を、指定されたロケールの日付値に変換し、その日付値から月の名前を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATEMN w .

説明

w

出力フィールドの幅を指定します。必要に応じて、出力形式の幅に合うように月名が短縮されます。

デフォルト: 10

範囲: 4–200

比較

NLDATEMN w .出力形式は MONNAME w .出力形式と似ていますが、NLDATEMN w .出力形式がロケール固有である点が違います。

例

これらの例では、15760 を入力値として使用します。この値は、2003 年 2 月 24 日に相当する SAS 日付値です。

ステートメント	結果
	----+-----1
options locale=English_UnitedStates; put month nldatemn.;	February
options locale=German_Germany; put month nldatemn.;	Februar

関連項目:

出力形式:

- “NLDATEw. 出力形式” (102 ページ)
- “NLDATEWw. 出力形式” (105 ページ)
- “NLDATEWNw. 出力形式” (106 ページ)

NLDATEWw. 出力形式

SAS 日付値を、指定されたロケールの日付値に変換し、その日付値から日付と曜日を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATEWw.

説明

w

出力フィールドの幅を指定します。必要に応じて、出力形式の幅に合うように日付と曜日が短縮されます。

デフォルト: 20

範囲: 10–200

比較

NLDATEWw.出力形式は WEEKDATEw.出力形式と似ていますが、NLDATEWw.出力形式がロケール固有である点が違います。

例

これらの例では、15760 を入力値として使用します。この値は、2003 年 2 月 24 日に相当する SAS 日付値です。

ステートメント	結果
	----+-----1-----+-----2
options locale=English_UnitedStates;	Mon, Feb 24, 03
date=15760;	Mon, Feb 24, 03
x=put(date,nldatew.);	Monday, February 24, 2003
y=put(date,nldatew20.);	
z=put(date,nldatew200.);	
run;	

ステートメント	結果
options locale=German_Germany;	Mo, 24. Feb 03
date=15760;	
x=put(date,nldatew.);	Mo, 24. Feb 03
y=put(date,nldatew20.);	
z=put(date,nldatew200.);	Montag, 24. Februar 2003
run;	

関連項目:

出力形式:

- “NLDATEw. 出力形式” (102 ページ)
- “NLDATEMNw. 出力形式” (104 ページ)
- “NLDATEWNw. 出力形式” (106 ページ)

NLDATEWNw. 出力形式

SAS 日付値を、指定されたロケールの日付値に変換し、その日付値から曜日を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATEWN_w.

説明

w

出力フィールドの幅を指定します。必要に応じて、出力形式の幅に合うように曜日が短縮されます。

デフォルト: 10

範囲: 4–200

比較

NLDATEWN_w.出力形式は DOWNAME_w.出力形式と似ていますが、NLDATEWN_w.出力形式がロケール固有である点が違います。

例

これらの例では、15760 を入力値として使用します。この値は、2003 年 2 月 24 日に相当する SAS 日付値です。

ステートメント	結果
	----+----1
options locale=English_UnitedStates; put date nldatewn.;	Monday
options locale=German_Germany; put date nldatewn.;	Montag

関連項目:

出力形式:

- “NLDATEw. 出力形式” (102 ページ)
- “NLDATEMNw. 出力形式” (104 ページ)
- “NLDATEWw. 出力形式” (105 ページ)

NLDATEYMw. 出力形式

SAS 日付値を、指定されたロケールの日付値に変換し、その値から年と月の名前を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATEYM^w.

説明

^w

出力フィールドの幅を指定します。

デフォルト: 16

範囲: 6–200

例

この例では、spanish_Spain ロケールオプションを使用します。

ステートメント	結果
---------	----

```
options locale=spanihs_Spain;
data_null;                                agosto de 2010
dy=today();                                ago de 10
x=put(dy, nldateym.);                       agosto de 2010
y=put(dy, nldateym12.);
z=put(dy, nldateym200.);
run;
```

関連項目:

出力形式:

- “[NLDATEMDw. 出力形式](#)” (103 ページ)

NLDATEYQw. 出力形式

SAS 日付値を、指定されたロケールの日付値に変換し、その値から年と四半期を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATEYQ_w.

説明

w
出力フィールドの幅を指定します。

デフォルト: 16

範囲: 4-200

例

この例では、fr_FR ロケールオプションを使用します。

ステートメント	結果
options locale=fr_FR;	
data _null_;	
dy=today();	+--- NLDATEYQ min=4 default=16
dt=datetime();	max=200 ---+
put "+--- NLDATEYQ min=4 default=16	16 T3 08
max=200 ---+";	4 ****
put '16' +5 dy nldateyq.;	14 T3 08
put '4' +5 dy nldateyq4.;	32 3e trimestre 2008
put '14' +5 dy nldateyq14.;	200
put '32' +5 dy nldateyq32.;	3e trimestre 2008
put '200' +5 dy nldateyq200.;	
run;	

NLDATEYRw. 出力形式

SAS 日付値を、指定されたロケールの日付値に変換し、その値から年を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATEYR w .

説明

w

出力フィールドの幅を指定します。

デフォルト: 16

範囲: 2–200

例

この例では、fr_FR ロケールオプションを使用します。

ステートメント	結果
options locale=fr_FR;	
data _null_;	+--- NLDATEYR min=2 default=16 max=200
dy=today();	---+
dt=datetime();	2008
put "+--- NLDATEYR min=2 default=16	08
max=200 ---+";	2008
put dy nldateyr.;	2008
put dy nldateyr2.;	
put dy nldateyr8.;	
put dy nldateyr200.;	
run;	

NLDATEYWw. 出力形式

SAS 日付値を、指定されたロケールの日付値に変換し、その値から年と週を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATEYWw.

説明

w
出力フィールドの幅を指定します。

デフォルト: 16

範囲: 5–200

例

この例では、fr_FR ロケールオプションを使用します。

ステートメント	結果
options locale=fr_FR;	
data _null_;	16 Week 33 2008
dy=today();	5 *****
dt=datetime();	8 W33 08
put "+--- NLDATEYW min=5 default=16 max=200 ---+";	32 Week 33 2008
put '16' +5 dy nldateyw.;	200
put '5' +5 dy nldateyw5.;	Week 33 2008
put '8' +5 dy nldateyw8.;	
put '32' +5 dy nldateyw32.;	
put '200' +5 dy nldateyw200.;	
run;	

NLDATMw. 出力形式

SAS 日時値を、指定されたロケールの日時値に変換し、その日時値から日時を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATM_w.

説明

w

出力フィールドの幅を指定します。必要に応じて、出力形式の幅に合うように日時値が短縮されます。

デフォルト: 30

範囲: 10–200

比較

NLDATM_w出力形式は DATETIME_w出力形式に似ていますが、NLDATM_w出力形式がロケール固有である点が違います。

例

これらの例では、1361709583 を入力値として使用します。この値は、2003 年 2 月 24 日の午後 12:39:43 に相当する SAS 日時値です。

ステートメント	結果
	----+----1----+----2----+----3
options locale=English_UnitedStates; put day nldatm.;	24Feb03:12:39:43
options locale=German_Germany; put day nldatm.;	24. Februar 2003 12.39 Uhr

関連項目:

出力形式:

- “NLDATMAP_w. 出力形式” (112 ページ)
- “NLDATMTM_w. 出力形式” (115 ページ)
- “NLDATMW_w. 出力形式” (117 ページ)

NLDATMAP_w. 出力形式

SAS 日時値を、指定されたロケールの日時値に変換し、その日時値から午前または午後付きの日時を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMAP_w.

説明

w

出力フィールドの幅を指定します。必要に応じて、出力形式の幅に合うように日時値が短縮されます。

デフォルト: 32

範囲: 16–200

比較

NLDATMAP_w.出力形式は DATEAMPM_w.出力形式と似ていますが、NLDATMAP_w.出力形式がロケール固有である点が違います。

例

これらの例では、1361709583 を入力値として使用します。この値は、2003 年 2 月 24 日の午後 12:39:43 に相当する SAS 日時値です。

ステートメント	結果
	----+----1----+----2----+----3
options locale=English_UnitedStates; put event nldatmap.;	February 24, 2003 12:39:43 PM
options locale=Spanish_Mexico; put event nldatmap.;	24 de febrero de 2003 12:39:43 PM

関連項目:

出力形式:

- “NLDATMw. 出力形式” (111 ページ)
- “NLDATMTMw. 出力形式” (115 ページ)
- “NLDATMWw. 出力形式” (117 ページ)

NLDATMDTw. 出力形式

SAS 日時値を、指定されたロケールの日時値に変換し、その値から月の名前、月の日および年を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMDTw.

説明

w
出力フィールドの幅を指定します。

デフォルト: 20

範囲: 10-200

例

この例では、en_US ロケールオプションを使用します。

ステートメント	結果
put 86400,nldatmdt.;	January 02, 1960
put 86400,dtdate.;	02JAN60

関連項目:

出力形式:

- [“NLDATMMDw. 出力形式” \(114 ページ\)](#)

NLDATMMDw. 出力形式

SAS 日時値を、指定されたロケールの日時値に変換し、その値から月の名前と月の日を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMMDw.

説明

w
出力フィールドの幅を指定します。

デフォルト: 16

範囲: 6–200

例

この例では、en_US ロケールオプションを使用します。

ステートメント	結果
put 86400 nldatmmd.;	January 02

関連項目:

出力形式:

- [“NLDATMYMw. 出力形式” \(119 ページ\)](#)

NLDATMMNw. 出力形式

SAS 日時値を、指定されたロケールの日時値に変換し、その値から月の名前を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMMNw.

説明**w**

出力フィールドの幅を指定します。

デフォルト: 10**範囲:** 4–200**例**

この例では、en_US ロケールオプションを使用します。

ステートメント	結果
data _null_;	
dt = datetime();	+--- NLDATMMN min=4 default=10
dy = date();	max=200 ---+
put "+--- NLDATEMN min=4 default=10	October
max=200 ---+";	Oct
put dt nldatmmn.;	October
put dt nldatmmn4.;	October
put dt nldatmmn10.;	
put dt nldatmmn200.;	
run;	

NLDATMTMw. 出力形式

SAS 日時値の時間部分を、指定されたロケールの時間値に変換し、その時間値から時刻を書き出します。

カテゴリ: 日付と時間**配置:** 左**構文**

NLDATMTMw.

説明**w**

出力フィールドの幅を指定します。

デフォルト: 16**範囲:** 16–200**比較**

NLDATMTMw.出力形式は TODw.出力形式に似ていますが、NLDATMTMw.出力形式がロケール固有である点が違います。

例

これらの例では、1361709583 を入力値として使用します。この値は、2003 年 2 月 24 日の午後 12:39:43 に相当する SAS 日時値です。

ステートメント	結果
	----+----1
options locale=English_UnitedStates; put event nldatmtm.;	12:39:43
options locale=German_Germany; put event nldatmtm.;	12.39 Uhr

関連項目:

出力形式:

- “NLDATMw. 出力形式” (111 ページ)
- “NLDATMAPw. 出力形式” (112 ページ)
- “NLDATMWw. 出力形式” (117 ページ)

NLDATMTZw. 出力形式

ロケールの SAS 日時値の時間部分を、時刻とタイムゾーンに変換します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMTZ w .

説明

w

出力フィールドの幅を指定します。必要に応じて、出力形式の幅に合うように曜日と日時が短縮されます。

デフォルト: 32

範囲: 16–200

例

この例は現在の日時値を使用します。

ステートメント	結果
<pre>options locale=fr_FR; data test; x=datetime(); put x=nldatmtz.; run;</pre>	<pre>x=10 h 40 -0400</pre>

NLDATMWw. 出力形式

SAS 日時値を、曜日および日時形式の、ロケール依存の日時文字列に変換します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMW w .

説明

w

出力フィールドの幅を指定します。必要に応じて、出力形式の幅に合うように曜日と日時が短縮されます。

デフォルト: 30

範囲: 16–200

比較

NLDATMW w 出力形式は、TWMDY w 出力形式に似ていますが、NLDATMW w 出力形式がロケール固有である点が違います。

例

これらの例では、1361709583 を入力値として使用します。この値は、2003 年 2 月 24 日の午後 12:39:43 に相当する SAS 日時値です。

ステートメント	結果
	<pre>----+-----1-----+-----2-----+-----3</pre>
<pre>options locale=English_UnitedStates; data null; x=put(1361709583,nldatmw.); y=put(1361709583,nldatmw30.); z=put(1361709583,nldatmw200.); run;</pre>	<pre>Mon, Feb 24, 2003 12:39:43 PM Mon, Feb 24, 2003 12:39:43 PM Monday, February 24, 2003 12:39:43 PM</pre>

ステートメント	結果
options locale=german_germany;	Mo, 24. Feb 2003 12.39 Uhr
data null;	Mo, 24. Feb 2003 12.39 Uhr
x=put(1361709583,nldatmw.);	Montag, 24. Februar 2003 12.39 Uhr
y=put(1361709583,nldatmw30.);	
z=put(1361709583,nldatmw200.);	
run;	

関連項目:

出力形式:

- “NLDATMw. 出力形式” (111 ページ)
- “NLDATMAPw. 出力形式” (112 ページ)
- “NLDATMTMw. 出力形式” (115 ページ)

NLDATMWNw. 出力形式

SAS 日時値を、指定されたロケールの日時値に変換し、その値から曜日を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMWN_w.

説明

w
出力フィールドの幅を指定します。

デフォルト: 30

範囲: 16–200

例

SAS 日時値から曜日を書き出します。

```
now = datetime() ;
put now nldatmwn. ;
```

NLDATMWZw. 出力形式

指定されたロケールの SAS 日時値を、曜日、日時、およびタイムゾーン値に変換します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMWZw.

説明

w

出力フィールドの幅を指定します。必要に応じて、出力形式の幅に合うように曜日と日時が短縮されます。

デフォルト: 40

範囲: 16–200

例

この例は現在の日時値を使用します。

ステートメント	結果
<pre>options locale=fr_FR; data test; x=datetime(); put x=nldatmwz.; run;</pre>	<pre>x=vendredi 18 mars 2011 10 h 40 -0400</pre>

NLDATMYMw. 出力形式

SAS 日時値を、指定されたロケールの日時値に変換し、その値から年と月の名前を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMYMw.

説明

w

出力フィールドの幅を指定します。

デフォルト: 16

範囲: 6–200

例

この例では、en_US ロケールオプションを使用します。

ステートメント	結果
options locale=en_US;	January 1960
data_null;	January 1960
x=put(86400,nldatmym.);	
y=put(86400,nldatmym12.);	
put x=;	
put y=;	
run;	

関連項目:

出力形式:

- [“NLDATMMDw. 出力形式” \(114 ページ\)](#)

NLDATMYQw. 出力形式

SAS 日時値を、指定されたロケールの日時値に変換し、その値から年と四半期を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMYQ_w.

説明

w 出力フィールドの幅を指定します。

デフォルト: 16

範囲: 4–200

例

この例では、fr_FR ロケールオプションを使用します。

ステートメント	結果
options locale=fr_FR;	
data _null_;	+--- NLDATMYQ min=4 default=16
dy=today();	max=200 ---+
dt=datetime();	16 T3 08
put "+--- NLDATMYQ min=4 default=16	4 ****
max=200 ---+";	14 T3 08
put '16' +5 dt nldatmyq.;	32 3e trimestre 2008
put '4' +5 dt nldatmyq4.;	200 3e trimestre 2008
put '14' +5 dt nldatmyq14.;	
put '32' +5 dt nldatmyq32.;	
put '200' +5 dt nldatmyq200.;	
run;	

NLDATMYRw. 出力形式

SAS 日時値を、指定されたロケールの日時値に変換し、その値から年を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMYR w .

説明

w
出力フィールドの幅を指定します。

デフォルト: 16

範囲: 2–200

例

この例では、en_US ロケールオプションを使用します。

ステートメント	結果
options locale=fr_FR;	
data _null_;	+--- NLDATMYR min=2 default=16
dy=today();	max=200 ---+
dt=datetime();	2008
put "+--- NLDATMYR min=2 default=16	08
max=200 ---+";	2008
put dt nldatmyr.;	2008
put dt nldatmyr2.;	
put dt nldatmyr32.;	
put dt nldatmyr200.;	
run;	

NLDATMYWw. 出力形式

SAS 日時値を、指定されたロケールの日時値に変換し、その値から年と週の名前を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMYWw.

説明

w
出力フィールドの幅を指定します。

デフォルト: 16

範囲: 5–200

例

この例では、fr_FR ロケールオプションを使用します。

ステートメント	結果
options locale=fr_FR;	
data _null_;	+--- NLDATMYW min=5 default=16
dy=today();	max=200 ---+
dt=datetime();	16 Week 33 2008
put "+--- NLDATMYW min=5 default=16	5 *****
max=200 ---+";	8 W33 08
put '16' +5 dt nldatmyw.;	32 Week 33 2008
put '5' +5 dt nldatmyw5.;	200
put '8' +5 dt nldatmyw8.;	Week 33 2008
put '32' +5 dt nldatmyw32.;	
put '200' +5 dt nldatmyw200.;	
run;	

NLDATMZw. 出力形式

SAS 日時値を、タイムゾーンおよび日時形式の、ロケール依存の日時文字列に変換します。

カテゴリ: 日付と時間

配置: 左

構文

NLDATMZ w .

説明

w

出力フィールドの幅を指定します。必要に応じて、出力形式の幅に合うように曜日と日時が短縮されます。

デフォルト: 40

範囲: 16–200

例

この例は現在の日時値を使用します。

ステートメント	結果
options locale=fr_FR;	x=18 mars 2011 10 h 40 -0400
data test;	
x=datetime();	
put x=nldatmz.;	
run;	

NLMNIAEDw.d 出力形式

アラブ首長国連邦の国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIAEDw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d
10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 3

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmniaed32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	(AED1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLAEDx.d 出力形式” \(157 ページ\)](#)

NLMNIAUDw.d 出力形式

オーストラリアの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIAUD $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmniaud32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(AUD1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNLAUDw.d 出力形式” (158 ページ)

NLMNIBGNw.d 出力形式

ブルガリアの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIBGN $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmbnbn32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(BGN1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLBGN \$w.d\$ 出力形式” \(159 ページ\)](#)

NLMNIBRL $w.d$ 出力形式

ブラジルの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIBRL $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789, nlmnibr132.2);
```

```
y=put(-1234.56789, dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(BRL1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLBRLw.d 出力形式” \(160 ページ\)](#)

NLMNICADw.d 出力形式

カナダの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNICADw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmcad32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	(CAD1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLCADw.d 出力形式” \(160 ページ\)](#)

NLMNICHFw.d 出力形式

リヒテンシュタインとスイスの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNICHFw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmmichf32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(CHF1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLCHFw.d 出力形式” \(161 ページ\)](#)

NLMNICNYw.d 出力形式

中国の国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNICNYw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 02

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmmicny32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(CNY1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLCNYw.d 出力形式” \(162 ページ\)](#)

NLMNICZKw.d 出力形式

チェコ共和国の国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNICZKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmiczk32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	(CZK1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNLCZKw.d 出力形式” (163 ページ)

NLMNIDKKw.d 出力形式

デンマーク、フェロー諸島およびグリーンランドの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIDKKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=put (-1234.56789,nlmidkk32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(DKK1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- [“NLMNLDKKw.d 出力形式” \(164 ページ\)](#)

NLMNIEEKw.d 出力形式

エストニアの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIEEK $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlennieek32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(EEK1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- [“NLMNLEEKw.d 出力形式” \(165 ページ\)](#)

NLMNIEGPw.d 出力形式

エジプトの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIEGPw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 3

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmniegp32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(EGP1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLEGPw.d 出力形式” \(166 ページ\)](#)

NLMNIEURw.d 出力形式

ベルギー、フィンランド、フランス、ドイツ、ギリシャ、アイルランド、イタリア、ルクセンブルク、マルタ、オランダ、ポルトガル、スロベニア、スペインの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIEUR $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが Locale=German_Germany に設定されています。

```
x=put(-1234.56789,nlmmieur32.2);
y=put(-1234.56789,nlmmleur32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-1.234,57EUR
put y=;	-1.234,57€

関連項目:

出力形式:

- [“NLMNLEUR \$w.d\$ 出力形式” \(167 ページ\)](#)

NLMNIGBP $w.d$ 出力形式

英国の国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIGBPw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmgbp32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(GBP1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLGBPw.d 出力形式” \(168 ページ\)](#)

NLMNIHKDw.d 出力形式

香港の国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIHKDw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmnihkd32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(HKD1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- “NLMNLHKDw.d 出力形式” (169 ページ)

NLMNIHRKw.d 出力形式

クロアチアの国際表示形式で通貨データを書き出します。

カテゴリ: 数値**配置:** 左**構文**

NLMNIHRKw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmihrk32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	(HRK1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLHRKw.d 出力形式” \(169 ページ\)](#)

NLMNIHUFw.d 出力形式

ハンガリーの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIHUFw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmmihuf32.2);
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(HUF1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLHUFw.d 出力形式” \(170 ページ\)](#)

NLMNIIDRw.d 出力形式

インドネシアの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIIDRw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmmiidr32.2);
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(IDR1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLIDRw.d 出力形式” \(171 ページ\)](#)

NLMNIILSw.d 出力形式

イスラエルの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIILSw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmaniis32.2);
```

```
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	(ILS1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNLILSw.d 出力形式” (172 ページ)

NLMNIINRw.d 出力形式

インドの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIINRw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=put (-1234.56789,nlmiinr32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(INR1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- [“NLMNLINRw.d 出力形式” \(173 ページ\)](#)

NLMNIJPYw.d 出力形式

日本の国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIJPYw.d

説明

w

出カフィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmnijpy32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(JPY1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- [“NLMNLJPYw.d 出力形式” \(174 ページ\)](#)

NLMNIKRWw.d 出力形式

韓国の国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIKRWw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d
 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlkrw32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	(KRW1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLKRWw.d 出力形式” \(175 ページ\)](#)

NLMNILTLw.d 出力形式

リトアニアの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNILTL $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmlt132.2);
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(LTL1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLLTLw.d 出力形式” \(176 ページ\)](#)

NLMNILVLw.d 出力形式

ラトビアの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNILVL $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmlvl32.2);
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(LVL1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLLVL \$w.d\$ 出力形式” \(177 ページ\)](#)

NLMNIMOP $w.d$ 出力形式

マカオの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIMOP $w.d$

説明**w**

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

x=put (-1234.56789,nlmnimop32.2);

y=put (-1234.56789,dollar32.2);

ステートメント	結果
	----+-----1-----+
put x=;	(MOP1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- “NLMNLMOPw.d 出力形式” (178 ページ)

NLMNIMXNw.d 出力形式

メキシコの国際表示形式で通貨データを書き出します。

カテゴリ: 数値**配置:** 左**構文**

NLMNIMXNw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmmnmxn32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	(MXN1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNLMXN*w.d* 出力形式” (178 ページ)

NLMNIMYR*w.d* 出力形式

マレーシアの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIMYR*w.d*

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmmimyr32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(MYR1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLMYRw.d 出力形式” \(179 ページ\)](#)

NLMNINOKw.d 出力形式

ノルウェーの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNINOKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmmiinok32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(NOK1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNLNOKw.d 出力形式” (180 ページ)

NLMNINZDw.d 出力形式

ニュージーランドの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNINZDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlminzd32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	(NZD1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLNZDw.d 出力形式” \(181 ページ\)](#)

NLMNIPLNw.d 出力形式

ポーランドの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIPLNw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=put (-1234.56789,nlmpiLn32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1-----+
put x=;	(PLN1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- [“NLMNLPLNw.d 出力形式” \(182 ページ\)](#)

NLMNIRUBw.d 出力形式

ロシアの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIRUBw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmdirub32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(RUB1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- [“NLMNLRUBw.d 出力形式” \(183 ページ\)](#)

NLMNISEKw.d 出力形式

スウェーデンの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNISEKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmnisek32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(SEK1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLSEKw.d 出力形式” \(184 ページ\)](#)

NLMNISGDw.d 出力形式

シンガポールの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNISGD $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlisgd32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(SGD1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLSGD \$w.d\$ 出力形式” \(185 ページ\)](#)

NLMNITHB $w.d$ 出力形式

タイの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNITHBw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmnithb32.2);
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(THB1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLTHBw.d 出力形式” \(186 ページ\)](#)

NLMNITRYw.d 出力形式

トルコの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNITRYw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 4

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmitry32.2);
```

```
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(TRY1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- [“NLMNLTRYw.d 出力形式” \(187 ページ\)](#)

NLMNITWDw.d 出力形式

台湾の国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNITWDw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmitwd32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1-----+
put x=;	(TWD1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLTWDw.d 出力形式” \(187 ページ\)](#)

NLMNIUSDw.d 出力形式

プエルトリコと米国の国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIUSDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 912

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmiusd32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(USD1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLUSDw.d 出力形式” \(188 ページ\)](#)

NLMNIZARw.d 出力形式

南アフリカの国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNIZARw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmiuzar32.2);
y=put (-1234.56789,dollar32.2);
```


ステートメント	結果
	----+-----1-----+
put x=;	(ZAR1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNLZARw.d 出力形式” (189 ページ)

NLMNLAEDx.d 出力形式

アラブ首長国連邦のローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLAEDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 3

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmnlaed32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	(AED1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNIAEDw.d 出力形式” (124 ページ)

NLMNLAUDw.d 出力形式

オーストラリアのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLAUDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=put (-1234.56789,nlmlaud32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(AU\$1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- [“NLMNIAUDw.d 出力形式” \(124 ページ\)](#)

NLMNLBGNw.d 出力形式

ブルガリアのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLBGNw.d

説明

w

出カフィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlbgn32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(BGN1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- [“NLMNIBGNw.d 出力形式” \(125 ページ\)](#)

NLMNLBRLw.d 出力形式

ブラジルのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLBRLw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d
必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlnbrl32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(R\$1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIBRLw.d 出力形式” \(126 ページ\)](#)

NLMNLCADw.d 出力形式

カナダのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLCAD $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlcad32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(CA\$1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNICADw.d 出力形式” \(127 ページ\)](#)

NLMNLCHFw.d 出力形式

リヒテンシュタインとスイスのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLCHF $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlchf32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	SFr.1,234.57
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNICHF $w.d$ 出力形式” (128 ページ)

NLMNLCNY $w.d$ 出力形式

中国のローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLCNY $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlncy32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(RMB1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNICNYw.d 出力形式” (129 ページ)

NLMNLCZKw.d 出力形式

チェコ共和国のローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLCZKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlnczk32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	(CZK1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNICZKw.d 出力形式” (130 ページ)

NLMNLDKKw.d 出力形式

デンマーク、フェロー諸島およびグリーンランドのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLDKKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=put (-1234.56789,nlmdlkk32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(kr1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIDKKw.d 出力形式” \(131 ページ\)](#)

NLMNLEEKw.d 出力形式

エストニアのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLEEKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=put (-1234.56789,nlmlleek32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(Kr1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “[NLMNIEEKw.d 出力形式](#)” (132 ページ)

NLMNLEGPw.d 出力形式

エジプトのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLEGPw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 3

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmnlegp32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	(EGP1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNIEGPw.d 出力形式” (133 ページ)

NLMNLEURw.d 出力形式

オーストリア、ベルギー、フィンランド、フランス、ドイツ、ギリシャ、アイルランド、イタリア、ルクセンブルク、マルタ、オランダ、ポルトガル、スロベニア、スペインのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLEURw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが German_Germany に設定されていません。

```
x=put (-1234.56789,nlmmieur32.2);
y=put (-1234.56789,nlmmieur32.2);
```

ステートメント	結果
	----+----1----+----2----+
put x=;	-1.234,57EUR
put y=;	€-1.234,57

関連項目:**出力形式:**

- “NLMNIEURw.d 出力形式” (133 ページ)

NLMNLGBPw.d 出力形式

英国のローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLGBPw.d

説明

w

出カフィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlgbp32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(£1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- “NLMNIGBPw.d 出力形式” (134 ページ)

NLMNLHKDw.d 出力形式

香港のローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLHKDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlhkd32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	(HK\$1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIHKDw.d 出力形式” \(135 ページ\)](#)

NLMNLHRKw.d 出力形式

クロアチアのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLHRK $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlhrk32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(Kn1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIHRK \$w.d\$ 出力形式” \(136 ページ\)](#)

NLMNLHUF $w.d$ 出力形式

ハンガリーのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLHUFw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlhuf32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(Ft1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIHUFw.d 出力形式” \(137 ページ\)](#)

NLMNLIDRw.d 出力形式

インドネシアのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLIDRw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmlidr32.2);
```

```
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(Rp1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- “NLMNIIDRw.d 出力形式” (138 ページ)

NLMNLILSw.d 出力形式

イスラエルのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLILSw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlnls32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	(ILS1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIIISw.d 出力形式” \(139 ページ\)](#)

NLMNLINRw.d 出力形式

インドのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLINRw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmlinr32.2);
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(INR1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIINRw.d 出力形式” \(140 ページ\)](#)

NLMNLJPYw.d 出力形式

日本の国際表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLJPYw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmljpy32.2);
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(JPY1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIJPYw.d 出力形式” \(141 ページ\)](#)

NLMNLKRWw.d 出力形式

韓国のローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLKRWw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=put (-1234.56789,nlmlkrw32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	(KRW1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNKRWw.d 出力形式” (142 ページ)

NLMNLLTLw.d 出力形式

リトアニアのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLLTLw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=put (-1234.56789,nlmlt132.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(LT1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- “NLMNLTW.d 出力形式” (142 ページ)

NLMNLLVLw.d 出力形式

ラトビアのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLLVLw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlvlv132.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(Ls1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- “NLMNLLVLw.d 出力形式” (143 ページ)

NLMNLMOPw.d 出力形式

マカオのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLMOPw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d
10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmnlmop32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(P1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIMOPw.d 出力形式” \(144 ページ\)](#)

NLMNLMXNw.d 出力形式

メキシコのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLMXNw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlmxn32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(MX\$1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIMXNw.d 出力形式” \(145 ページ\)](#)

NLMNLMYRw.d 出力形式

マレーシアのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLMYR*w.d*

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlmyr32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(R1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNIMYR*w.d* 出力形式” (146 ページ)

NLMNLNOK*w.d* 出力形式

ノルウェーのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLNOK*w.d*

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlnok32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(kr1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNINOKw.d 出力形式” \(147 ページ\)](#)

NLMNLNZDw.d 出力形式

ニュージーランドのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLNZDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlnzd32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----
put x=;	(NZ\$1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNINZD*w.d* 出力形式” (148 ページ)

NLMNLPLN*w.d* 出力形式

ポーランドのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLPLN*w.d*

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlpln32.2);
y=put (-1234.56789,dollar32.2)
```

ステートメント	結果
	----+----1-----+
put x=;	(PLN1,234.57
put y=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNIPLNw.d 出力形式” (149 ページ)

NLMNLRUBw.d 出力形式

ロシアのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLRUBw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlrub32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(RUB1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- “[NLMNIRUBw.d 出力形式](#)” (150 ページ)

NLMNLSEKw.d 出力形式

スウェーデンのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLSEKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmnlsek32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	(kr1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNISEKw.d 出力形式” \(151 ページ\)](#)

NLMNLSGDw.d 出力形式

シンガポールのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLSGDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=put (-1234.56789,nlmlsgd32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(SG\$1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- “[NLMNISGDw.d 出力形式](#)” (151 ページ)

NLMNLTHBw.d 出力形式

タイのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLTHBw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlnthb32.2);
```

```
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(THB1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- “[NLMNITHBw.d 出力形式](#)” (152 ページ)

NLMNLTTRYw.d 出力形式

トルコのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLTTRYw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d
 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 4

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlntry32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	(YTL1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNLTTRYw.d 出力形式” \(153 ページ\)](#)

NLMNLTWDw.d 出力形式

台湾のローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLTWD $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 2

範囲: 0–28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmltwd32.2);
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(NT\$1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNITWD \$w.d\$ 出力形式” \(154 ページ\)](#)

NLMNLTUSD $w.d$ 出力形式

プエルトリコと米国のローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLUSDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmlusd32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(US\$1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNIUSDw.d 出力形式” \(155 ページ\)](#)

NLMNLZARw.d 出力形式

南アフリカのローカル表示形式で通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNLZARw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 12

範囲: 8-32

d10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 2

範囲: 0-28

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put(-1234.56789,nlmlzar32.2);
```

```
y=put(-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(R1,234.57)
put y=;	\$-1,234.57

関連項目:**出力形式:**

- “NLMNIZARw.d 出力形式” (156 ページ)

NLMNYw.d 出力形式

ローカル通貨を使用して、指定されたロケールに定義されたローカル表示形式で通貨データを書き出します。

カテゴリ: 数値**配置:** 左**構文**

NLMNYw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

数値の小数点以下の桁数を指定します。

デフォルト: 0

範囲: 0-31

詳細

NLMNY $w.d$ 出力形式では、2 の補数で表される負の値を含む、バイナリ整数(固定小数点)値を読み込みます。NLMNY $w.d$ 出力形式では、そのロケールでの通貨記号、桁区切り文字および小数点区切り文字を使用して数値を書き出します。

注: NLMNY $w.d$ 出力形式では通貨データを変換しないため、出力形式が適用された数値は現在のロケールの通貨値と等しくなります。

比較

NLMNY $w.d$ 出力形式と NLMNYI $w.d$ 出力形式では、ロケールに依存する桁区切り文字と小数点区切り文字を使用して通貨データを書き出します。ただし、NLMNYI $w.d$ 出力形式では USD などの 3 文字の国際通貨コードを使用するのに対し、NLMNY $w.d$ 出力形式では \$ などの地域の通貨記号を使用します。

NLMNY $w.d$ 出力形式は DOLLAR $w.d$ 出力形式と似ていますが、NLMNY $w.d$ 出力形式がロケール固有である点が違います。

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=put (-1234.56789,nlmy32.2);
y=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	(\$1,234.57)
put y=;	\$-1,234.57

関連項目:

出力形式:

- [“NLMNYIw.d 出力形式” \(191 ページ\)](#)

入力形式:

- [“NLMNYw.d 入力形式” \(408 ページ\)](#)
- [“NLMNYIw.d 入力形式” \(410 ページ\)](#)

NLMNYlw.d 出力形式

指定されたロケールに定義された国際表示形式を使用して通貨データを書き出します。

カテゴリ: 数値

配置: 左

構文

NLMNYIw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

数値の小数点以下の桁数を指定します。

デフォルト: 0

範囲: 0–31

詳細

NLMNYIw.d 出力形式では、2 の補数で表される負の値を含む、バイナリ整数(固定小数点)値を読み込みます。NLMNYIw.d 出力形式では、国際通貨コード、およびロケールに依存する桁区切り文字と小数点区切り文字を使用して数値を書き出します。国際通貨コードの位置もロケールに応じて決まります。

注: NLMNYIw.d 出力形式では通貨データを変換しないため、出力形式が適用された数値は現在のロケールの通貨値と等しくなります。

比較

NLMNYw.d 出力形式と NLMNYIw.d 出力形式では、ロケールに依存する桁区切り文字と小数点区切り文字を使用して通貨データを書き出します。ただし、NLMNYIw.d 出力形式では USD などの 3 文字の国際通貨コードを使用するのに対し、NLMNYw.d 出力形式では \$ などの地域の通貨記号を使用します。

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-1234.56789,nlmnyi32.2);
y=put (-1234.56789,nlmny32.2);
z=put (-1234.56789,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	(USD1,234.57)
put y=;	(\$1,234.57)
put z=;	\$-1,234.57

関連項目:

出力形式:

- “NLMNYw.d 出力形式” (190 ページ)

入力形式:

- “NLMNYw.d 入力形式” (408 ページ)
- “NLMNYIw.d 入力形式” (410 ページ)

NLNUMw.d 出力形式

指定されたロケールに定義されたローカル表示形式で数値を書き出します。

カテゴリ: 数値

配置: 左

構文

NLNUMw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点区切り文字が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

詳細

NLMUMw.d 出力形式では、2 の補数で表される負の値を含む、バイナリ整数(固定小数点)値を読み込みます。NLNUMw.d 出力形式では、そのロケールでの桁区切り文字および小数点区切り文字を使用して数値を書き出します。

比較

NLNUMw.d 出力形式では、ロケールに依存する桁区切り文字と小数点区切り文字を使用して数値を書き出します。NLNUMIw.d 出力形式では、桁区切り文字としてカンマ(,)を、小数点区切り文字としてピリオド(.)を使用して数値を書き出します。

w 値または *d* 値が、出力形式が適用された数値を生成するほど十分に大きくない場合、NLNUMw.d 出力形式では、小数精度が若干失われるとしても、可能な限り桁区切り文字を出力するアルゴリズムを使用します。

例

```
x=put(-1234356.7891, nlnum32.2);
```

ステートメント	結果
	----+----1----
options LOCALE=English_UnitedStates; put x=;	-1,234,356.79
options LOCALE=German_Germany; put x=;	-1.234.356,79

関連項目:**出力形式:**

- “NLNUMIw.d 出力形式” (194 ページ)

入力形式:

- “NLNUMw.d 入力形式” (411 ページ)
- “NLNUMIw.d 入力形式” (412 ページ)

NLNUMIw.d 出力形式

指定されたロケールに定義された国際表示形式で数値を書き出します。

カテゴリ: 数値

配置: 左

構文

NLNUMIw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

範囲: 1–32

d

10^{*d*} で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

詳細

NLNUMIw.d 出力形式では、2 の補数で表される負の値を含む、バイナリ整数(固定小数点)値を読み込みます。NLNUMIw.d 出力形式では、すべてのロケールに対し、桁区切り文字としてカンマ(,)を、また小数点区切り文字としてピリオド(.)を使用して数値を書き出します。

比較

NLNUMIw.d 出力形式では、指定されたロケールに定義された国際表示形式で数値データを書き出します。NLNUMIw.d 出力形式では、桁区切り文字としてカンマ(,)を、小数点区切り文字としてピリオド(.)を使用して数値を書き出します。

w 値または d 値が、出力形式が適用された数値を生成するほど十分に大きくない場合、NLNUMw.d 出力形式では、小数精度が若干失われるとしても、可能な限り桁区切り文字を出力するアルゴリズムを使用します。

例

```
x=put(-1234356.7891,nlnumi32.2);
```

ステートメント	結果
	----+-----1-----+
options LOCALE=English_UnitedStates; put x=;	-1,234,356.79
options LOCALE=German_Germany; put x=;	-1,234,356.79

関連項目:

出力形式:

- [“NLNUMw.d 出力形式” \(193 ページ\)](#)

入力形式:

- [“NLNUMw.d 入力形式” \(411 ページ\)](#)
- [“NLNUMIw.d 入力形式” \(412 ページ\)](#)

NLPCTw.d 出力形式

指定されたロケールに定義されたローカル表示形式で百分率のデータを書き出します。

カテゴリ: 数値

配置: 左

構文

NLPCTw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

範囲: 4-32

d

10^d で数値を除算するよう指定します。データに小数点区切り文字が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0-31

比較

NLPCTw.d 出力形式では、指定されたロケールに定義されたローカル表示形式で百分率のデータを書き出します。NLPCTw.d 出力形式では、ロケールに依存する桁区切り文字と小数点区切り文字を使用して百分率値を書き出します。NLPCTIw.d 出力形式では、桁区切り文字としてカンマ(,)を、また小数点区切り文字としてピリオド(.)を使用して百分率値を書き出します。

NLPCTw.d 出力形式は PERCENTw.d 出力形式に似ていますが、NLPCTw.d 出力形式がロケール固有である点が違います。

例

```
x=put(-12.3456789,nlpct32.2);
y=put(-12.3456789,nlpcti32.2);
z=put(-12.3456789,percent32.2);
```

ステートメント	結果
	----+-----1
options LOCALE=English_UnitedStates;	-1,234.57%
put x=;	-1,234.57%
put y=;	(1234.57%)
put z=;	
options LOCALE=German_Germany;	-1.234,57%
put x=;	-1,234.57%
put y=;	(1234.57%)
put z=;	

関連項目:

出力形式:

- “NLPCTIw.d 出力形式” (197 ページ)

入力形式:

- “NLPCTw.d 入力形式” (413 ページ)
- “NLPCTw.d 入力形式” (414 ページ)

NLPCTw.d 出力形式

指定されたロケールに定義された国際表示形式で百分率のデータを書き出します。

カテゴリ: 数値

配置: 左

構文

NLPCTw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

範囲: 4–32

d

10^d で数値を除算するよう指定します。データに小数点区切り文字が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

比較

NLPCTw.d 出力形式では、指定されたロケールに定義された国際表示形式で百分率のデータを書き出します。NLPCTw.d 出力形式では、ロケールに依存する桁区切り文字と小数点区切り文字を使用して百分率値を書き出します。NLPCTw.d 出力形式では、桁区切り文字としてカンマ(,)を、また小数点区切り文字としてピリオド(.)を使用して百分率値を書き出します。

NLPCTw.d 出力形式は PERCENTw.d 出力形式に似ていますが、NLPCTw.d 出力形式がロケール固有である点が違います。

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (-12.3456789,nlpcti32.2);
y=put (-12.3456789,percent32.2);
```

ステートメント

結果

-----1

ステートメント	結果
put x=;	-1,234.57%
put y=;	(1234.57)

関連項目:

出力形式:

- “NLPCTw.d 出力形式” (195 ページ)

入力形式:

- “NLPCTw.d 入力形式” (413 ページ)
- “NLPCTw.d 入力形式” (414 ページ)

NLPCTNw.d 出力形式

百分率を書き出します。負の値にはマイナス記号を使用します。

カテゴリ: 数値

配置: 右

構文

NLPCTN w . d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

範囲: 4-32

ヒント: 出力フィールドの幅数には、数値が負か正かにかかわらず、マイナス記号(-)、パーセント記号(%), 末尾の空白分を含める必要があります。

d

数値の小数点以下の桁数を指定します。この引数はオプションです。

範囲: 0-31

要件 w より小さい数にします。

詳細

NLPCTN w . d 出力形式は、100 を乗算した負の値の先頭にマイナス記号を追加し、出力形式を適用した値の最後にパーセント記号(%)を追加します。

例

```
x=-0.02;
```

ステートメント	結果
put x nlpctn6.;	x=-2%
put x percentn6.;	x=-2%

NLPCTPw.d 出力形式

ロケール固有の数値を百分率として書き出します。

カテゴリ: 数値

配置: 右

構文

NLPCTPw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

範囲: 4-32

ヒント: 出力フィールドの幅数には、パーセント記号(%)分を含める必要があります。

d

数値の小数点以下の桁数を指定します。この引数はオプションです。NLPCTP 出力形式の桁区切り文字と小数点の記号は、ロケールに固有です。

範囲: 0-31

要件 *w* より小さい数にします。

詳細

NLPCTPw.d 出力形式は、100 を乗算して出力形式を適用した値の最後にパーセント記号(%)を追加します。NLPCTPw.d 出力形式は PERCENTw.d 出力形式に似ていますが、NLPCTPw.d 出力形式がロケール固有である点が違います。

例

```
x=-0.02;
```

ステートメント	結果
put x nlpctp6.;	-2%
put x percent6.;	(2%)

NLPVALUE $w.d$ 出力形式

指定されたロケールのローカル表示形式で p 値を書き出します。

カテゴリ: 数値

配置: 左

構文

NLPVALUE $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

範囲: 3–32

d

10^d で数値を除算するよう指定します。データに小数点区切り文字が含まれる場合、 d 値は無視されます。

デフォルト: 4

範囲: 1–30

例

この例では、`german_Germany` ロケールオプションを使用します。

ステートメント:

```
options locale=german_germany;
data _null_;
put "+--- nlpvalue min=3 default=6 max=32 ---+";
x=0.1248;
put x= +5 x pvalue. +5 x nlpvalue.;
put x= +5 x pvalue3.1 +5 x nlpvalue3.1;
put x= +5 x pvalue20.2 +5 x nlpvalue20.2;
put x= +5 x pvalue32.3 +5 x nlpvalue32.3;
run;
```

結果:

```
+--- nlpvalue min=3 default=6 max=32 ---+
x=0.1248 0.1248 0,1248
x=0.1248 0.1 0,1
x=0.1248 0.12 0,12
x=0.1248 0.125 0,125
```

関連項目:

出力形式:

- “PVALUE $w.d$ Format” in *SAS Formats and Informats: Reference*

NLSTRMONw.d 出力形式

指定されたロケールの月名を書き出します。

カテゴリ: 数値

配置: 左

構文

NLSTRMONw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 20

範囲: 200-1

d

次の項目を指定します。

- 00000001: 短縮形式で書き出します。
- 00000010: 大文字形式で書き出します。

デフォルト: 0

範囲: 0-3

詳細

NLSTRMONw.d 出力形式では、指定されたロケールの月名として 1 から 12 の SAS 値を書き出します。次の例では、English_UnitedStates ロケールを使用します。

- 1 = 第 1 月(January)
- 2 = 第 2 月(February)
- 3 = 第 3 月(March)
- 4 = 第 4 月(April)
- 5 = 第 5 月(May)
- 6 = 第 6 月(June)
- 7 = 第 7 月(July)
- 8 = 第 8 月(August)
- 9 = 第 9 月(September)
- 10 = 第 10 月(October)
- 11 = 第 11 月(November)
- 12 = 第 12 月(December)

例

この例では、English_UnitedSates セッションエンコーディングを使用します。

ステートメント	結果
Data _null_ ;	
monnum = 1 ; /* January=1, December=12 */	January
put monnum NLSTRMON20. ;	Jan
put monnum NLSTRMON20.1; /* decimal .1	JANUARY
specified use abbreviation. */	JAN
put monnum NLSTRMON20.2;	
put monnum NLSTRMON20.3;	
run;	

NLSTRQTRw.d 出力形式

指定されたロケールの四半期の形式で数値を書き出します。

カテゴリ: 数値

配置: 左

構文

NLSTRQTRw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 20

範囲: 1-200

d

次の項目を指定します。

- 00000001: 短縮形式で書き出します。
- 00000010: 大文字形式で書き出します。

デフォルト: 3

範囲: 0-3

詳細

NLSTRQTRw.d 出力形式では、指定されたロケールの四半期名として 1 から 4 の SAS 値を書き出します。次の例では、English_UnitedStates ロケールを使用します。

- 1 = 1st quarter
- 2 = 2nd quarter
- 3 = 3rd quarter

- 4 = 4th quarter

例

この例では、English_UnitedStates セッションエンコーディングを使用します。

ステートメント	結果
Data _null_ ;	
qtrnum = 1 ; /* January=1, December=12 */	1st quarter
put qtrnum NLSTRQTR20. ;	1st quarter
put qtrnum NLSTRQTR20.1; /* decimal .1	1ST QUARTER
specified use abbreviation. */	1ST QUARTER
put qtrnum NLSTRQTR20.2;	
put qtrnum NLSTRQTR20.3; run;	

NLSTRWKw.d 出力形式

指定されたロケールの曜日の形式で数値を書き出します。

カテゴリ: 数値

配置: 左

構文

NLSTRWKw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 20

範囲: 1–200

d
次の項目を指定します。

- 00000001: 短縮形式で書き出します。
- 00000010: 大文字形式で書き出します。

デフォルト: 0

範囲: 0–3

詳細

NLSTQTRw.d 出力形式では、指定されたロケールの曜日名として 1 から 7 の SAS 値を書き出します。次の例では、English_UnitedStates ロケールを使用します。

- 1 = 週の 1 日目 (Monday)
- 2 = 週の 2 日目 (Tuesday)

- 3 = 週の 3 日目 (Wednesday)
- 4 = 週の 4 日目 (Thursday)
- 5 = 週の 5 日目 (Friday)
- 6 = 週の 6 日目 (Saturday)
- 7 = 週の 7 日目 (Sunday)

例

この例では、English_UnitedSates セッションエンコーディングを使用します。

ステートメント	結果
Data _null_ ;	
wknum = 1 ; /* Sunday=1, Saturday=7 */	Sunday
put wknum NLSTRWK20. ;	Sun
put wknum NLSTRWK20.1; /* decimal .1 specified use abbreviation. */	SUNDAY
put wknum NLSTRWK20.2;	SUN
put wknum NLSTRWK20.3;	
run;	

NLTIMAP_w. 出力形式

SAS 時間値を、指定されたロケールの時間値に変換し、その時間値から午前または午後付きの時間値を書き出します。NLTIMAP は、SAS 日時値も変換します。

カテゴリ: 日付と時間

配置: 左

構文

NLTIMAP_w.

説明

w

出力フィールドの幅を指定します。

デフォルト: 10

範囲: 4–200

比較

NLTIMAP_w 出力形式は、TIMEAMP_w 出力形式と似ていますが、NLTIMAP_w 出力形式がロケール固有である点が違います。

例

これらの例では、59083 を入力値として使用します。この値は、午後 04:24:43 に相当する SAS 日時値です。

ステートメント	結果
	----+-----1-----+
options locale=English_UnitedStates; put time nltimap.;	4:24:43 PM
options locale=German_Germany; put time nltimap.;	16.24 Uhr

関連項目:

出力形式:

- [“NLTIMEw. 出力形式” \(205 ページ\)](#)

NLTIMEw. 出力形式

SAS 時間値を、指定されたロケールの時間値に変換し、その時間値から時刻を書き出します。NLTIME は、SAS 日時値も変換します。

カテゴリ: 日付と時間

配置: 左

構文

NLTIMEw.

説明

w 入力幅を指定します。

デフォルト: 20

範囲: 10–200

比較

NLTIMEw.出力形式は、TIMEw.出力形式に似ていますが、NLTIMEw.出力形式はロケール固有である点が違います。

例

これらの例では、59083 を入力値として使用します。この値は、午後 04:24:43 に相当する SAS 日時値です。

ステートメント	結果
	----+-----1-----+
options locale=English_UnitedStates; put time nlttime.;	4:24:43
options locale=German_Germany; put time nlttime.;	16.24

関連項目:

出力形式:

- [“NLTIMAPw. 出力形式” \(204 ページ\)](#)

\$UCS2Bw. 出力形式

現在の SAS セッションのエンコーディングを使用する文字列を処理し、ビッグエンディアン、16 ビット、UCS2 形式の UNICODE (ユニコード)文字列を書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS2B^w.

説明

^w

出力フィールドの幅を指定します。16 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 2–32767

詳細

\$UCS2B^w.出力形式では、ビッグエンディアン、16 ビット、UCS2 (universal character set code in two octets)形式の UNICODE (ユニコード)文字列を書き出します。現在の SAS セッションのエンコーディングを使用する文字列を処理します。

比較

\$UCS2B^w.出力形式では、\$UCS2BE^w.出力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1
<pre>data_null; x = '太'; y=put(x, \$ucs2b2.); put y \$hex.; run;</pre>	5927

関連項目:

出力形式:

- “\$UCS2Lw. 出力形式” (208 ページ)
- “\$UCS2Xw. 出力形式” (211 ページ)
- “\$UTF8Xw. 出力形式” (226 ページ)
- “\$UCS2BEw. 出力形式” (207 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2BEw. 入力形式” (420 ページ)
- “\$UCS2Lw. 入力形式” (421 ページ)
- “\$UCS2Xw. 入力形式” (423 ページ)
- “\$UTF8Xw. 入力形式” (436 ページ)

\$UCS2BEw. 出力形式

ビッグエンディアン、16 ビット、UCS2 形式の UNICODE (ユニコード)文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS2BEw.

説明

w

出力フィールドの幅を指定します。16 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

\$UCS2BEw.出力形式では、現在の SAS セッションのエンコーディングで文字列を書き出します。ビッグエンディアン、16 ビット、UCS2 (universal character set code in two octets)形式の UNICODE (ユニコード)文字列を処理します。

比較

\$UCS2BEw.出力形式では、\$UCS2BEw.出力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+----1
<pre>x = '592700410042'x; put x \$ucs2be.;</pre>	大 AB

関連項目:

出力形式:

- “\$UCS2Bw. 出力形式” (206 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2BEw. 入力形式” (420 ページ)

\$UCS2Lw. 出力形式

現在の SAS セッションのエンコーディングを使用する文字列を処理し、リトルエンディアン、16 ビット、UCS2 形式の UNICODE (ユニコード)文字列を書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS2Lw.

説明

w

出力フィールドの幅を指定します。16 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 2-32767

詳細

\$UCS2Lw.出力形式では、リトルエンディアン、16ビット、UCS2 (universal character set code in two octets)形式の UNICOD (ユニコード)文字列を書き出します。現在の SAS セッションのエンコーディングを使用する文字列を処理します。

比較

\$UCS2Lw.出力形式では、\$UCS2LEw.出力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1
<pre>data_null; x = '大'; y=put(x,\$ucs2l2.); put y \$hex.; run;</pre>	2759

関連項目:

出力形式:

- “\$UCS2Bw. 出力形式” (206 ページ)
- “\$UCS2LEw. 出力形式” (209 ページ)
- “\$UCS2Xw. 出力形式” (211 ページ)
- “\$UTF8Xw. 出力形式” (226 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2Lw. 入力形式” (421 ページ)
- “\$UCS2LEw. 入力形式” (422 ページ)
- “\$UCS2Xw. 入力形式” (423 ページ)
- “\$UTF8Xw. 入力形式” (436 ページ)

\$UCS2LEw. 出力形式

リトルエンディアン、16ビット、UCS2 形式の UNICOD (ユニコード)文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。

カテゴリ: 文字

配置: 左

構文

`$UCS2LEw.`

説明

`w`

出力フィールドの幅を指定します。16ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

`$UCS2LEw.`出力形式では、現在の SAS セッションのエンコーディングで文字列を書き出します。リトルエンディアン、16ビット、UCS2 (universal character set code in two octets)形式の UNICODE (ユニコード)文字列を処理します。

比較

`$UCS2LEw.`出力形式では、`$UCS2Lw.`出力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1
<code>x ='275941004200'x;</code> <code>put x \$ucs2le.;</code>	大 AB

関連項目:

出力形式:

- “[\\$UCS2Lw. 出力形式](#)” (208 ページ)

入力形式:

- “[\\$UCS2Lw. 入力形式](#)” (421 ページ)
- “[\\$UCS2LEw. 入力形式](#)” (422 ページ)

\$UCS2Xw. 出力形式

現在の SAS セッションのエンコーディングを使用する文字列を処理し、ネイティブエンディアン、16 ビット、UCS2 形式の UNICODE (ユニコード)文字列を書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS2Xw.

説明

w
出力フィールドの幅を指定します。16 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 2–32767

詳細

\$UCS2Xw.出力形式では、動作環境のバイトオーダー、16 ビットの UCS2 (universal character set code in two octets)形式の UNICODE (ユニコード)文字列を書き出します。

比較

\$UCS2Xw.出力形式では、\$UCS2XEw.出力形式とは逆の処理を行います。同一の動作環境内のデータを変換する場合は、\$UCS2Xw.出力形式を使用します。異なる動作環境でデータを変換する場合は、\$UCS2Bw.出力形式または\$UCS2Lw.出力形式を使用します。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)セッションエンコーディングを使用します。

ステートメント	結果
	----+----1
x = '大'; put x \$ucs2x2.;	'5927'x (binary) or '2759'x (little endian)

関連項目:

出力形式:

- “\$UCS2Bw. 出力形式” (206 ページ)

- “\$UCS2XEw. 出力形式” (212 ページ)
- “\$UCS2Lw. 出力形式” (208 ページ)
- “\$UTF8Xw. 出力形式” (226 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2Lw. 入力形式” (421 ページ)
- “\$UCS2Xw. 入力形式” (423 ページ)
- “\$UCS2XEw. 入力形式” (424 ページ)
- “\$UTF8Xw. 入力形式” (436 ページ)

\$UCS2XEw. 出力形式

ネイティブエンディアン、16ビット、UCS2形式のUNICODE(ユニコード)文字列を処理し、その文字列を現在のSASセッションのエンコーディングで書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS2XEw.

説明

w

出力フィールドの幅を指定します。16ビットサイズのUNICODE文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

\$UCS2XEw.出力形式では、現在のSASセッションのエンコーディングで文字列を書き出します。ネイティブエンディアン、16ビット、UCS2(universal character set code in two octets)形式のUNICODE(ユニコード)文字列を処理します。

比較

\$UCS2XEw.出力形式では、\$UCS2XEw.出力形式とは逆の処理を行います。

例

この例では、UNIX動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	-----1
<pre>x ='e5a4a7'x; /* Japanese '大 大' ' in UTF8 */; put x \$utf8xe10.;</pre>	

関連項目:

出力形式:

- “\$UCS2Xw. 出力形式” (211 ページ)

入力形式:

- “\$UCS2Xw. 入力形式” (423 ページ)
- “\$UCS2XEw. 入力形式” (424 ページ)

\$UCS4Bw. 出力形式

現在の SAS セッションのエンコーディングを使用する文字列を処理し、ビッグエンディアン、32 ビット、UCS4 形式の UNICODE (ユニコード)文字列を書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS4Bw.

説明

w

出力フィールドの幅を指定します。32 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 4

範囲: 4–32767

詳細

\$UCS4Bw.出力形式では、ビッグエンディアン、32 ビット、UCS4 (universal character set code in four octets)形式の UNICODE (ユニコード)文字列を書き出します。現在の SAS セッションのエンコーディングを使用する文字列を処理します。

比較

\$UCS4Bw.出力形式では、\$UCS4BEw.出力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	-----1
<pre>x = '大'; put x \$ucs4b4.;</pre>	'00005927'x (バイナリ)

関連項目:

出力形式:

- “\$UCS2Lw. 出力形式” (208 ページ)
- “\$UCS2Xw. 出力形式” (211 ページ)
- “\$UCS4BEw. 出力形式” (214 ページ)
- “\$UCS4Lw. 出力形式” (215 ページ)
- “\$UCS4Xw. 出力形式” (218 ページ)
- “\$UTF8Xw. 出力形式” (226 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2Lw. 入力形式” (421 ページ)
- “\$UCS2Xw. 入力形式” (423 ページ)
- “\$UCS4Bw. 入力形式” (425 ページ)
- “\$UCS4Lw. 入力形式” (426 ページ)
- “\$UCS4Xw. 入力形式” (427 ページ)
- “\$UTF8Xw. 入力形式” (436 ページ)

\$UCS4BEw. 出力形式

ビッグエンディアン、32ビット、UCS4形式のUNICODE(ユニコード)文字列を処理し、その文字列を現在のSASセッションのエンコーディングで書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS4BEw.

説明

w

出力フィールドの幅を指定します。32 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

\$UCS4BEw.出力形式では、現在の SAS セッションのエンコーディングで文字列を書き出します。ビッグエンディアン、32 ビット、UCS4 (universal character set code in four octets)形式の UNICODE (ユニコード)文字列を処理します。

比較

\$UCS4BEw.出力形式では、\$UCS4BEw.出力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1
x = '000059270000004100000042'x; put x \$ucs4be.;	大 AB

関連項目:

出力形式:

- “\$UCS4Bw. 出力形式” (213 ページ)

入力形式:

- “\$UCS4Bw. 入力形式” (425 ページ)

\$UCS4Lw. 出力形式

現在の SAS セッションのエンコーディングを使用する文字列を処理し、リトルエンディアン、32 ビット、UCS4 形式の UNICODE (ユニコード)文字列を書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS4Lw.

説明

w

出力フィールドの幅を指定します。32ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 4

範囲: 4–32767

詳細

`$UCS4Lw`出力形式では、リトルエンディアン、32ビット、UCS4 (universal character set code in four octets)形式の UNICODE (ユニコード)文字列を書き出します。現在の SAS セッションのエンコーディングを使用する文字列を処理します。

比較

`$UCS4Lw`出力形式では、`$UCS4LEw`出力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1
<code>data_null;</code>	2759
<code>x = '大';</code>	
<code>y=put(x,\$ucs4l4.);</code>	
<code>put y \$hex.;</code>	
<code>run;</code>	

関連項目:

出力形式:

- “`$UCS2Bw` 出力形式” (206 ページ)
- “`$UCS2Xw` 出力形式” (211 ページ)
- “`$UCS4Bw` 出力形式” (213 ページ)
- “`$UCS4LEw` 出力形式” (217 ページ)
- “`$UCS4Xw` 出力形式” (218 ページ)
- “`$UTF8Xw` 出力形式” (226 ページ)

入力形式:

- “`$UCS2Bw` 入力形式” (419 ページ)
- “`$UCS2Lw` 入力形式” (421 ページ)
- “`$UCS2Xw` 入力形式” (423 ページ)

- “\$UCS4Bw. 入力形式” (425 ページ)
- “\$UCS4Lw. 入力形式” (426 ページ)
- “\$UCS4Xw. 入力形式” (427 ページ)
- “\$UTF8Xw. 入力形式” (436 ページ)

\$UCS4LEw. 出力形式

リトルエンディアン、32 ビット、UCS4 形式の UNICODE (ユニコード)文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS4LEw.

説明

`w`

出力フィールドの幅を指定します。32 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

\$UCS4LEw.出力形式では、現在の SAS セッションのエンコーディングで文字列を書き出します。リトルエンディアン、32 ビット、UCS4 (universal character set code in four octets)形式の UNICODE (ユニコード)文字列を処理します。

比較

\$UCS4LEw.出力形式では、\$UCS4LEw.出力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1
<pre>x = '275900004100000042000000'x; put x \$ucs4le.;</pre>	大 AB

関連項目:**出力形式:**

- “\$UCS4Lw. 出力形式” (215 ページ)

入力形式:

- “\$UCS4Lw. 入力形式” (426 ページ)

\$UCS4Xw. 出力形式

現在の SAS セッションのエンコーディングを使用する文字列を処理し、ネイティブエンディアン、32 ビット、UCS4 形式の UNICODE (ユニコード)文字列を書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS4Xw.

説明

w

出力フィールドの幅を指定します。32 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 4

範囲: 4–32767

詳細

\$UCS4Xw.出力形式では、動作環境のバイトオーダー、32 ビットの UCS4 (universal character set code in four octets)形式の UNICODE (ユニコード)文字列を書き出します。

比較

\$UCS4Xw.出力形式では、\$UCS4XEW.出力形式とは逆の処理を行います。同一の動作環境内のデータを変換する場合は、\$UCS4Xw.出力形式を使用します。異なる動作環境でデータを変換する場合は、\$UCS4Bw.出力形式または\$UCS4Lw.出力形式を使用します。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)セッションエンコーディングを使用します。

ステートメント	結果
	-----1

ステートメント	結果
<pre>x = '大'; put x \$ucs4x4.;</pre>	'00005927'x (binary) or '27590000'x (little endian)

関連項目:

出力形式:

- “\$UCS2Lw. 出力形式” (208 ページ)
- “\$UCS4XEw. 出力形式” (219 ページ)
- “\$UCS2Xw. 出力形式” (211 ページ)
- “\$UCS4Bw. 出力形式” (213 ページ)
- “\$UCS4Lw. 出力形式” (215 ページ)
- “\$UTF8Xw. 出力形式” (226 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2Lw. 入力形式” (421 ページ)
- “\$UCS2Xw. 入力形式” (423 ページ)
- “\$UCS4Bw. 入力形式” (425 ページ)
- “\$UCS4Bw. 出力形式” (213 ページ)
- “\$UCS4Lw. 入力形式” (426 ページ)
- “\$UCS4Xw. 入力形式” (427 ページ)
- “\$UTF8Xw. 入力形式” (436 ページ)

\$UCS4XEw. 出力形式

ネイティブエンディアン、32ビット、UCS4形式のUNICODE(ユニコード)で文字列を処理し、その文字列を現在のSASセッションのエンコーディングで書き出します。

カテゴリ: 文字

配置: 左

構文

\$UCS4XEw.

説明

w

出力フィールドの幅を指定します。32ビットサイズのUNICODE文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

\$UCS4XEw.出力形式では、現在の SAS セッションのエンコーディングで文字列を書き出します。ネイティブエンディアン、32 ビット、UCS4 (universal character set code in four octets)形式の UNICODE (ユニコード)文字列を処理します。

比較

\$UCS4XEw.出力形式では、\$UCS4XEw.出力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+----1
<pre>x = '275900004100000042000000'x; put x \$ucs4be4.;</pre>	大AB (little endian)

関連項目:

出力形式:

- “\$UCS4Xw. 出力形式” (218 ページ)

入力形式:

- “\$UCS4Xw. 入力形式” (427 ページ)

\$UESCw. 出力形式

現在の SAS セッションでエンコードされる文字列を処理し、その文字列を Unicode エスケープ(UESC)表記で書き出します。

カテゴリ: 文字

配置: 左

構文

\$UESCw.

説明

w
入力幅を指定します。

デフォルト: 8
 範囲: 1–32000

詳細

0 から 9、a から z、A から Z など、すべての動作環境で使用可能でない文字は、UESC で表す必要があります。\$UESCw はネストできます。

比較

\$UESCw 出力形式では、\$UESCEw 出力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされた日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+-----2
<code>x='大';</code>	¥u5927
<code>y='u5927'</code>	¥uu5927
<code>z='uu5927';</code>	¥uuu5927
<code>put x = \$uesc10. ;</code>	
<code>put y = \$uesc10. ;</code>	
<code>put z = \$uesc10. ;</code>	

関連項目:

出力形式:

- [“\\$UESCEw. 出力形式” \(221 ページ\)](#)

入力形式:

- [“\\$UESCw. 入力形式” \(429 ページ\)](#)
- [“\\$UESCEw. 入力形式” \(430 ページ\)](#)

\$UESCEw. 出力形式

Unicode エスケープ(UESC)表記の文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。

カテゴリ: 文字

配置: 左

構文

\$UESCEw.

説明

w

出力フィールドの幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

データが現在の SAS セッションのエンコーディングによってサポートされていない場合、データは UESC のまま残ります。

比較

\$UESCEw. 出力形式では、\$UESCw.出力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされている日本語(Shift_JIS)セッションエンコーディングを使用します。

ステートメント	結果
	----+-----1-----+-----2
x=put ('¥u5927', \$uesce10.) ;	x= 太
x=put ('¥uu5927', \$uesce10.) ;	x=¥u5927
x=put ('¥uuu5927', \$uesce10.) ;	x=¥uu5927

関連項目:

出力形式:

- “\$UESCw. 出力形式” (220 ページ)

入力形式:

- “\$UESCw. 入力形式” (429 ページ)
- “\$UESCEw. 入力形式” (430 ページ)

\$UNCRw. 出力形式

現在の SAS セッションでエンコードされる文字列を処理し、その文字列を数値文字表現(NCR)で書き出します。

カテゴリ: 文字

配置: 左

構文

\$UNCRw.

説明

w
 出力フィールドの幅を指定します。
 デフォルト: 8
 範囲: 1–32000

比較

\$UNCRw.出力形式では、\$UNCREw.出力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされている日本語(Shift_JIS)セッションエンコーディングを使用します。

ステートメント	結果
	----+-----1-----+-----2
<pre>x='91E5'x ; /* Japanese '大' in Shift-JIS */</pre>	大 abc
<pre>y='abc' ; put x \$uncr10.; put y \$uncr10.;</pre>	

関連項目:

出力形式:

- [“\\$UNCREw. 出力形式” \(223 ページ\)](#)

入力形式:

- [“\\$UNCRw. 入力形式” \(431 ページ\)](#)
- [“\\$UNCREw. 入力形式” \(432 ページ\)](#)

\$UNCREw. 出力形式

数値文字表現(NCR)の文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。

カテゴリ: 文字

配置: 左

構文

`$UNCREw.`

説明

`w`

出力フィールドの幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

各国語の文字は NCR で表記する必要があります。

比較

`$UNCREw.`出力形式では、`$UNCRw.`出力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされている日本語(Shift_JIS)セッションエンコーディングを使用します。

ステートメント	結果
	----+-----1
<pre>x='&#22823;abc'; put x \$uncr10.;</pre>	大 abc

関連項目:

出力形式:

- “[\\$UNCRw. 出力形式](#)” (222 ページ)

入力形式:

- “[\\$UNCRw. 入力形式](#)” (431 ページ)
- “[\\$UNCREw. 入力形式](#)” (432 ページ)

\$UPARENw. 出力形式

現在の SAS セッションでエンコードされる文字列を処理し、その文字列を Unicode かつこ(UPAREN)表記で書き出します。

カテゴリ: 文字

配置: 左

構文

\$SUPARENw.

説明

w
 出力フィールドの幅を指定します。
 デフォルト: 8
 範囲: 27–32000

詳細

文字列は、かっこおよび Unicode の 16 進表現によってエンコードされます。

比較

\$SUPARENw.出力形式では、\$SUPARENw.出力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされている日本語(Shift_JIS)セッションエンコーディングを使用します。

ステートメント	結果
	----+-----1-----+-----2-----+-----3-----+
x='太';	<u5927>
y='abc3';	<u0061> <u0062> <u0063> <u0033>
put x \$uparen7.;	
put y \$uparen28.;	

関連項目:

出力形式:

- [“\\$SUPARENw. 出力形式” \(225 ページ\)](#)

入力形式:

- [“\\$SUPARENw. 入力形式” \(433 ページ\)](#)
- [“\\$SUPARENw. 入力形式” \(434 ページ\)](#)

\$SUPARENw. 出力形式

Unicode かっこ(UPAREN)内の文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。

カテゴリ: 文字

配置: 左

構文

`$UPARENw.`

説明

`w`
 出力フィールドの幅を指定します。
 デフォルト: 8
 範囲: 1–32000

比較

`$UPARENw.`出力形式では、`$UPARENw.`出力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされた日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+
<code>x='<u0061><u0062><u0063><u0033>';</code>	abc3
<code>put x \$uparene4.;</code>	

関連項目:

出力形式:

- “`$UPARENw.` 出力形式” (224 ページ)

入力形式:

- “`$UPARENw.` 入力形式” (433 ページ)
- “`$UPARENw.` 入力形式” (434 ページ)

`$UTF8Xw.` 出力形式

現在の SAS セッションのエンコーディングを使用する文字列を処理し、UTF-8 (Universal Transformation Format)エンコーディングの文字列を書き出します。

カテゴリ: 文字

配置: 左

構文

`$UTF8Xw.`

説明

w

出力フィールドの幅を指定します。変数が入る十分な文字幅を指定します。文字の幅は、個々の文字のコードポイント値によって異なります。

デフォルト: 8

範囲: 2–32767

比較

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)セッションエンコーディングを使用します。

ステートメント	結果
	----+-----1
<pre>x = '91E5'x; ; /* Japanese '太 ' in Shift-JIS */ put x \$utf8x10.;</pre>	x='e5a4a7'x

関連項目:

出力形式:

- “\$UCS2Bw. 出力形式” (206 ページ)
- “\$UCS2Lw. 出力形式” (208 ページ)
- “\$UCS2Xw. 出力形式” (211 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2Lw. 入力形式” (421 ページ)
- “\$UCS2Xw. 入力形式” (423 ページ)

\$UTF8XEw. 出力形式

UTF-8 (Universal Transformation Format)の文字列を処理し、その文字列を現在の SAS セッションのエンコーディングで書き出します。

カテゴリ: 文字

配置: 左

構文

\$UTF8XEw.

説明

w

出力フィールドの幅を指定します。変数が入る十分な文字幅を指定します。文字の幅は、個々の文字のコードポイント値によって異なります。

デフォルト: 8

範囲: 1–32000

比較

\$UTF8XE*w*.出力形式では、\$UTF8X*w*.出力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)セッションエンコーディングを使用します。

ステートメント	結果
	----+-----1
<pre>x = unicode('u5927'); put x \$utf8xe10.;</pre>	大

関連項目:

出力形式:

- “\$UTF8X*w*. 出力形式” (226 ページ)

入力形式:

- “\$UTF8X*w*. 入力形式” (436 ページ)

\$VSLOG*w*. 出力形式

視覚的順序で並ぶ文字列を処理し、その文字列を左から右の論理的順序で書き出します。

カテゴリ: BIDI テキストの操作

配置: 左

構文

\$VSLOG*w*.

説明

w

出力フィールドの幅を指定します。

デフォルト: 200

範囲: 1–32767

詳細

\$VSLOGw.出力形式は、視覚的順序で保存されるデータを転送するときに使用します。たとえば、UNIX サーバーから Windows クライアントにデータを転送する場合があります。

注: \$VSLOGw.出力形式では、データ文字列の一部の組み合わせが正しく処理されません。

比較

\$VSLOGw.出力形式では、\$VSLOGRw.出力形式とは逆の処理を行います。

例

次の例では、“flight”のヘブライ語の入力値 `פּוּרְיָוִט` を使用します。

ステートメント	結果
	----+-----1-----+-----2-----+
<code>put text \$vslog12.;</code>	<code>פּוּרְיָוִט flight</code>

次の例では、“computer”のアラビア語の入力値 `كَمْبِيوتَر` を使用します。

ステートメント	結果
	----+-----1-----+-----2-----+
<code>put text \$vslog12.;</code>	<code>كَمْبِيوتَر computer</code>

関連項目:

出力形式:

- [“\\$VSLOGRw. 出力形式” \(229 ページ\)](#)

入力形式:

- [“\\$VSLOGw. 入力形式” \(437 ページ\)](#)
- [“\\$VSLOGRw. 入力形式” \(438 ページ\)](#)

\$VSLOGRw. 出力形式

視覚的順序で並ぶ文字列を処理し、その文字列を右から左の論理的順序で書き出します。

カテゴリ: BIDI テキストの操作

配置: 左

構文

`$VSLOGRw`.

説明

`w`

出力フィールドの幅を指定します。

デフォルト: 200

範囲: 1-32767

詳細

`$VSLOGRw`出力形式は、視覚的順序で保存されるデータを転送するときに使用します。たとえば、UNIX サーバーから Windows クライアントにデータを転送する場合があります。

注: `$VSLOGRw`出力形式では、データ文字列の一部の組み合わせが正しく処理されません。

比較

`$VSLOGRw`出力形式では、`$VSLOGRw`出力形式とは逆の処理を行います。

例

次の例では、“flight”のヘブライ語の入力値 `תִּירוּס` を使用します。

ステートメント	結果
	----+-----1-----+
<code>put text \$logvs12;</code>	<code>flight תִּירוּס</code>

次の例では、“computer”のアラビア語の入力値 `تاذ` を使用します。

ステートメント	結果
	----+-----1-----+
<code>put text \$logvs12;</code>	<code>تاذ computer</code>

関連項目:

入力形式:

- “`$VSLOGw` 入力形式” (437 ページ)
- “`$VSLOGRw` 入力形式” (438 ページ)

WEEKUw. 出力形式

U アルゴリズムを使用して、10 進数の週番号を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

WEEKUw.

説明

w

出力フィールドの幅を指定します。

デフォルト: 11

範囲: 3–200

詳細

WEEKUw.出力形式では、週番号形式で書き出します。WEEKUw.出力形式では、指定された幅に応じてさまざまな形式で書き出します。U アルゴリズムでは、年内の週番号を使用して SAS 日付値を計算します(日曜日を週の最初の日と見なします)。週番号値は、先頭に 0 を付けた 0 から 53 の範囲の 10 進数として表され、最大値は 53 になります。たとえば、年の 5 週目は 05 として表されます。

幅、出力形式および例については、次の表を参照してください。

幅	出力形式	例
3-4	Www	w01
5-6	yyWww	03W01
7-8	yyWwwdd	03W0101
9-10	yyyyWwwdd	2003W0101
11-200	yyyy-Www-dd	2003-W01-01

比較

WEEKVw.出力形式では、01 から 53 の範囲の 10 進数として週番号を書き出します。各週は月曜日から始まり、年の第 1 週には 1 月 4 日と年の最初の木曜日の両方が含まれます。1 月の最初の月曜日が 2 日、3 日または 4 日の場合、それより前の日は前年の最後の週に組み込まれます。WEEKWw.出力形式では、00 から 53 の範囲の 10 進数として年の週番号を書き出します。第 1 週の最初の日は月曜日になります。WEEKUw.出力形式では、先頭に 0 を付けた 0 から 53 の範囲の 10 進数として年の週番号を書き出します(日曜日が週の最初の日です)。

例

```
sasdate = '01JAN2003'd;
```

ステートメント	結果
	----+----1----
v=put (sasdate, weeku3.);	W00
w=put (sasdate, weeku5.);	03W00
x=put (sasdate, weeku7.);	03W0004
y=put (sasdate, weeku9.);	2003W0004
z=put (sasdate, weeku11.);	2003-W00-04
put v;	
put w;	
put x;	
put y;	
put z;	

関連項目:

出力形式:

- “WEEKVw. 出力形式” (232 ページ)
- “WEEKWw. 出力形式” (234 ページ)

WEEKVw. 出力形式

V アルゴリズムを使用して、10 進数の週番号を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

WEEKVw.

説明

w

出力フィールドの幅を指定します。

デフォルト: 11

範囲: 3–200

詳細

WEEKVw. 出力形式では、指定した幅に応じてさまざまな形式を書き出します。V アルゴリズムでは、先頭に 0 を付けた 01 から 53 の範囲の 10 進数として表され、最大値が 53 になる週番号値を使用して、SAS 日付値を計算します。各週は月曜日から始まり、年の第 1 週は 1 月 4 日と年の最初の木曜日の両方を含む週です。1 月の最初の月曜日が 2 日、3 日または 4 日の場合、それより前の日は前年の最後の週に組み込まれます。たとえば、年の 5 週目は 06 として表されます。

幅、出力形式および例については、次の表を参照してください。

幅	出力形式	例
3-4	Www	w01
5-6	yyWww	03W01
7-8	yyWwwdd	03W0101
9-10	yyyyWwwdd	2003W0101
11-200	yyyy-Www-dd	2003-W01-01

比較

WEEKVw.出力形式では、01 から 53 の範囲の 10 進数として週番号を書き出します。各週は月曜日から始まり、年の第 1 週には 1 月 4 日と年の最初の木曜日の両方が含まれます。1 月の最初の月曜日が 2 日、3 日または 4 日の場合、それより前の日は前年の最後の週に組み込まれます。WEEKWw.出力形式では、00 から 53 の範囲の 10 進数として年の週番号を書き出します。第 1 週の最初の日は月曜日になります。WEEKUw.出力形式では、先頭に 0 を付けた 0 から 53 の範囲の 10 進数として年の週番号を書き出します(日曜日が週の最初の日です)。

例

```
sasdate='01JAN2003'd;
```

ステートメント	結果
	----+-----1-----+
v=put(sasdate, weekv3.);	W01
w=put(sasdate, weekv5.);	03W01
x=put(sasdate, weekv7.);	03W0103
y=put(sasdate, weekv9.);	2003W0103
z=put(sasdate, weekv11.);	2003-W01-03
put v;	
put w;	
put x;	
put y;	
put z;	

関連項目:

出力形式:

- [“WEEKUw. 出力形式” \(231 ページ\)](#)
- [“WEEKWw. 出力形式” \(234 ページ\)](#)

WEEKWw. 出力形式

W アルゴリズムを使用して、10 進数の週番号を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

WEEKWw.

説明

w
出力フィールドの幅を指定します。

デフォルト: 11

範囲: 3–200

詳細

WEEKWw 出力形式では、指定した幅に応じてさまざまな形式を書き出します。W アルゴリズムでは、年内の週番号を使用して SAS 日付値を計算します(月曜日を週の最初の日と見なします)。週番号値は、先頭に 0 を付けた 0 から 53 の範囲の 10 進数として表され、最大値は 53 になります。たとえば、年の 5 週目は 05 として表されます。

幅、出力形式および例については、次の表を参照してください。

幅	出力形式	例
3-4	Www	w01
5-6	yyWww	03W01
7-8	yyWwwdd	03W0101
9-10	yyyyWwwdd	2003W0101
11-200	yyyy-Www-dd	2003-W01-01

比較

WEEKVw 出力形式では、01 から 53 の範囲の 10 進数として週番号を書き出します。月曜日と年の第 1 週から始まる週には、1 月 4 日と年の最初の木曜日が含まれます。1 月の最初の月曜日が 2 日、3 日または 4 日の場合、それより前の日は前年の最後の週に組み込まれます。WEEKWw 出力形式では、00 から 53 の範囲の 10 進数として年の週番号を書き出します。第 1 週の最初の日が月曜日になります。WEEKUw 出力形式では、先頭に 0 を付けた 0 から 53 の範囲の 10 進数として年の週番号を書き出します(日曜日が週の最初の日です)。

例

```
sasdate = '01JAN2003'd;
```

ステートメント	結果
	----+-----1----
v=put(sasdate, weekw3.);	W03
w=put(sasdate, weekw5.);	03W03
x=put(sasdate, weekw7.);	03W0003
y=put(sasdate, weekw9.);	2003W0003
z=put(sasdate, weekw11.);	2003-W00-03
put v;	
put w;	
put x;	
put y;	
put z;	

関連項目:

出力形式:

- [“WEEKUw. 出力形式” \(231 ページ\)](#)
- [“WEEKVw. 出力形式” \(232 ページ\)](#)

YENw.d 出力形式

円記号、桁区切り、小数点付きの数値を出力します。

カテゴリ: 数値

配置: 右

構文

YENw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 8

範囲: 1–32

d

数値の小数点以下の桁数を指定します。

範囲: 0–9

詳細

YENw.d 出力形式は、先頭に円記号を付け、各値を 3 桁ごとにカンマで区切って数値を書き出します。

円記号文字のコードの 16 進表現は、EBCDIC のシステムでは 5B、ASCII のシステムでは 5C です。これらのコードが表す通貨記号は、他の国では異なる場合があります。

例

```
put cost yen10.2;

data _null_;
value=1254.71;
put value yen10.2;
run;
```

Cost	結果
	----+-----1
1254.71	¥1,254.71

関連項目:

入力形式:

- “YENw.d 入力形式” (439 ページ)

YYWEEKUw. 出力形式

曜日情報を除外し、U アルゴリズムを使用して、10 進数の週番号を書き出します。

カテゴリ: 日付と時間
配置: 左

構文

YYWEEKUw.

説明

w
 出力フィールドの幅を指定します。
デフォルト: 7
範囲: 3-8

詳細

YYWEEKUw. 出力形式では、週番号形式で書き出します。YYWEEKUw. 出力形式では、指定した幅に応じてさまざまな形式で書き出します。U アルゴリズムでは、年内の週番号を使用して SAS 日付値を計算します(日曜日を週の最初の日と見なします)。

幅、出力形式および例については、次の表を参照してください。

幅	出力形式	例
3-4	Www	W01
5-6	yyWww	07W01
7	yyyyWww	2007W01
8	yyyy-Www	2007-W01
9-above	無効	無効

比較

YYWEEKUw.出力形式は WEEKUw.出力形式に似ていますが、YYWEEKUw.出力形式では曜日情報を示しません。また、YYWEEKUw.出力形式では、8 より大きい幅は受け付けません。

例

```
sasdate = '01JAN2007'd;
```

ステートメント	結果
	-----1-----+
u=put(sasdate,yyweeku3.);	W00
v=put(sasdate,yyweeku4.);	W00
w=put(sasdate,yyweeku5.);	07W00
x=put(sasdate,yyweeku6.);	07W00
y=put(sasdate,yyweeku7.);	2007W00
z=put(sasdate,yyweeku8.);	2007-W00
put u;	
put v;	
put w;	
put x;	
put y;	
put z;	

関連項目:

出力形式:

- [“WEEKUw. 出力形式” \(231 ページ\)](#)

YYWEEKVw. 出力形式

曜日情報を除外し、V アルゴリズムを使用して、10 進数の週番号を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

YYWEEKV_w.

説明

w

出力フィールドの幅を指定します。

デフォルト: 7

範囲: 3-8

詳細

YYWEEKV_w出力形式では、指定した幅に応じてさまざまな形式を書き出します。V アルゴリズムでは、先頭に0を付けた01から53の範囲の10進数として表され、最大値が53になる週番号値を使用して、SAS日付値を計算します。各週は月曜日から始まり、年の第1週は1月4日と年の最初の木曜日の両方を含む週です。1月の最初の月曜日が2日、3日または4日の場合、それより前の日は前年の最後の週に組み込まれます。たとえば、年の5週目は06として表されます。

幅、出力形式および例については、次の表を参照してください。

幅	出力形式	例
3-4	Www	w01
5-6	yyWww	07W01
7	yyyyWww	2007W01
8	yyyy-Www	2007-W01
9-above	無効	無効

比較

YYWEEKV_w出力形式はWEEKV_w出力形式に似ていますが、YYWEEKV_w出力形式では曜日情報を示しません。また、YYWEEKV_w出力形式では、8より大きい幅は受け付けません。

例

```
sasdate = '01JAN2007'd;
```

ステートメント	結果
	-----1-----+

ステートメント	結果
u=put(sasdate,yyweekv3.);	W01
v=put(sasdate,yyweekv4.);	W01
w=put(sasdate,yyweekv5.);	07W01
x=put(sasdate,yyweekv6.);	07W01
y=put(sasdate,yyweekv7.);	2007W01
z=put(sasdate,yyweekv8.);	2007-W01
put u;	
put v;	
put w;	
put x;	
put y;	
put z;	

関連項目:

出力形式:

- [“WEEKVw. 出力形式” \(232 ページ\)](#)

YYWEEKWw. 出力形式

曜日情報を除外し、W アルゴリズムを使用して、10 進数の週番号を書き出します。

カテゴリ: 日付と時間

配置: 左

構文

YYWEEKW_w.

説明

w
出力フィールドの幅を指定します。

デフォルト: 7

範囲: 3–8

詳細

YYWEEKW_w.出力形式では、指定した幅に応じてさまざまな形式を書き出します。W アルゴリズムでは、年内の週番号を使用して SAS 日付値を計算します。

幅、出力形式および例については、次の表を参照してください。

幅	出力形式	例
3-4	W _{ww}	W01
5-6	yyW _{ww}	07W01

7	yyyyWww	2007W01
8	yyyy-Www	2007-W01
9-above	無効	無効

比較

YYWEEKW_w.出力形式は WEEKW_w.出力形式に似ていますが、YYWEEKW_w.出力形式では曜日情報を示しません。また、YYWEEKW_w.出力形式は 8 より大きい幅は受け付けません。

例

```
sasdate = '01JAN2007'd
```

ステートメント	結果
	-----1-----
u=put(sasdate,yyweekw3.);	W01
v=put(sasdate,yyweekw4.);	W01
w=put(sasdate,yyweekw5.);	07W01
x=put(sasdate,yyweekw6.);	07W01
y=put(sasdate,yyweekw7.);	2007W01
z=put(sasdate,yyweekw8.);	2007-W01
put u;	
put v;	
put w;	
put x;	
put y;	
put z;	

関連項目:

出力形式:

- [“WEEKW_w. 出力形式” \(234 ページ\)](#)

5 部

各国語サポート関連の関数

10 章		
	SAS 文字列関数の国際化の互換性	243
11 章		
	関数エントリ	255

10 章

SAS 文字列関数の国際化の互換性

SAS 文字列関数の国際化の互換性 243

SAS 文字列関数の国際化の互換性

SAS では、文字データを簡単に操作できる文字列関数と CALL ルーチンを提供します。従来の SAS 文字列関数の多くは、1 文字のサイズが常に 1 バイトであることを前提としています。この処理は、1 バイト文字セット(SBCS)のデータでは正常に動作します。ただし、これらの関数と CALL ルーチンの一部を 2 バイト文字セット(DBCS)または複数バイト文字セット(MBCS)のデータに使用すると、多くの場合データは適切に処理されず、正しい結果が得られません。

DBCS エンコーディングでは、各文字を表示するためにさまざまなバイト数が必要です。MBCS は DBCS と同じ意味で使用される場合があります。

この問題を解決するために、DBCS と MBCS データを慎重に処理する文字列操作が可能な、K 関数と呼ばれる文字列関数と CALL ルーチンが導入されました。このページでは、各 SAS 文字列関数の I18N 互換性のレベルを説明します。I18N とは、国際化の略称です。互換性とは、特定の文字列関数を使用するプログラムが、プログラムを変更することなく異なる言語とロケールに適合できるかどうかを意味します。

ユーザーが K 関数を適切に使用するには、バイトベースのオフセット長と文字ベースのオフセット長の違いを理解する必要があります。ほとんどの K 関数では、文字ベースのオフセットまたは長さが必要です。SBCS 環境では、バイトベースの単位と文字ベースの単位は同一です。ただし、DBCS または MBCS 環境ではこの 2 つに大きな違いがあり、プログラマはこれを区別する必要があります。ユーザーが K 関数を使用するには、プログラムロジックの変更が必要な場合があります。ほとんどの K 関数では、現在の SAS セッションエンコーディングでエンコードされた文字列が必要です。

文字列関数は、関数が DBCS、MBCS または SBCS を処理できるかどうかに応じて I18N レベルが割り当てられます。レベルの説明を次に示します。

I18N レベル 0

この関数は SBCS データ向けに設計されています。この関数を DBCS または MBCS データの処理に使用しないでください。

I18N レベル 1

DBCS または MBCS データを処理する場合、可能な限りこの関数の使用は避けてください。I18N レベル 1 の関数は、特定の環境下では DBCS または MBCS エンコーディングを使用すると正常に動作しない場合があります。

I18N レベル 2

この関数は SBCS、DBCS、MBCS (UTF-8)データに使用できます。

表 10.1 SAS 文字列関数

関数	説明	I18N レベル 0	I18N レベル 1	I18N レベル 2
“ANYALNUM Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から英数字を検索し、最初に検索された文字の位置を返します。			X
“ANYALPHA Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から英字を検索し、最初に検索された文字の位置を返します。			X
“ANYCNTRL Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列からコントロール文字を検索し、最初に検索された文字の位置を返します。			X
“ANYDIGIT Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から数字を検索し、最初に検索された数字の位置を返します。			X
“ANYFIRST Function” in <i>SAS Functions and CALL Routines: Reference</i>	VALIDVARNAME=V7 において、SAS 変数名の開始文字として有効な文字を文字列から検索し、最初に検索された文字の位置を返します。			X
“ANYGRAPH Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列からグラフィカル文字を検索し、最初に検索された文字の位置を返します。			X
“ANYLOWER Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から小文字を検索し、最初に検索された文字の位置を返します。			X
“ANYNAME Function” in <i>SAS Functions and CALL Routines: Reference</i>	VALIDVARNAME=V7 において、SAS 変数名として有効な文字を文字列から検索し、最初に検索された文字の位置を返します。			X
“ANYPRINT Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から印刷可能な文字を検索し、最初に検索された文字の位置を返します。			X
“ANYPUNCT Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から句読文字を検索し、最初に検索された文字の位置を返します。			X

関数	説明	I18N レベル 0	I18N レベル 1	I18N レベル 2
“ANYSPACE Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から空白文字 (空白、水平タブ、垂直タブ、キャリッジリターン、ラインフィード、フォームフィード) を検索します。最初に検索された文字の位置を返します。			X
“ANYUPPER Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から大文字を検索し、最初に検索された文字の位置を返します。			X
“ANYXDIGIT Function” in <i>SAS Functions and CALL Routines: Reference</i>	数字を表す 16 進法の文字を文字列から検索し、最初に検索された文字の位置を返します。			X
“BYTE Function” in <i>SAS Functions and CALL Routines: Reference</i>	ASCII 照合順序または EBCDIC 照合順序の 1 文字を返します。	X		
“CAT Function” in <i>SAS Functions and CALL Routines: Reference</i>	先頭または末尾の空白を削除せずに、連結文字列を返します。			X
“CATS Function” in <i>SAS Functions and CALL Routines: Reference</i>	先頭と末尾の空白を削除して、連結文字列を返します。	X		
“CATT Function” in <i>SAS Functions and CALL Routines: Reference</i>	末尾の空白を削除して、連結文字列を返します。	X		
“CATX Function” in <i>SAS Functions and CALL Routines: Reference</i>	先頭と末尾の空白を削除し、区切り文字を挿入して、文字列を返します。	X		
“CHOOSEC Function” in <i>SAS Functions and CALL Routines: Reference</i>	引数のリストからの選択結果を表す文字値を返します。			X
“CHOOSEN Function” in <i>SAS Functions and CALL Routines: Reference</i>	引数のリストからの選択結果を表す数値を返します。			X
“COALESCEC Function” in <i>SAS Functions and CALL Routines: Reference</i>	数値の引数のリストからの最初の非欠損値を返します。			X

関数	説明	I18N レベル 0	I18N レベル 1	I18N レベル 2
“COALESCEC Function” in <i>SAS Functions and CALL Routines: Reference</i>	ASCII 照合順序または EBCDIC 照合順序の文字列を返します。	X		
“COMPARE Function” in <i>SAS Functions and CALL Routines: Reference</i>	2 つの文字列を比較し、異なる文字が検出された場合には最も左にある文字の位置を返し、異なる文字が検出されない場合には 0 を返します。	X		
“COMPARE Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列内のワード間にある複数の空白を取り除きます。	X		
“COMPGED Function” in <i>SAS Functions and CALL Routines: Reference</i>	2 つの文字列間の一般化編集距離を返します。	X		
“COMPLEV Function” in <i>SAS Functions and CALL Routines: Reference</i>	2 つの文字列間のレーベンシュタインの編集距離を返します。	X		
“COMPRESS Function” in <i>SAS Functions and CALL Routines: Reference</i>	元の文字から指定した文字を削除した文字列を返します。	X		
“COUNT Function” in <i>SAS Functions and CALL Routines: Reference</i>	指定した部分文字列が文字列内に含まれる回数を数えます。		X	
“COUNTC Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字のリストに表示される(または表示されない)文字列内の文字の個数を数えます。		X	
“DEQUOTE Function” in <i>SAS Functions and CALL Routines: Reference</i>	引用符で始まる文字列から、対となる引用符を削除し、閉じる側の引用符より右側にあるすべての文字を削除します。			X
“FIND Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列内の特定の部分文字列を検索します。		X	
“FINDC Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字のリストにある各文字を文字列から検索します。		X	

関数	説明	I18N レベル 0	I18N レベル 1	I18N レベル 2
“HTMLDECODE Function” in <i>SAS Functions and CALL Routines: Reference</i>	HTML 数値文字参照または HTML 文字実体参照を含む文字列をデコードし、デコードされた文字列を返します。			X
“HTMLENCODE Function” in <i>SAS Functions and CALL Routines: Reference</i>	HTML 文字実体参照を使用して文字列をエンコードし、エンコードされた文字列を返します。			X
“IFC Function” in <i>SAS Functions and CALL Routines: Reference</i>	式の真、偽、欠損に基づいて文字値を返します。			X
“IFN Function” in <i>SAS Functions and CALL Routines: Reference</i>	式の真、偽、欠損に基づいて数値を返します。			X
“INDEX Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字式から文字列を検索し、最初に検索された文字列の最初の文字の位置を返します。	X		
“INDEXC Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字式から指定した文字すべてを検索し、その文字の位置を返します。	X		
“INDEXW Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字式から単語として指定した文字列を検索し、単語の最初の文字の位置を返します。	X		
“KCOMPARE 関数” (p. 264)	文字式の比較結果を返します。			X
“KCOMPRESS 関数” (p. 265)	指定された文字を文字式から削除します。			X
“KCOUNT 関数” (p. 266)	式の 2 バイト文字の数を返します。			X
“KCVT 関数” (p. 266)	データのある種類のエンコーディングデータから別の種類のエンコーディングデータに変換します。			X
“KINDEX 関数” (p. 268)	文字式から文字列を検索します。			X

関数	説明	I18N レベル 0	I18N レベル 1	I18N レベル 2
“KINDEXC 関数” (p. 268)	文字式から指定した文字を検索します。			X
“KLEFT 関数” (p. 269)	不要な先頭の DBCS の空白と SO-SI を削除し、文字式を左詰めにします。			X
“KLENGTH 関数” (p. 270)	引数の長さを返します。			X
“KLOWCASE 関数” (p. 270)	引数のすべての文字を小文字に変換します。			X
“KREVERSE 関数” (p. 276)	文字式を逆にします。			X
“KRIGHT 関数” (p. 276)	末尾の DBCS の空白と SO-SI を削除し、文字式を右詰めにします。			X
“KSCAN 関数” (p. 277)	指定された単語を文字式から選択します。			X
“KSTRCAT 関数” (p. 278)	複数の文字式を連結します。			X
“KSUBSTR 関数” (p. 278)	引数から部分文字列を抽出します。			X
“KSUBSTRB 関数” (p. 279)	引数内の部分文字列のバイト位置に従って、引数から部分文字列を抽出します。			X
“KTRANSLATE 関数” (p. 280)	文字式の特定の文字を置換します。			X
“KTRIM 関数” (p. 281)	末尾の DBCS の空白と SO-SI を文字式から削除します。			X
“KTRUNCATE 関数” (p. 281)	数値を指定した長さで切り捨てます。			X
“KUPCASE 関数” (p. 282)	引数内のすべての文字を大文字に変換します。			X
“KUPDATE 関数” (p. 282)	文字値の内容を挿入、削除および置換します。			X

関数	説明	I18N レベル 0	I18N レベル 1	I18N レベル 2
“KUPDATEB 関数” (p. 284)	引数内の文字値のバイト位置に従って、文字値の内容を挿入、削除および置換します。			X
“KVERIFY 関数” (p. 285)	式に固有の最初の文字の位置を返します。			X
“LEFT Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列を左詰めにします。	X		
“LENGTH Function” in <i>SAS Functions and CALL Routines: Reference</i>	末尾の空白を除いた文字列の長さを返します。文字列が空白の場合には、1 を返します。	X		
“LENGTHC Function” in <i>SAS Functions and CALL Routines: Reference</i>	末尾の空白を含めた文字列の長さを返します。			X
“LENGTHM Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列に割り当てられたメモリの量を返します(バイト単位)。			X
“LENGTHN Function” in <i>SAS Functions and CALL Routines: Reference</i>	末尾の空白を除いた文字列の長さを返します。	X		
“LOWCASE Function” in <i>SAS Functions and CALL Routines: Reference</i>	引数のすべての文字を小文字に変換します。			X
“MISSING Function” in <i>SAS Functions and CALL Routines: Reference</i>	引数が欠損値を含むかどうかの結果を表す数値を返します。			X
“NLITERAL Function” in <i>SAS Functions and CALL Routines: Reference</i>	指定した文字列を SAS 名前リテラルに変換します。			X
“NOTALNUM Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から英数字ではない文字を検索し、最初に検索された文字の位置を返します。			X
“NOTALPHA Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から英字ではない文字を検索し、最初に検索された文字の位置を返します。			X

関数	説明	I18N レベル 0	I18N レベル 1	I18N レベル 2
“NOTCNTRL Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列からコントロール文字ではない文字を検索し、最初に検索された文字の位置を返します。			X
“NOTDIGIT Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から数字ではない文字を検索し、最初に検索された文字の位置を返します。			X
“NOTFIRST Function” in <i>SAS Functions and CALL Routines: Reference</i>	VALIDVARNAME=V7 の場合、SAS 変数名として無効な開始文字を文字列から検索し、最初に検索された文字の位置を返します。			X
“NOTGRAPH Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列からグラフィカル文字ではない文字を検索し、最初に検索された文字の位置を返します。			X
“NOTLOWER Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から小文字ではない文字を検索し、最初に検索された文字の位置を返します。			X
“NOTNAME Function” in <i>SAS Functions and CALL Routines: Reference</i>	VALIDVARNAME=V7 の場合、SAS 変数名として無効な文字を文字列から検索し、最初に検索された文字の位置を返します。			X
“NOTPRINT Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から印刷不可文字を検索し、最初に検索された文字の位置を返します。	X		
“NOTPUNCT Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から句読文字ではない文字を検索し、最初に検索された文字の位置を返します。	X		

関数	説明	I18N レベル 0	I18N レベル 1	I18N レベル 2
“NOTSPACE Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から空白文字 (空白、水平タブ、垂直タブ、キャリッジリターン、ラインフィード、フォームフィード) ではない文字を検索し、最初に検索された文字の位置を返します。	X		
“NOTUPPER Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から大文字ではない文字を検索し、最初に検索された文字の位置を返します。			X
“NOTXDIGIT Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から 16 進数字ではない文字を検索し、最初に検索された文字の位置を返します。			X
“NVALID Function” in <i>SAS Functions and CALL Routines: Reference</i>	SAS 変数名として使用する文字列の有効性を確認します。	X		
“PROPCASE Function” in <i>SAS Functions and CALL Routines: Reference</i>	引数のすべての単語を適切に大文字と小文字に変換します。			X
“QUOTE Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字値に二重引用符を付加します。			X
“RANK Function” in <i>SAS Functions and CALL Routines: Reference</i>	ASCII 照合順序または EBCDIC 照合順序による文字の位置を返します。	X		
“REPEAT Function” in <i>SAS Functions and CALL Routines: Reference</i>	最初の引数を n+1 回繰り返した文字値を返します。		X	
“REVERSE Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列を逆にします。	X		
“RIGHT Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字式を右詰めにします。	X		
“SCAN Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から n 番目の単語を返します。	X		

関数	説明	I18N レベル 0	I18N レベル 1	I18N レベル 2
“SOUNDEX Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列をエンコードして検索しやすくします。	X		
“SPEDIS Function” in <i>SAS Functions and CALL Routines: Reference</i>	2 つの単語の一致尤度を調べて、2 つの単語間のスペルの違いをコストで表します。	X		
“STRIP Function” in <i>SAS Functions and CALL Routines: Reference</i>	先頭と末尾の空白を削除して文字列を返します。	X		
“SUBPAD Function” in <i>SAS Functions and CALL Routines: Reference</i>	必要に応じて空白埋め込みを使用し、指定した長さの部分文字列を返します。		X	
“SUBSTR (left of =) Function” in <i>SAS Functions and CALL Routines: Reference</i>	引数から部分文字列を抽出します。	X		
“SUBSTRN Function” in <i>SAS Functions and CALL Routines: Reference</i>	部分文字列を返します。長さがゼロの結果も返せます。		X	
“TRANSLATE Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列に含まれる特定の文字を置き換えます。	X		
“TRANTAB 関数” (306 ページ) “TRANTAB 関数” (306 ページ)	指定された変換テーブルを使用してデータをトランスコードします。	X		
“TRANWRD Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列に含まれる特定の部分文字列をすべて置き換えるか削除します。			X
“TRIM Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字列から末尾の空白を取り除きます。文字列が欠損値の場合は、1 つの空白を返します。	X		
“TRIMN Function” in <i>SAS Functions and CALL Routines: Reference</i>	文字式から末尾の空白を取り除きます。文字式が欠損値の場合は、長さがゼロの文字列を返します。	X		
“UPCASE Function” in <i>SAS Functions and CALL Routines: Reference</i>	引数内のすべての文字を大文字に変換します。			X

関数	説明	I18Nレベル0	I18Nレベル1	I18Nレベル2
“URLDECODE Function” in <i>SAS Functions and CALL Routines: Reference</i>	URLのエスケープ構文を使用してデコードされた文字列を返します。			X
“URLENCODE Function” in <i>SAS Functions and CALL Routines: Reference</i>	URLのエスケープ構文を使用してエンコードされた文字列を返します。			X
“VERIFY Function” in <i>SAS Functions and CALL Routines: Reference</i>	他の文字列に存在しない文字の最初の出現位置を返します。	X		

11 章

関数エントリ

カテゴリ別の関数	256
ディクショナリ	258
ENCODCOMPAT 関数	258
ENCODISVALID 関数	260
GETLOCENV 関数	260
GETPXLANGUAGE 関数	261
GETPXLOCALE 関数	262
GETPXREGION 関数	263
KCOMPARE 関数	264
KCOMPRESS 関数	265
KCOUNT 関数	266
KCVT 関数	266
KINDEX 関数	268
KINDEXC 関数	268
KLEFT 関数	269
KLENGTH 関数	270
KLOWCASE 関数	270
KPROPCASE 関数	271
KPROPCHAR 関数	273
KPROPDATA 関数	274
KREVERSE 関数	276
KRIGHT 関数	276
KSCAN 関数	277
KSTRCAT 関数	278
KSUBSTR 関数	278
KSUBSTRB 関数	279
KTRANSLATE 関数	280
KTRIM 関数	281
KTRUNCATE 関数	281
KUPCASE 関数	282
KUPDATE 関数	282
KUPDATEB 関数	284
KVERIFY 関数	285
NLDATE 関数	285
NLDATM 関数	288
NLTIME 関数	290
SASMSG 関数	292
SASMSG1 関数	294
SORTKEY 関数	297
SETLOCALE 関数	299
TRANTAB 関数	306

UNICODE 関数	307
UNICODEC 関数	308
UNICODELEN 関数	310
UNICODEWIDTH 関数	311
VARTRANSCODE 関数	311
VTRANSCODE 関数	313
VTRANSCODEX 関数	314

カテゴリ別の関数

各国語サポートに関するカテゴリ:

表 11.1 各国語サポート関連の関数のカテゴリ

カテゴリ	説明
文字	文字データを処理します。
通貨の換算	1 つの通貨を別の通貨に換算します。
DBCS	2 バイト文字セットを処理します。
日付と時間	日時データを処理します。
ロケール	指定されたロケールに基づいてデータを処理します。
変数情報	変数情報を処理します。

カテゴリ	言語要素	説明
DBCS	KCOMPARE 関数 (p. 264)	文字式の比較結果を返します。
	KCOMPRESS 関数 (p. 265)	指定された文字を文字式から削除します。
	KCOUNT 関数 (p. 266)	式の 2 バイト文字の数を返します。
	KINDEX 関数 (p. 268)	文字式から文字列を検索します。
	KINDEXC 関数 (p. 268)	文字式から指定した文字を検索します。
	KLEFT 関数 (p. 269)	不要な先頭の DBCS の空白と SO/SI を削除し、文字式を左詰めにします。
	KLENGTH 関数 (p. 270)	引数の長さを返します。
	KLOWCASE 関数 (p. 270)	引数のすべての文字を小文字に変換します。
	KPROPCASE 関数 (p. 271)	中国語、日本語、韓国語、台湾語(CJKT)文字を変換します。
	KPROPCHAR 関数 (p. 273)	特殊文字を標準文字に変換します。

カテゴリ	言語要素	説明
	KPROPDATA 関数 (p. 274)	印刷不可文字を削除または変換します。
	KREVERSE 関数 (p. 276)	文字式を逆にします。
	KRIGHT 関数 (p. 276)	末尾の DBCS の空白と SO/SI を削除し、文字式を右詰めにします。
	KSCAN 関数 (p. 277)	指定された単語を文字式から選択します。
	KSTRCAT 関数 (p. 278)	複数の文字式を連結します。
	KSUBSTR 関数 (p. 278)	引数から部分文字列を抽出します。
	KSUBSTRB 関数 (p. 279)	引数内の部分文字列のバイト位置に従って、引数から部分文字列を抽出します。
	KTRANSLATE 関数 (p. 280)	文字式の特定の文字を置換します。
	KTRIM 関数 (p. 281)	末尾の DBCS の空白と SO/SI を文字式から削除します。
	KTRUNCATE 関数 (p. 281)	マルチバイト文字を壊すことなく、文字列をバイト単位で、指定した長さに切ります。
	KUPCASE 関数 (p. 282)	引数内のすべての文字を大文字に変換します。
	KUPDATE 関数 (p. 282)	文字値の内容を挿入、削除および置換します。
	KUPDATEB 関数 (p. 284)	引数内の文字値のバイト位置に従って、文字値の内容を挿入、削除および置換します。
	KVERIFY 関数 (p. 285)	式に固有の最初の文字の位置を返します。
エンコーディング	ENCODCOMPAT 関数 (p. 258)	2つのエンコーディング間のトランスコーディング互換性を確認します。
	ENCODISVALID 関数 (p. 260)	有効なエンコーディング名を指定します。
日付と時間	NLDATE 関数 (p. 285)	SAS 日付値を、日付のディスクリプタを使用して指定したロケールの日付値に変換します。
	NLDATM 関数 (p. 288)	SAS 日時値を、日時形式のディスクリプタを使用して指定したロケールの時間値に変換します。
	NLTIME 関数 (p. 290)	SAS 時間値または SAS 日時値を、NLTIME ディスクリプタを使用して指定したロケールの時間値に変換します。
変数情報	VARTRANSCODE 関数 (p. 311)	SAS データセット変数のトランスコーディング属性を返します。
	VTRANSCODE 関数 (p. 313)	指定された文字変数でトランスコーディングが有効かどうかを示す値を返します。

カテゴリ	言語要素	説明
	VTRANSCODEX 関数 (p. 314)	指定された引数でトランスコーディングが有効かどうかを示す値を返します。
文字	KCVT 関数 (p. 266)	データがある種類のエンコーディングデータから別の種類のエンコーディングデータに変換します。
	TRANTAB 関数 (p. 306)	指定された変換テーブルを使用してデータをトランスコードします。
	UNICODE 関数 (p. 307)	Unicode 文字を現在の SAS セッションエンコーディングに変換します。
	UNICODEC 関数 (p. 308)	現在の SAS セッションエンコーディングの文字を Unicode 文字に変換します。
	UNICODELEN 関数 (p. 310)	Unicode データの文字単位の長さを指定します。
	UNICODEWIDTH 関数 (p. 311)	Unicode データの表示単位の長さを指定します。
ロケール	GETLOCENV 関数 (p. 260)	現在のロケール/言語環境を返します。
	GETPXLANGUAGE 関数 (p. 261)	現在の 2 文字の言語コードを返します。
	GETPXLOCALE 関数 (p. 262)	SAS ロケールの POSIX ロケール値を返します。
	GETPXREGION 関数 (p. 263)	現在の 2 文字の地域コードを返します。
	SASMSG 関数 (p. 292)	データセットからメッセージを指定します。返されるメッセージは現在のロケールと指定されたキーに基づきます。
	SASMSG1 関数 (p. 294)	データセットからメッセージを指定します。メッセージは、指定されたロケール値と指定されたキー値に基づきます。
	SORTKEY 関数 (p. 297)	言語並べ替えキーを作成します。
	SETLOCALE 関数 (p. 299)	現在の SAS ロケールのロケールキーを指定します。

ディクショナリ

ENCODCOMPAT 関数

2 つのエンコーディング間のトランスコーディング互換性を確認します。

カテゴリ: エンコーディング

構文

ENCODCOMPAT(*source1*, <*source2*>)

必須引数

source1

エンコーディングを表す文字列です。

source2

エンコーディングを表す文字列です。この引数はオプションです。

詳細

エンコーディングを1つ指定した場合、この関数は現在の SAS セッションのエンコーディングとの互換性を確認します。

エンコーディングを2つ指定した場合、この関数は2つのエンコーディング間の互換性を確認します。

この関数は2つのエンコーディング識別子を比較し、データをトランスコードする必要があるかどうかを評価します。*Source1* はソースのエンコーディングです。*Source2* はターゲットのエンコーディングです。7ビット ASCII を他のタイプの ASCII にトランスコーディングする場合の互換性はありますが、ASCII を7ビット ASCII にトランスコーディングする場合は、互換性がないことがあります。

ENCODCOMPAT 関数は次の値を指定します。

- 1 *Source1* は有効なエンコーディング名ではありません。
- 2 *Source2* は有効なエンコーディング名ではありません。
- 0 エンコーディング間に互換性はありません。トランスコーディングが必要です。
- 1 エンコーディング間の互換性があります。トランスコーディングは不要です。
- 2 改行文字が検出されました。

例

次の例で ENCODCOMPAT の機能を説明します。

ステートメント	結果
<pre>/* session encoding is wlatin1 */ isCompat= EncodCompat("xyz"); put isValid;</pre>	-1
<pre>/* session encoding is wlatin1 */ isCompat= EncodCompat("cp1252"); put isValid;</pre>	1
<pre>isCompat= EncodCompat ("ebcdic1149","open_ed-1149"); put isValid;</pre>	2
<pre>isCompat= EncodCompat ("cp1251","ebcdic1149"); put isValid;</pre>	0

ENCODISVALID 関数

有効なエンコーディング名を指定します。

カテゴリ: エンコーディング

構文

`ENCODISVALID(source)`

必須引数

source

エンコーディング名を表す文字列です。

詳細

ENCODISVALID 関数は次の値を返します。

- 0 文字列は有効なエンコーディング名ではありません。
- 1 文字列は有効な短いエンコーディング名です。
- 2 文字列は有効な長いエンコーディング名です。
- 3 文字列は有効なエイリアスエンコーディング名です。

例

次の例で ENCODISVALID の機能を説明します。

SAS ステートメント	結果
<pre>isValid= EncodIsValid("xyz"); put isValid;</pre>	0
<pre>isValid= EncodIsValid("wlt2"); put isValid;</pre>	1
<pre>isValid= EncodIsValid("wlatin2"); put isValid;</pre>	2
<pre>isValid= EncodIsValid("cp1250"); put isValid;</pre>	3

GETLOCENV 関数

現在のロケール/言語環境を返します。

カテゴリ: ロケール

構文

GETLOCENV()

詳細

GETLOCENV 関数は、有効な SAS ロケールのロケール/言語環境値を返します。次の環境値が表示されます。

SBCS

SAS セッションエンコーディングは SBCS (1 バイト文字セット)です。文字列の操作のため、SASWZSD が読み込まれます。

DBCS

SAS セッションエンコーディングは DBCS (2 バイト文字セット)です。文字列の操作のため、SASWZSD が読み込まれます。

MBCS

SAS セッションエンコーディングは Unicode (UTF8)です。文字列の操作のため、SASWZSU が読み込まれます。

空白値が返された場合は、WZSS サブシステムが使用できません。このアクションは、構成またはインストールでエラーが発生したことを示します。

例

次の例では、LOCALE=システムオプションが French_France に設定されています。

ステートメント	結果
option locale=french_france; environ=getlocenv(); put environ;	SBCS

GETPXLANGUAGE 関数

現在の 2 文字の言語コードを返します。

カテゴリ: ロケール

構文

GETPXLANGUAGE()

詳細

GETPXLANGUAGE 関数は、LOCALE=SAS システムオプションの現在の値に基づいて、2 文字の言語コードを返します。言語名の長さは 2 文字です。値を受け取る変数のサイズが 2 文字より小さい場合、値は切り捨てられます。

例

最初の例では、LOCALE=システムオプションが French_France に設定されています。2 番目の例では German に設定されています。3 番目の例では English_United States に設定されています。

ステートメント	結果
option locale=french_france; lang=getpxLanguage(); put lang;	fr
option locale=German; lang=getpxLanguage(); put lang;	de
option locale=en_US; lang=getpxLanguage(); put lang;	ja

関連項目:

システムオプション:

- [“LOCALE システムオプション” \(470 ページ\)](#)

関数:

- [“GETPXREGION 関数” \(263 ページ\)](#)
- [“GETPXLOCALE 関数” \(262 ページ\)](#)

GETPXLOCALE 関数

SAS ロケールの POSIX ロケール値を返します。

カテゴリ: ロケール

構文

GETPXLOCALE(<source>)

必須引数

<source>

ロケール名を指定する省略可能な引数です。

詳細

GETPXLOCALE 関数は、有効な SAS ロケール名に対する POSIX ロケール値を返します。無効なロケール名を指定すると、ヌル文字列が返されます。<source>引数に

値を指定しない場合、関数は現在の SAS セッションの POSIX 名を返します。POSIX ロケール名の長さは 5 文字です。値を受け取る変数のサイズが 5 文字より小さい場合、値は切り捨てられます。

例

最初の例では、LOCALE=システムオプションが French_France に設定されています。2 番目の例では、<source>引数が German_Germany に設定されています。3 番目の例では、<source>引数が English_United States に設定されています。

ステートメント	結果
option locale=french_france; locale=getpxLocale(); put locale;	fr_FR
locale=getpxLocale("german_germany"); put locale;	de_DE
locale=getpxLocale("english_unitedstates"); put locale;	en_US

関連項目:

システムオプション:

- [“LOCALE システムオプション” \(470 ページ\)](#)

関数:

- [“GETPXLANGUAGE 関数” \(261 ページ\)](#)
- [“GETPXREGION 関数” \(263 ページ\)](#)

GETPXREGION 関数

現在の 2 文字の地域コードを返します。

カテゴリ: ロケール

構文

GETPXREGION()

詳細

GETPXREGION 関数は、現在の LOCALE=SAS システムオプションに基づいて、2 文字の地域コードを返します。地域名の長さは 2 文字です。値を受け取る変数のサイズが 2 文字より小さい場合、値は切り捨てられます。

例

最初の例では、LOCALE=システムオプションが French_France に設定されています。2 番目の例では German に設定されています。3 番目の例では English_United States に設定されています。

ステートメント	結果
<pre>option locale=french_france; region=getpxRegion(); put region;</pre>	FR
<pre>option locale=German; region=getpxRegion(); put region;</pre>	DE
<pre>option locale=en_US; region=getpxRegion(); put region;</pre>	US

関連項目:

システムオプション:

- [“LOCALE システムオプション” \(470 ページ\)](#)

関数:

- [“GETPXLOCALE 関数” \(262 ページ\)](#)
- [“GETPXLANGUAGE 関数” \(261 ページ\)](#)

KCOMPARE 関数

文字式の比較結果を返します。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、“COMPARE Function” in *SAS Functions and CALL Routines: Reference* です。

構文

KCOMPARE(*source*, <*pos*, <*count*,>> *findstr*)

必須引数

source

比較される文字式を指定します。

pos

比較を開始する *source* の開始位置を指定します。*pos* を省略した場合、*source* 全体が比較されます。*pos* が 0 よりも小さい場合、*source* は SO/SI 文字を含まない拡張 DBCS データとして判断されます。

count

比較するバイト数を指定します。*count* を省略した場合、末尾の空白を除き、*pos* の後ろのすべての *source* が比較されます。

findstr

source と比較する文字式を指定します。

詳細

KCOMPARE では、値を次のように返します。

- *source* が *findstr* より小さい場合は負の値
- *source* が *findstr* と等しい場合は 0
- *source* が *findstr* より大きい場合は正の値

KCOMPRESS 関数

指定された文字を文字式から削除します。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、COMPARE で、SAS 関数と CALL ルーチン: リファレンスを参照してください。

構文

KCOMPRESS(*source*,<*characters-to-remove*>)

必須引数***source***

削除する文字を含む文字式を指定します。KCOMPRESS は、*source* のみが指定されると、すべての 1 バイトおよび 2 バイトの空白を削除してこの式を返します。

characters-to-remove

KCOMPRESS が文字式から削除する文字を指定します。*characters-to-remove* を省略すると、KCOMPRESS はすべての空白を削除します。

ヒント: 文字のリテラル文字列を引用符で囲みます。

関連項目:**関数:**

- “KLEFT 関数” (269 ページ)
- “KTRIM 関数” (281 ページ)

KCOUNT 関数

式の 2 バイト文字の数を返します。

カテゴリ: DBCS

構文

KCOUNT(*source*)

必須引数

source

数える文字式を指定します。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KCVT 関数

データある種類のエンコーディングデータから別の種類のエンコーディングデータに変換します。

カテゴリ: 文字

構文

KCVT(*text*, *intype*, *outtype*, <*options*,...>)

必須引数

text

変換する文字変数を指定します。

intype

データのエンコーディングを指定します。テキストのエンコーディングは、入力データのエンコーディングと一致する必要があります。有効な値については、“[データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値](#)” (563 ページ)を参照してください。

ASCHANY および EBCDICANY は、無効なエンコーディング値です。

outtype

文字データに変換するエンコーディングを指定します。有効な値については、“[データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値](#)” (563 ページ)を参照してください。

ASCHANY および EBCDICANY は、無効なエンコーディング値です。

options

文字データオプションを指定します。次のオプションを使用できます。

NOSOSI NOSHIFT	シフトコードと半角文字は含めません。
INPLACE	文字データを変換によって置き換えます。INPLACE オプションは、異なるホスト間で文字データの長さが異なる場合に、同じ位置を確保するために指定します。たとえば、INPLACE オプションは、データをシフトコードが必要なホストから、シフトコードが不要な別のホストに変換します。シフトコードに関して <i>outtype</i> に変換される文字データの長さが、 <i>intype</i> に指定された長さよりも長い場合には、切り捨てが行われます。
KANA	文字データの列に、半角カタカナ文字を含めます。
UPCASE	2 バイトの英字を大文字に変換します。
LOWCASE	2 バイトの英字を小文字に変換します。
KATA2HIRA	カタカナデータをひらがなに変換します。
HIRA2KATA	ひらがなデータをカタカナに変換します。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ) を参照してください。

KCVT 関数は、SBCS、DBCS および MBCS 文字列をエンコーディングデータに変換します。たとえば、KCVT 関数では、ASCII コードデータから UCS2 エンコーディングデータ、ギリシャ語コードデータから UTF-8、日本語 SJIS コードデータから別の日本語コードデータの変換が可能です。指定可能な *intype* および *outtype* オプションの種類は、UCS2、UCS2L、UCS2B、UTF8 です。DBCS モードを有効にするには、構成ファイルまたはコマンドラインで、次の SAS オプションを指定します。

- DBCS
- DBCSLANG Japanese | Korean | Chinese | Taiwanese
- DBCSTYPE dbcstype value

例

次のコードは、*my-input-file* として指定された外部テキストファイルについて IBM PC コードを DEC コードに変換し、OUTDD に書き出します。

```
data _null_;
infile 'my-input-file';
file outdd noprint;
input @1 text $char80.;
text = kcvf(text, 'pcibm', 'dec');
put @1 text $char80.;
run;
```

関連項目:

システムオプション:

- “[DBCS システムオプション: UNIX、Windows および z/OS](#)” (460 ページ)
- “[DBCSLANG システムオプション: UNIX、Windows および z/OS](#)” (461 ページ)

- “DBCSTYPE システムオプション: UNIX、Windows および z/OS” (462 ページ)

プロシジャ:

- 16 章, “DBCSTAB プロシジャ,” (515 ページ)

KINDEX 関数

文字式から文字列を検索します。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、INDEX です。SAS 関数と CALL ルーチン: リファレンスを参照してください。

構文

KINDEX(*source*, *excerpt*)

必須引数

source

検索する文字式を指定します。

excerpt

文字式で検索する文字列を指定します。

ヒント: 文字のリテラル文字列を引用符で囲みます。

詳細

制限と詳しい情報については、“SAS 文字列関数の国際化の互換性” (243 ページ)を参照してください。

KINDEX 関数は、*source* 内を左から右へ、*excerpt* で指定された文字列が最初に現れる個所を検索し、*source* 内でのその文字列の先頭文字の位置を返します。*source* 内に文字列が見つからない場合、KINDEX は値 0 を返します。文字列が複数回現れる場合、KINDEX は、最初の出現個所の位置のみを返します。

関連項目:

関数:

- “KINDEXC 関数” (268 ページ)

KINDEXC 関数

文字式から指定した文字を検索します。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、“INDEXC Function” in SAS Functions and CALL Routines: Reference です。

構文

KINDEXC(*source*,*excerpt-1*<,... *excerpt-n*>)

必須引数

source

検索する文字式を指定します。

excerpt

文字式で検索する文字を指定します。

ヒント: 複数の *excerpt* を指定する場合は、カンマで区切ります。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ) を参照してください。

KINDEXC 関数は、*source* を左から右へ、*excerpt* 内に含まれるいずれかの文字が最初に現れる個所を検索し、*source* 内でのその文字の位置を返します。*excerpt-1* から *excerpt-n* までの文字がいずれも *source* 内で見つからなかった場合、KINDEXC は値 0 を返します。

比較

KINDEXC 関数は、文字列に含まれる個々の文字が最初に現れる個所を検索するのに対し、KINDEX 関数は、文字列がパターンとして最初に現れる個所を検索します。

関連項目:

関数:

- “[KINDEX 関数](#)” (268 ページ)

KLEFT 関数

不要な先頭の DBCS の空白と SO/SI を削除し、文字式を左詰めにします。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、LEFT で、[SAS 関数と CALL ルーチン: リファレンス](#)を参照してください。

構文

KLEFT(*argument*)

必須引数

argument

SAS 文字式を指定します。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KLEFT は、引数を返して先頭の空白を削除します。

関連項目:

関数:

- “[KCOMPRESS 関数](#)” (265 ページ)
- “[KRIGHT 関数](#)” (276 ページ)
- “[KTRIM 関数](#)” (281 ページ)

KLENGTH 関数

引数の長さを返します。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、LENGTH で、[SAS 関数と CALL ルーチン: リファレンス](#)を参照してください。

構文

KLENGTH(*argument*)

必須引数

argument

SAS 式を指定します。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KLENGTH 関数は、引数内で最も右にある空白以外の文字の位置を表す整数を返します。引数の値がない場合、KLENGTH は値 1 を返します。引数が初期化されていない数値変数の場合、KLENGTH は値 12 を返し、SAS ログに数値が文字値に変換されたという注意を出力します。

KLOWCASE 関数

引数のすべての文字を小文字に変換します。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、LOWCASE で、[SAS 関数と CALL ルーチン: リファレンス](#)を参照してください。

構文

KLOWCASE(*argument*)

必須引数

argument

SAS 文字式を指定します。

詳細

制限と詳しい情報については、“SAS 文字列関数の国際化の互換性” (243 ページ)を参照してください。

KLOWCASE 関数は、文字引数をコピーし、すべての大文字を小文字に変換して、変更された値を結果として返します。

KPROPCASE 関数

中国語、日本語、韓国語、台湾語(CJKT)文字を変換します。

カテゴリ: DBCS

構文

str=KPROPCASE(<*instr*> , (<*options*>))

必須引数

str

変換された、現在の SAS セッションエンコーディングのデータ文字列。

instr

入力データ文字列。

options

指定されたオプションに基づいて、日本語、中国語、韓国語、台湾語文字を変換します。

HALF-KATAKANA, FULL-KATAKANA

半角カタカナを全角カタカナに変換します。このオプションは、日本語エンコーディングでのみ使用します。

制限事項: このオプションは、FULL-KATAKANA, HALF-KATAKANA オプションと同時に使用できません。

FULL-KATAKANA, HALF-KATAKANA

全角カタカナを半角カタカナに変換します。このオプションは、日本語エンコーディングでのみ使用します。

制限事項: このオプションは、HALF-KATAKANA, FULL-KATAKANA オプションと同時に使用できません。

KATAKANA, ROMAJI

カタカナ文字列をローマ字文字列に変換します。このオプションは、日本語エンコーディングでのみ使用します。

制限事項: このオプションは、ROMAJI, KATAKANA オプションと同時に使用できません。

ROMAJI, KATAKANA

ローマ字文字列をカタカナ文字列に変換します。このオプションは、日本語エンコーディングでのみ使用します。

制限事項: このオプションは、KATAKANA, ROMAJI オプションと同時に使用できません。

FULL-ALPHABET, HALF-ALPHABET

全角アルファベット文字を半角アルファベット文字に変換します。このオプションは、日本語、中国語、韓国語、台湾語エンコーディングでのみ使用します。

制限事項: このオプションは、HALF-ALPHABET, FULL-ALPHABET オプションと同時に使用できません。

HALF-ALPHABET, FULL-ALPHABET

半角アルファベット文字を全角アルファベット文字に変換します。このオプションは、日本語、中国語、韓国語、台湾語エンコーディングでのみ使用します。

制限事項: このオプションは、FULL-ALPHABET, HALF-ALPHABET オプションと同時に使用できません。

LOWERCASE, UPPERCASE

小文字アルファベット文字を大文字アルファベット文字に変換します。

制限事項: このオプションは、UPPERCASE, LOWERCASE オプションと同時に使用できません。

UPPERCASE, LOWERCASE

大文字アルファベット文字を小文字アルファベット文字に変換します。

制限事項: このオプションは、LOWERCASE, UPPERCASE オプションと同時に使用できません。

PROPER

エンコーディングに基づいて、次のデフォルトオプションを指定します。

- 日本語エンコーディング:
- HALF-KATAKANA, FULL-KATAKANA
- FULL-ALPHABET, HALF-ALPHABET
- LOWERCASE, UPPERCASE
- 韓国語エンコーディング:
- FULL-ALPHABET, HALF-ALPHABET
- 中国語エンコーディング:
- FULL-ALPHABET, HALF-ALPHABET
- 台湾語エンコーディング:
- FULL-ALPHABET, HALF-ALPHABET

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

指定されたオプションとデフォルトオプションに基づいて、入力文字列を変換します。KPROPCASE 関数は、中国語、日本語、韓国語、台湾語(CJKT)環境をサポートします。

例

次に、KPROPCASE 関数の使用例を示します。

```

length fullkana halfkana upper lower fullalpha $ 200;
length str1 str2 str3 str4 str5 str7 str8 $ 30 str6 $44;
lower = 'do-naxtutsu'; /* Doughnuts in Japanese Roman word. */
upper = 'DO-NAXTUTSU'; /* Doughnuts in Japanese word. */
fullkana = unicode('\u30C9\u30FC\u30CA\u30C3\u30C4');
halfkana = unicode('\uFF84\uFF9E\uFF70\uFF85\uFF6F\uFF82');
fullalpha = unicode('\uFF24\uFF2F\uFF0D\uFF2E\uFF21\uFF38\uFF34\uFF35\uFF34\uFF33\uFF35');
str1 = kpropcase(fullkana, 'full-katakana,half-katakana');
if (halfkana EQ trim(str1)) then
put str1= $hex14.;
str2 = kpropcase(halfkana, 'half-katakana, full-katakana');
if (fullkana EQ trim(str2)) then
put str2= $hex22.;
str3 = kpropcase(fullkana, 'katakana,romaji');
if (trim(str3) EQ upper) then
put str3= ;
str4 = kpropcase(upper, 'romaji,katakana');
if (trim(str4) EQ fullkana) then
put str4= $hex22.;
str5 = kpropcase(fullalpha, 'full-alphabet, half-alphabet');
if (trim(upper) EQ str5) then
put str5=;
str6 = kpropcase(upper, 'half-alphabet, full-alphabet');
if (trim(str6) EQ fullalpha) then
put str6= $hex46.;
str7 = kpropcase(lower, 'lowercase, uppercase');
if (trim(str7) EQ upper) then
put str7=;
str8 = kpropcase(upper, 'uppercase, lowercase');
if (trim(str8) EQ lower) then
put str8=;
RESULTS:
str1=C4DEB0C5AFC220
str2=8368815B83698362836320
str3=DO-NAXTUTSU
str4=8368815B83698362836320
str5=DO-NAXTUTSU
str6=8263826E817C826D826082778273827482738272827420
str7=DO-NAXTUTSU
str8=do-naxtutsu

```

KPROPCHAR 関数

特殊文字を標準文字に変換します。

カテゴリ: DBCS

構文

```
str=KPROPCHAR(<instr> )
```

必須引数

str

結果文字列。特殊文字を標準文字に変換します。

instr

入力データ文字列。

詳細

特殊文字を標準文字に変換します。KPROPCHAR 関数では、次の範囲の文字を変換します。

- かつこで囲まれた英数字値: \u2460 から \u24FF。参照先: <http://www.unicode.org/charts/PDF/U2460.pdf>
- Dingbats: \u2776 から \u2793。参照先: <http://www.unicode.org/charts/PDF/U2700.pdf>
- かつこで囲まれた CJK 文字と月: \u3200 から \u32FF。参照先: <http://www.unicode.org/charts/PDF/U3200.pdf>

例

次に、KPROPCHAR 関数の使用例を示します。

```
length in1 out1 $30 ;
in1=unicode ('\u2460\u2473\u277F\u325F');
out1=KPROPCHAR(in1);
put out1;
RESULTS:
(1) (20) (-10) (35)
```

KPROPDATA 関数

印刷不可文字を削除または変換します。

カテゴリ: DBCS

構文

```
str=KPROPDATA(<instr> (<option, input encode name, output encode name> ))
```

必須引数

str

変換された、セッションエンコーディングでのデータ文字列。

instr

入力データ文字列。

options

印刷不可文字の処理に関する指示を指定します。

UESC

Unicode エスケープ文字列(たとえば、\u0000\u1234)を使用して印刷不可文字を変換します。

TRIM

印刷不可文字を削除します。置換文字は使用しません。

BLANK または '**'**

各印刷不可文字を 1 バイトの空白で置換します。

QUESTION または '?'

印刷不可文字を 1 バイトの '?' で置換します。

HEX

印刷不可文字を 16 進表現(たとえば、0x810x82)で置換します。

TRUNCATE または **TRUNC**

最初の印刷不可文字が出現したときにデータ文字列を切り捨てます。

REMOVE

いずれかの印刷不可文字が検出された場合にデータ文字列を削除します。

NCR

Unicode にコードが存在する場合は、NCR 表示形式を使用して印刷不可文字をエンコードします。

input encode name

必要に応じて、入力データのエンコーディング名を指定します。入力エンコーディング名を指定しないと、KPROPDATA 関数では、現在の SAS セッションでエンコードされた文字列としてデータを処理します。SAS エンコーディング名についての詳細は、“データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値” (563 ページ)を参照してください。

output encode name

出力データのエンコーディング名を指定します。エンコーディング名を指定しないと、KPROPDATA 関数では、現在の SAS セッションエンコーディングとして出力を認識します。SAS エンコーディング名についての詳細は、“データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値” (563 ページ)を参照してください。

詳細

入力データ文字列を現在の SAS セッションエンコーディングに変換し、オプションに基づいて印刷不可文字を削除または置換します。

例

次に、KPROPDATA 関数の使用例を示します。

```
length instr $12;
length str1 str2 str3 str4 str5 str6 str7 str8 str9 str10$ 50;
instr = "534153"x||"ae"x || " System";
put instr;
str1 = kpropdata(instr);
put str1= +2 str1= $hex26.;
str2 = kpropdata(instr,'UESC');
put str2= +2 str2= $hex26.;;
str3 = kpropdata(instr, 'UESC','wlatin1');
put str3= +2 str3= $hex34.;
str4 = kpropdata(instr,'TRIM','wlatin1');
put str4= +2 str4= $hex26.;
```

```

str5 = kpropdata(instr,'BLANK', 'wlatin1');
put str5= +2 str5= $hex26.;
str6 = kpropdata(instr,'?', 'wlatin1');
put str6= +2 str6= $hex26.;
str7 = kpropdata(instr,'hex', 'wlatin1');
put str7= +2 str7= $hex26.;
str8 = kpropdata(instr,'TRUNC', 'wlatin1');
put str8= +2 str8= $hex26.;
str9 = kpropdata(instr,'REMOVE', 'wlatin1');
put str9= +2 str9= $hex26.;
str10 = kpropdata(instr,'NCR', 'wlatin1');
put str10= +2 str10= $hex26.;
RESULTS:
SAS? System
str1=SAS? System str1=534153AE2053797374656D2020
str2=SAS? System str2=534153AE2053797374656D2020
str3=SAS\uff6e System str3=5341535C75666636652053797374656D20
str4=SAS System str4=5341532053797374656D202020
str5=SAS System str5=534153202053797374656D2020
str6=SAS? System str6=5341533F2053797374656D2020
str7=SAS\xAE System str7=5341535C784145205379737465
str8=SAS str8=5341532020202020202020202020
str9= str9=202020202020202020202020202020
str10=SAS® System str10=53415326233137343B20537973

```

KREVERSE 関数

文字式を逆にします。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、REVERSE で、*SAS 関数と CALL ルーチン: リファレンス*を参照してください。

構文

KREVERSE(*argument*)

必須引数

argument

SAS 文字式を指定します。

詳細

制限と詳しい情報については、“*SAS 文字列関数の国際化の互換性*” (243 ページ)を参照してください。

KRIGHT 関数

末尾の DBCS の空白と SO/SI を削除し、文字式を右詰めにします。

カテゴリ: DBCS

ヒント: 参照先: “RIGHT Function” in *SAS Functions and CALL Routines: Reference*

構文

KRIGHT(*argument*)

必須引数

argument

SAS 文字式を指定します。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KRIGHT 関数は、末尾の空白を値の先頭に移動した引数を返します。引数の長さはありません。

関連項目:

関数:

- “[KCOMPRESS 関数](#)” (265 ページ)
- “[KLEFT 関数](#)” (269 ページ)
- “[KTRIM 関数](#)” (281 ページ)

KSCAN 関数

指定された単語を文字式から選択します。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、SCAN で、[SAS 関数と CALL ルーチン: リファレンス](#)を参照してください。

構文

KSCAN(*argument*,*n*<, *delimiters*>)

必須引数

argument

文字式を指定します。

n

KSCAN が文字式内で選択する単語の番号を算出する数式を指定します。

ヒント: *n* が負の場合、KSCAN は、文字式内の単語を文字列の最後から検索して選択します。*|n|*が文字式内の単語数よりも大きい場合、KSCAN は空白値を返します。

delimiters

KSCAN が文字式内で単語区切り文字として使用する文字を生成する、文字変数を指定します。

デフォルト: ASCII 環境で *delimiters* を省略すると、SAS は *blank(空白)* . < (+ & ! \$ *); ^ - /, % | を使用します。^ 文字のない ASCII 環境の場合、KSCAN はかわりに ~ 文字を使用します。

EBCDIC 環境で *delimiters* を省略すると、SAS は (*空白*) . < (+ | & ! \$ *); ^ - /, % | φ を使用します。

ヒント: *delimiters* を定数として表す場合は、*delimiters* を引用符で囲みます。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ) を参照してください。

区切り文字が文字列内で先頭の単語より前にあっても、KSCAN に影響はありません。複数の区切り文字が連続する場合、KSCAN はそれらをまとめて 1 つとして扱います。

KSTRCAT 関数

複数の文字式を連結します。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、CAT で、*SAS 関数と CALL ルーチン: リファレンス* を参照してください。

構文

KSTRCAT(*argument-1*, *argument-2*<, ... *argument-n*>)

必須引数

argument

1 バイトまたは 2 バイトの文字式を指定します。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ) を参照してください。

KSTRCAT では、複数の 1 バイトまたは 2 バイトの文字式を連結します。また、式の間にある不要な SO/SI のペアを削除します。

KSUBSTR 関数

引数から部分文字列を抽出します。

カテゴリ: DBCS

ヒント: 参照先: “SUBSTR (left of =) Function” in *SAS Functions and CALL Routines: Reference*

構文

KSUBSTR(*argument*, *position*<, *n*>)

必須引数**argument**

SAS 文字式を指定します。

position

文字の開始位置となる数値式を指定します。

n

抽出する部分文字列の長さを示す数値式を指定します。

操作: n が *argument* 内で *position* より後に残った式の長さよりも長い場合、式の残りの部分が抽出されます。

ヒント: n を省略すると、式の残りの部分が抽出されます。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KSUBSTR 関数は、式内の *argument* で指定された部分を返します。この部分は、*position* で指定された文字から始まり、 n 文字の長さです。

KSUBSTR で作成される変数の長さは、*argument* の長さから求めることができます。

関連項目:**関数:**

- “[KSUBSTRB 関数](#)” (279 ページ)

KSUBSTRB 関数

引数内の部分文字列のバイト位置に従って、引数から部分文字列を抽出します。

カテゴリ: DBCS

構文

KSUBSTRB(*argument*,*position*< n >)

必須引数**argument**

SAS 文字式を指定します。

position

開始文字位置をバイト単位で指定します。

n

抽出する部分文字列の長さをバイト単位で指定します。

操作: n が *argument* 内で *position* より後に残った式の長さ(バイト単位)よりも長い場合、式の残りの部分が抽出されます。

ヒント: n を省略すると、式の残りの部分が抽出されます。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KSUBSTRB 関数は、式内の *argument* で指定された部分を返します。この部分は、*position* で指定されたバイト単位で始まり、*n* バイトの長さです。

KSUBSTRB で作成される変数の長さは、*argument* の長さから求めることができます。

関連項目:

関数:

- “[KSUBSTR 関数](#)” (278 ページ)

KTRANSLATE 関数

文字式の特定の文字を置換します。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、TRANSLATE で、[SAS 関数と CALL ルーチン: リファレンス](#)を参照してください。

参照項目: z/OS での KTRANSLATE 関数

構文

KTRANSLATE(*source,to-1,from-1*<,...*to-n,from-n*>)

必須引数

source

元の文字値を含む SAS 式を指定します。

to

KTRANSLATE で代替文字として使用する文字を指定します。

from

KTRANSLATE で置換する文字を指定します。

操作: *to* および *from* の値は、文字対文字の関係で対応します。KTRANSLATE は、*from* の 1 文字を *to* の 1 文字に変換するという具合に処理を続けます。*to* の文字が *from* の文字より少ない場合、KTRANSLATE は *from* の余った文字を空白に変更します。*to* の文字が *from* の文字より多い場合、KTRANSLATE は *to* の余った文字を無視します。

注: 部の動作環境では、*to* と *from* の引数ペアが必要です。他の動作環境では、照合順序のセグメントが NULL の *from* 引数に置き換わります。詳細については、動作環境に関する SAS のドキュメントを参照してください。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KTRANSLATE を使用して、1 バイト文字式を 2 バイト文字式に変換したり、2 バイト文字式を 1 バイト文字式に変換したりできます。

KTRANSLATE が受け入れる *to* および *from* の引数ペアの最大数は、SAS を実行するときの動作環境に応じて異なります。短い引数のペアを数個使用する場合と、長い引数のペアを少数使用する場合とでは、機能的な違いはありません。

KTRIM 関数

末尾の DBCS の空白と SO/SI を文字式から削除します。

カテゴリ: DBCS

ヒント: 非 DBCS で相当する関数は、“TRIM Function” in *SAS Functions and CALL Routines: Reference* です。

構文

KTRIM(*argument*)

必須引数

argument

SAS 文字式を指定します。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KTRIM では、文字の引数をコピーし、すべての末尾の空白を削除して、その結果調整された引数を返します。引数が空白の場合、KTRIM は空白を返します。連結では末尾の空白は削除されないため、KTRIM は連結する場合に便利です。

KTRIM の結果を変数に割り当てる場合、受け取る変数の長さには影響しません。調整された値が受け取る変数の長さより短い場合、その変数への割り当て時に空白で値が埋め込まれます。

関連項目:

関数:

- “[KCOMPRESS 関数](#)” (265 ページ)
- “[KLEFT 関数](#)” (269 ページ)
- “[KRIGHT 関数](#)” (276 ページ)

KTRUNCATE 関数

マルチバイト文字を壊すことなく、文字列をバイト単位で、指定した長さに切ります。

カテゴリ: DBCS

構文

KTRUNCATE(*argument*, *number*, *length*)

必須引数

argument

SAS 文字式を指定します。

number

数値です。

length

整数です。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KTRUNCATE 関数では、完全な長さの *number*(倍精度浮動小数点型として保存)を *length* で指定したより小さいバイト数に切り捨て、切り捨てられたバイトを 0 で埋め込みます。切り捨てとその後の展開は、最初に完全な長さより短い数を保存した結果を複製し、次にそれを読み込みます。

KUPCASE 関数

引数内のすべての文字を大文字に変換します。

カテゴリ: DBCS

ヒント: 参照先: “UPCASE Function” in *SAS Functions and CALL Routines: Reference*

構文

KUPCASE(*argument*)

必須引数

argument

SAS 文字式を指定します。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KUPCASE 関数は、文字引数をコピーし、すべての小文字を大文字に変換して、変更された値を結果として返します。

KUPDATE 関数

文字値の内容を挿入、削除および置換します。

カテゴリ: DBCS

構文

KUPDATE(*argument*,*position*,*n*<, *characters-to-replace*>)

KUPDATE(*argument*,*position*<,*n*> , *characters-to-replace*)

必須引数

argument

文字変数を指定します。

position

文字の開始位置となる数値式を指定します。

n

置き換える部分文字列の長さを示す数値式を指定します。

制限事項:

n は、*argument* 内で、*position* 以後に残った式の長さ以下である必要があります。

n の指定はオプションですが、この関数で *n* と *characters-to-replace* を両方とも省略することはできません。

ヒント: *n* を省略すると、*characters-to-replace* のすべての文字を使用して *argument* の値を置き換えます。

characters-to-replace

argument の内容を置き換える文字式を指定します。

制限事項: *characters-to-replace* の指定はオプションですが、この関数で *characters-to-replace* と *n* を両方とも省略することはできません。

ヒント: 文字のリテラル文字列を引用符で囲みます。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KUPDATE 関数は、*argument* の値を *characters-to-replace* の式で置き換えます。KUPDATE は、*position* で指定された開始文字から *n* 文字分を置き換えます。

注: NLSCOMPATMODE システムオプションを ON に設定する場合、パラメータ *characters-to-replace* では、以前の SAS リリースに基づいてデータが処理されます。NLSCOMPATMODE をオフにすると、*characters-to-replace* では 9.2 の機能が使用されます。例については、次の表を参照してください。

ステートメント	結果
NLSCOMPATEMODE kkupdate ("123456", 2,3);	156
NLSCOMPATEMODE kupdate ("123456", 2,3,"abcd");	1abcd56
NONLSCOMPATEMODE kupdate ("123456", 2,3);	1 56
NONLSCOMPATEMODE kupdate ("123456", 2,3,"abcd");	1abc56

関連項目:**関数:**

- “[KUPDATEB 関数](#)” (284 ページ)

システムオプション:

- “[NLSCOMPATMODE システムオプション: z/OS](#)” (473 ページ)

KUPDATEB 関数

引数内の文字値のバイト位置に従って、文字値の内容を挿入、削除および置換します。

カテゴリ: DBCS

構文

KUPDATEB(*argument*,*position*,*n*<, *characters-to-replace*>)

KUPDATEB(*argument*,*position* <, *n*> , *characters-to-replace*)

必須引数***argument***

文字変数を指定します。

position

開始文字位置をバイト単位で指定します。

n

バイト単位に置き換えられる部分文字列の長さを指定します。

制限事項:

n は、*position* の後の *argument* に残る式の長さ(バイト)よりも大きくできません。

n の指定はオプションですが、この関数で *n* と *characters-to-replace* を両方とも省略することはできません。

ヒント: *n* を省略すると、*characters-to-replace* のすべての文字を使用して *argument* の値を置き換えます。

characters-to-replace

argument の内容を置き換える文字式を指定します。

制限事項: *characters-to-replace* の指定はオプションですが、この関数で *characters-to-replace* と *n* を両方とも省略することはできません。

ヒント: 文字のリテラル文字列を引用符で囲みます。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KUPDATEB 関数は、*argument* の値を *characters-to-replace* の式で置き換えます。KUPDATEB は、*position* で指定したバイト単位で開始する *n* バイト単位を置き換えます。

関連項目:

関数:

- [“KUPDATE 関数” \(282 ページ\)](#)

KVERIFY 関数

式に固有の最初の文字の位置を返します。

カテゴリ: DBCS

ヒント: 参照先: “VERIFY Function” in *SAS Functions and CALL Routines: Reference*

構文

`KVERIFY(source,excerpt-1<,...excerpt-n>)`

必須引数

source

SAS 文字式を指定します。

excerpt

SAS 文字式を指定します。複数の *excerpt* を指定する場合は、カンマで区切ります。

詳細

制限と詳しい情報については、“[SAS 文字列関数の国際化の互換性](#)” (243 ページ)を参照してください。

KVERIFY 関数は、いずれの *excerpt* 内にも存在しない文字のうち、最初の文字の *source* 内での位置を返します。KVERIFY は、*source* 内のすべての文字が、少なくとも 1 つの *excerpt* 内にあることを確認すると、0 を返します。

NLDATE 関数

SAS 日付値を、日付のディスクリプタを使用して指定したロケールの日付値に変換します。

カテゴリ: 日付と時間

構文

`NLDATE(date,descriptor)`

必須引数

date

SAS 日付値を指定します。

descriptor

日付と時間の出力形式を指定する、変数または式です。次のディスクリプタでは、大文字と小文字が区別されます。

- #
先頭の 0 を出力結果から削除します。
- %%
%文字を指定します。
- %a
省略形の曜日のディスクリプタを指定します。省略形の曜日のディスクリプタの範囲は、Mon から Sun です。
- %A
完全な曜日のディスクリプタを指定します。完全な曜日のディスクリプタの範囲は、Monday から Sunday です。
- %b
省略形の月のディスクリプタを指定します。省略形の月のディスクリプタの範囲は、Jan から Dec です。
- %B
完全な月のディスクリプタを指定します。完全な月のディスクリプタの範囲は、January から December です。
- %C
完全な月のディスクリプタを指定し、空白埋め込みを使用します。完全な月のディスクリプタの範囲は、January から December です。
- %d
日付のディスクリプタを指定し、0 埋め込みを使用します。日付修飾子の範囲は、01 から 31 です。
- %e
日付のディスクリプタを指定し、空白埋め込みを使用します。日付のディスクリプタの範囲は、01 から 31 です。
- %F
完全な曜日のディスクリプタを指定し、空白埋め込みを使用します。完全な曜日のディスクリプタの範囲は、Monday から Sunday です。
- %j
通算日のディスクリプタを 10 進数で指定し、先頭に 0 を使用します。通算日のディスクリプタの範囲は、1 から 366 です。
- %m
月のディスクリプタを指定し、0 埋め込みを使用します。月のディスクリプタの範囲は、01 から 12 です。
- %o
月のディスクリプタを指定します。月のディスクリプタの範囲は 1 から 12 で、空白埋め込みを使用します。
- %u
月曜日から日曜日を表す 1 から 7 の範囲の番号で、曜日のディスクリプタを指定します。
- %U
年内の週数を使用してディスクリプタ値を SAS 日付値として計算することで、通算週のディスクリプタを指定します(日曜日を週の開始日と見なします)。週数の値は、0 から 53 の範囲の 10 進数で表し、先頭に 0 を使用します。最大値は 53 です。
- %V
ディスクリプタ値を SAS 日付値として計算することで、通算週のディスクリプタを指定します。週数の値は、01 から 53 の範囲の 10 進数で表し、先頭に 0 を

使用します。最大値は 53 です。各週は月曜日から始まり、年の第 1 週は 1 月 4 日と年の最初の木曜日の両方を含む週です。1 月の最初の月曜日が 2 日、3 日または 4 日の場合、それより前の日は前年の最後の週に組み込まれません。

%w

日曜日から土曜日を表す 0 から 6 の範囲の番号で、曜日のディスクリプタを指定します。

%W

年内の週数を使用してディスクリプタ値を SAS 日付値として計算することで、通算週のディスクリプタを指定します(月曜日を週の開始日と見なします)。週数の値は、0 から 53 の範囲の 10 進数で表し、先頭に 0 を使用します。最大値は 53 です。

%y

年(2 桁)の修飾子を指定します。年のディスクリプタの範囲は、00 から 99 です。

%Y

年(4 桁)のディスクリプタを指定します。年のディスクリプタの範囲は、1970 から 2069 です。

詳細

NLDATE 関数では、SAS 日付値を、日付のディスクリプタを使用して指定したロケールの日付値に変換します。

例

SAS 日付値から作成されたログファイル名の例を次に示します。

ステートメント	結果
<pre>options locale=English_Unitedstates; logfile=nldate('24Feb2003'd, '%B-%d.log'); put logfile;</pre>	February-24.log
<pre>options locale=German_Germany; logfile=nldate('24Feb2003'd, '%B-%d.log'); put logfile;</pre>	Februar-24.log

SAS 日付値から作成された曜日名の例を次に示します。

ステートメント	結果
	----+-----1-----+
<pre>options locale=English_unitedstates; weekname=nldate('24Feb2003'd, '%A'); put weekname;</pre>	Monday

ステートメント	結果
<pre>options locale=German_Germany; weekname=nldate('24Feb2003'd, '%A'); put weekname;</pre>	Montag

関連項目:

出力形式:

- “[NLDATEw. 出力形式](#)” (102 ページ)

NLDATM 関数

SAS 日時値を、日時形式のディスクリプタを使用して指定したロケールの時間値に変換します。

カテゴリ: 日付と時間

構文

NLDATM(*datetime*, *descriptor*)

必須引数

datetime

SAS 日時値を指定します。

descriptor

日付と時間の出力形式を指定する、変数または式です。次のディスクリプタでは、大文字と小文字が区別されます。

#

先頭の 0 を出力結果から削除します。

%%

%文字を指定します。

%a

省略形の曜日のディスクリプタを指定します。省略形の曜日のディスクリプタの範囲は、Mon から Sun です。

%A

完全な曜日のディスクリプタを指定します。完全な曜日のディスクリプタの範囲は、Monday から Sunday です。

%b

省略形の月のディスクリプタを指定します。省略形の月のディスクリプタの範囲は、Jan から Dec です。

%B

完全な月のディスクリプタを指定します。完全な月のディスクリプタの範囲は、January から December です。

- %c**
完全な月のディスクリプタを指定し、空白埋め込みを使用します。完全な月のディスクリプタの範囲は、January から December です。
- %d**
日付のディスクリプタを指定し、0 埋め込みを使用します。日付のディスクリプタの範囲は、01 から 31 です。
- %e**
日付のディスクリプタを指定し、空白埋め込みを使用します。日付のディスクリプタの範囲は、01 から 31 です。
- %F**
完全な曜日のディスクリプタを指定し、空白埋め込みを使用します。完全な曜日のディスクリプタの範囲は、Monday から Sunday です。
- %H**
24 時間表示の時間のディスクリプタを指定します。時間のディスクリプタの範囲は、00 から 23 です。
- %I**
12 時間表示の時間のディスクリプタを指定します。時間のディスクリプタの範囲は、01 から 12 です。
- %j**
通算日のディスクリプタを 10 進数で指定し、先頭に 0 を使用します。通算日のディスクリプタの範囲は、1 から 366 です。
- %m**
月のディスクリプタを指定し、0 埋め込みを使用します。月のディスクリプタの範囲は、01 から 12 です。
- %M**
分のディスクリプタを指定します。分のディスクリプタの範囲は、00 から 59 です。
- %o**
月のディスクリプタを指定し、空白埋め込みを使用します。月のディスクリプタの範囲は、1 から 12 です。
- %p**
午前または午後のディスクリプタを指定します。
- %S**
秒のディスクリプタを指定します。秒のディスクリプタの範囲は、00 から 59 です。
- %u**
月曜日から日曜日を表す 1 から 7 の範囲の番号で、曜日のディスクリプタを指定します。
- %U**
年内の週数を使用してディスクリプタ値を SAS 日付値として計算することで、通算週のディスクリプタを指定します(日曜日を週の開始日と見なします)。週数の値は、0 から 53 の範囲の 10 進数で表します。先頭に 0 を使用し、最大値は 53 です。
- %V**
ディスクリプタ値を SAS 日付値として計算することで、通算週のディスクリプタを指定します。週数の値は、01 から 53 の範囲の 10 進数で表します。先頭に 0 を使用し、最大値は 53 です。各週は月曜日から始まり、年の第 1 週は 1 月 4 日と年の最初の木曜日の両方を含む週です。1 月の最初の月曜日が 2 日、

3 日または 4 日の場合、それより前の日は前年の最後の週に組み込まれません。

%w

日曜日から土曜日を表す 0 から 6 の範囲の番号で、曜日のディスクリプタを指定します。

%W

年内の週数を使用してディスクリプタ値を SAS 日付値として計算することで、通算週のディスクリプタを指定します(月曜日を週の開始日と見なします)。週数の値は、0 から 53 の範囲の 10 進数で表します。先頭に 0 を使用し、最大値は 53 です。

%y

年(2 桁)のディスクリプタを指定します。年のディスクリプタの範囲は、00 から 99 です。

%Y

年(4 桁)のディスクリプタを指定します。年のディスクリプタの範囲は、1970 から 2069 です。

詳細

NLDATM 関数では、SAS 日時値を、日時のディスクリプタを使用して指定したロケールの日時値に変換します。

例

SAS 日時値から作成された時間(午前または午後)の例を次に示します。

ステートメント	結果
	-----1-----+
options locale=English; time_ampm=nldatm('24Feb2003:12:39:43'dt, '%I%p'); put time_ampm;	12PM
options locale=German; time_ampm=nldatm('24Feb2003:12:39:43'dt, '%I%p'); put time_ampm;	12nachm

関連項目:

出力形式:

- [“NLDATMw. 出力形式” \(111 ページ\)](#)

NLTIME 関数

SAS 時間値または SAS 日時値を、NLTIME ディスクリプタを使用して指定したロケールの時間値に変換します。

カテゴリ: 日付と時間

構文

NLTIME(*time|datetime,descriptor,startpos*)

必須引数

time

SAS 時間値を指定します。

datetime

SAS 日時値を指定します。

descriptor

ディスクリプタの値を指定する、変数または式です。大文字または小文字で次のディスクリプタを入力できます。

#

先頭の 0 を出力結果から削除します。

%%

%文字を指定します。

%H

24 時間表示の時間のディスクリプタを指定します。時間のディスクリプタの範囲は、00 から 23 です。

%I

12 時間表示の時間のディスクリプタを指定します。時間のディスクリプタの範囲は、01 から 12 です。

%M

分の修飾子を指定します。分のディスクリプタの範囲は、00 から 59 です。

%P

午前または午後のディスクリプタを指定します。

%S

秒のディスクリプタを指定します。秒のディスクリプタの範囲は、00 から 59 です。

startpos

検索を開始する位置を指定し、検索の方向を指定する整数です。

詳細

NLTIME 関数では、SAS 時間値または SAS 日時値を、時間のディスクリプタを使用して指定したロケールの時間値に変換します。

例

SAS 時間から作成された午前または午後の時間の例を次に示します。

ステートメント	結果
	-----1-----+
options locale=English;	00 PM
time_ampm=nltime('12:39:43't,'%i%p');	
put time_ampm;	

ステートメント	結果
<pre>options locale=German; time_ampm=nltime('12:39:43't, '%i%p'); put time_ampm;</pre>	00 nachm

関連項目:

出力形式:

- [“NLTIMEw. 出力形式” \(205 ページ\)](#)

SASMSG 関数

データセットからメッセージを指定します。返されるメッセージは現在のロケールと指定されたキーに基づきます。

カテゴリ: ロケール

構文

SASMSG (*BASENAME*", "*KEY*", <<"*QUOTE*"|"*DQUOTE*"|"*NOQUOTE*">>
<, "*substitution 1*", ..., "*substitution 7*">>)

必須引数

BASENAME

メッセージを含むデータセットの名前

KEY

メッセージキー

注: 無効なキー名を指定すると、キー名が返されます。

QUOTE|DQUOTE|NOQUOTE

メッセージテキストと代入文字列に追加する括弧のタイプを指定します。

デフォルト: DQUOTE

substitution

文字列代入 文字列代入の最大件数は 7 です。

詳細

SAS メッセージデータセットは 7 ビット ASCII データセットでなければなりません。7 ビット ASCII エンコーディングで表現できない文字は '\uxxxx' のような Unicode エスケープ出力形式で表現されます。'xxxx' は、文字の Unicode 値の 10 進数表現です。

SASMSG 関数で使用するデータセットは、この関数の使用専用で作成してください。データセットは次の変数を含まなければなりません。

#	変数名	タイプ	長さ	説明
1	locale	char	5	メッセージの言語

#	変数名	タイプ	長さ	説明
2	key	char	60	メッセージを認識するキー
3	lineno	num	5	メッセージのライン番号(逆順)
4	text	text	1,200	メッセージのテキスト

データセットは、*locale*、*key*、および *lineno* を使用して並べ替える必要があります。変数 *lineno* は降順でなければなりません。locale と key の複合インデックスを定義する必要があります。並べ替えとインデックス付きのデータセットを作成するサンプルプログラムは次のようになります。

```
%let basename=MyProduct;

proc sort data=t.&basename;
  by locale key descending lineno;
run;

proc datasets lib=t
  memtype=data;
  modify &basename;
  index create indx=(LOCALE KEY);
run;
quit;
```

返されるメッセージは LOCALE システムオプションに基づきます。LOCALE オプションは *ll RR* で表されます。ll は 2 文字の言語コードを表し、PR は 2 文字の地域コードを表します。一致するものがない場合、関数は言語のみの一致を探します。locale と key のペアが見つからない場合、関数はデフォルトの英語(en)を使用します。英語(en)のキーが存在しない場合、キー名が返されます。

出力形式は変更することができます。出力形式コード %s を使用して、文字列代入を使用できます。代入順序を変更する事ができます。英語以外の言語へのメッセージの翻訳では、代入順序の変更が必要な場合があります。引数番号 #nn を指定することにより、出力文字列内で順序を変えることができます。nn は代入リストの引数番号になります。次の例では順序について説明します。

ステートメント	結果
<pre>msg = sasmsg ("nls.mymsg", "IN_CD_LOG", "noquote", "cat", "dog"); IN_CD_LOGINFO = My %#1s. Your %#2s</pre>	<pre>msg= My cat. Your dog.</pre>
<pre>IN_CD_LOGINFO = My %#2s. Your %#1s</pre>	<pre>msg= My dog. Your cat.</pre>

SASMSG 関数は %SYSFUNC マクロ関数と一緒にオープンコードマクロで使用されません。

%SYSFUNC マクロが呼ぶ関数に渡される引数は引用符で囲むはいけませんが、%SYSFUNC 外で SASMSG 関数に渡される引数は引用符で囲まなければなりません。

SASMSG 関数が%SYSFUNC マクロ関数と一緒に使われる場合、返される文字列は%NRBQUOTE 関数を使用して囲まれます。

例

例 1

次の例で SASMSG の出力形式の機能を説明します。

```
%macro demo_sasmsg;
data _null_;
msg = sasmsg("nls.mymsg", "IN_APW_SAVE_OK", "noquote");
put msg=;
run;
%mend demo_sasmsg;
```

SAS ステートメント	結果
options locale = en_US; %demo_sasmsg ;	msg=The Access Control key was successfully saved.
options locale = es_ES; %demo_sasmsg ;	msg=La clave de control de acceso se ha guardado.
options locale = french_France; %demo_sasmsg ;	msg=La clé de contrôle d'accès a bien été enregistrée.

例 2

次の例でオープンマクロの機能を説明します。

```
%MACRO PRT(loc,tb,key);
option locale=&loc;
%PUT %SYSFUNC(SASMSG(&tb,&key) );
%MEND PRT;
```

SAS ステートメント	結果
%PRT(en_US,&TABLEID,IN_EDIT)	"Edit"
%PRT(es_ES,&TABLEID,IN_EDIT)	"Editor"
%PRT(fr_FR,&TABLEID,IN_EDIT)	"Modifier"

SASMSG 関数

データセットからメッセージを指定します。メッセージは、指定されたロケール値と指定されたキー値に基づきます。

カテゴリ: ロケール

構文

```
SASMSGSL(("BASENAME", "KEY", "LOCALE", <<"Q"|"D"|"N">
<, "substitution 1", ..., "substitution 6">>)
```

必須引数

BASENAME

メッセージを含むデータセットの名前

KEY

メッセージキー

注: 無効なキー名を指定すると、キー名が返されます。

LOCALE

posix ロケール値 (ll_RR).

QUOTE|DQUOTE|NOQUOTE

メッセージテキストと代入文字列に追加するかっこのタイプを指定します。

デフォルト: DQUOTE

substitution

文字列代入 文字列代入の最大件数は 6 です。

詳細

SAS メッセージデータセットは 7 ビット ASCII データセットでなければなりません。7 ビット ASCII エンコーディングで表現できない文字は '\uxxxx' のような Unicode エスケープ出力形式で表現されます。'xxxx' は、文字の Unicode 値の 10 進数表現です。

SASMSGSL 関数で使用するデータセットは、この関数の使用専用で作成してください。データセットは次の変数を含まなければなりません。

#	変数名	タイプ	長さ	説明
1	locale	char	5	メッセージの言語
2	key	char	60	メッセージを認識するキー
3	lineno	num	5	メッセージのライン番号 (逆順)
4	text	text	1200	メッセージのテキスト

データセットは、*locale*、*key*、および *lineno* を使用して並べ替える必要があります。変数 *lineno* は降順でなければなりません。locale と key の複合インデックスを定義する必要があります。並べ替えとインデックス付きのデータセットを作成するサンプルプログラムは次のようになります。

```
%let basename=MyProduct;

proc sort data=t.&basename;
  by locale key descending lineno;
run;
```

```
proc datasets lib=t
  memtype=data;
  modify &basename;
  index create indx=(LOCALE KEY);
run;
quit;
```

返されるメッセージは LOCALE システムオプションに基づきます。LOCALE オプションは *ll RR* で表されます。*ll* は 2 文字の言語コードを表し、*PR* は 2 文字の地域コードを表します。一致するものがない場合、関数は言語のみの一致を探します。locale と key のペアが見つからない場合、関数はデフォルトの英語(en)を使用します。英語(en)のキーが存在しない場合、キー名が返されます。

出力形式は変更することができます。出力形式コード%s を使用して、文字列代入を使用できます。代入順序を変更することができます。英語以外の言語へのメッセージの翻訳では、代入順序の変更が必要な場合があります。引数番号#*nn* を指定することにより、出力文字列内で順序を変えることができます。*nn* は代入リストの引数番号になります。次の例では順序の変更について説明します。

ステートメント	結果
<pre>msg = sasmsgl ("nls.mymsg", "IN_CD_LOG", "en_US", "N", "cat", "dog"); IN_CD_LOGINFO = My %#1s. Your %#2s</pre>	<pre>msg= My cat. Your dog.</pre>
<pre>IN_CD_LOGINFO = My %#2s. Your %#1s</pre>	<pre>msg= My dog. Your cat.</pre>

SASMSGGL 関数は%SYSFUNC マクロ関数と一緒にオープンコードマクロで使用できません。

%SYSFUNC マクロが呼ぶ関数に渡される引数は引用符で囲んではいけません、%SYSFUNC の外の SASMSGGL 関数に渡される引数は引用符で囲まなければなりません。

SASMSGGL 関数が%SYSFUNC マクロ関数と一緒に使われる場合、返される文字列は%NRBQUOTE 関数を使用して囲まれます。

例

例 1

次の例で SASMSGGL の出力形式の機能を説明します。

ステートメント	結果
<pre>sasmsgl("nls.mymsg", "IN_APW_SAVE_OK", "e The Access Control key was successfully saved.</pre>	
<pre>sasmsgl("nls.mymsg", "IN_APW_SAVE_OK", "e La clave de control de acceso se ha guardado.</pre>	
<pre>sasmsgl("nls.mymsg", "IN_APW_SAVE_OK", "f La clé de contrôle d'accès a bien été enregistrée.</pre>	

例 2

次の例でオープンマクロの機能を説明します。

SAS ステートメント	結果
<code>%PUT %SYSFUNC(SASMSG1(NLS.MYDS, IN_ASD_LA "Edit"</code>	
<code>%PUT %SYSFUNC(SASMSG1(NLS.MYDS, IN_ASD_LA "Editor"</code>	
<code>%PUT %SYSFUNC(SASMSG1(NLS.MYDS, IN_ASD_LA "Modifier"</code>	

SORTKEY 関数

言語並べ替えキーを作成します。

カテゴリ: ロケール

構文

`sortKey(string, <locale, strength, case, numeric, order>)`

必須引数

string
文字式

locale
POSIX 名(ja_JP)形式でロケール名を指定します。ロケール名と POSIX 値のリストについては、[表 18.1 \(549 ページ\)](#)を参照してください。

strength
強度の値は、照合レベルに関連付けられます。5 つの照合レベル値があります。次の表に、5 つのレベルの情報を示します。強度のデフォルト値は、ロケールに関連付けられます。

値	照合の種類	説明
PRIMARY または P	PRIMARY では、基本文字間の差異を指定します("a"<"b"など)。	この差異は最強です。たとえば、辞書は基本文字により異なるセクションに分割されます。
SECONDARY または S	文字のアクセント記号が第 2 差異となります("as"<"às"<"at"など)。	文字間のその他の差異も、言語に応じて第 2 差異と考慮される場合があります。文字列内に第 1 差異がある場合は、第 2 差異は無視されます。
TERTIARY または T	大文字と小文字の差異は、第 3 レベルで区別されます("ao"<"Ao"<"aò"など)。	例として、通常の仮名と小文字の仮名の違いがあります。文字列内に第 1 または第 2 差異がある場合は、第 3 差異は無視されます。

QUATERNARY または Q	レベル 1 から 3 で句読点が無視される場合、句読点の有無で単語を区別するために追加レベルが使用されます("ab"<"a-b"<"aB"など)。	第 1、第 2 または第 3 差異がある場合は、この差異は無視されます。句読点の無視が必須の場合や日本語テキストを処理する場合は、第 4 レベルを使用する必要があります。
IDENTICAL ま たは I	他のすべてのレベルが等しい場合、最終的な決定をするために同一レベルが使用されます。各文字列の正規化形式 D の Unicode コードポイント値がこのレベルで比較され、レベル 1 から 4 で違いがないことが確認されます。	たとえば、ヘブライ語の詠唱マークのみがこのレベルで区別されます。2 つの文字列間でコードポイント値のみが異なることはほとんどないため、このレベルは必要最小限に使用します。

case order

大文字と小文字で並べ替えます。この引数は、TERTIARY、QUATERNARY または IDENTICAL レベルでのみ有効です。次の表に、case order 引数の値と情報を示します。

値	説明
UPPER または U	大文字、小文字の順序で並べ替えます。
LOWER または L	小文字、大文字の順序で並べ替えます。

numeric collation

数を表す文字のかわりに数値で番号を並べ替えます。

値	説明
NUMERIC または N	数値で番号(整数)を並べ替えます。たとえば、"8 Main St."は"45 Main St."より前になります。

collation order

collation の値には、Phonebook と Traditional という 2 種類があります。照合値を選択しない場合、ユーザーのロケールのデフォルト照合が選択されます。次の表に、詳細を示します。

値	説明
PHONEBOOK または P	文字の電話帳スタイルの並べ替えを指定します。PHONEBOOK はドイツ語でのみ選択します。
TRADITIONAL または T	文字の従来のスタイルの並べ替えを指定します。TRADITIONAL はスペイン語でのみ選択します。

詳細

SORTKEY 関数により、データの言語並べ替えキーが作成されます。少なくとも 1 つの引数を入力する必要があります。キーを受け取る変数の長さが十分でない場合、データが切り捨てられて警告が表示されます。

locale

ロケール値には POSIX 名(LL_RR)を使用します。LL は 2 文字の言語コードを表し、RR は 2 文字の地域コードを表します。たとえば、en_US は英語で、米国の POSIX 名です。en は英語を表し、US は米国を表します。ロケール値が指定されていない場合、セッションロケールが使用されます。

strength

strength 引数は、アクセント記号または大文字、小文字が照合または一致テキストに影響するかどうかを決定します。strength に値が指定されていない場合、ロケールにより値が決まります。次の値を strength に指定できます。

PRIMARY

基本文字が含まれます。たとえば、文字 A、a、Å のすべてが同様に処理されます。

SECONDARY

データを PRIMARY と同様に処理し、さらにアクセント記号を処理します。文字 A と a は同等に処理され、Å はアクセント記号付き文字として処理されません。

TERTIARY

データを SECONDARY と同様に処理し、さらに大文字、小文字を処理します。たとえば、A、a、Å はすべて異なる文字として処理されます。

QUATERNARY

データを TERTIARY と同様に処理し、さらに句読点を処理します。

IDENTICAL

データを QUATERNARY と同様に処理し、さらにコードポイントを処理します。

case order

大文字または小文字でデータを並べ替えるように指定します。次の表に、UPPER 値または LOWER 値を指定した例を示します。

UPPER	LOWER
Aztec	aztec
aztec	Aztec
Mars	mars
mars	Mars

collation order

照合順序値 PHONEBOOK は、ロケールがドイツ語でない場合は無視されます。

照合順序値 TRADITIONAL は、ロケールがスペイン語でない場合は無視されません。

他のロケールでは警告メッセージが表示されます。

SETLOCALE 関数

現在の SAS ロケールのロケールキーを指定します。

カテゴリ: ロケール

構文

SAS ロケールの設定

SETLOCALE(*sas_locale*)

単一ロケール要素のカスタマイズ

SETLOCALE(*key, value*)

単一ロケール要素のカスタマイズ

SETLOCALE(*category_name, sas_locale*)

必須引数

sas_locale

SAS 名または Posix 名を使用して SAS ロケール名を指定します。ロケールのエイリアスを指定する事もできます。

key

SAS ロケール要素キー 詳細セクションの要素キーのリストを参照してください。

value

ロケール要素の値を指定します。

category_name

カテゴリ名を指定します。

- LC_TIME
- LC_MONETARY
- LC_NUMERIC
- LC_ALL

詳細

次のロケール要素を変更することができます。 *key* の値は *max length* の値より小さくなくてはなりません。 *type* に次の値を指定できます。

- 0 文字列
- 1 符号なし整数 二重引用符を使用しなくてはなりません。

ロケール要素キー	最大長	タイプ	カテゴリ
DATESTYLE	3	0	
PAPERSIZE	8	0	
FTITLE	512	0	
FTEXT	512	0	
SIMFONT	512	0	
SORTSEQ	8	0	
MESSAGES	8	0	
FORMATNAME_DA TE	512	0	

ロケール要素キー	最大長	タイプ	カテゴリ
FORMATNAME_DATE TIME	512	0	
FORMATNAME_TIME	512	0	
FORMATNAME_NUMERIC	512	0	
FORMATNAME_PERCENT	512	0	
FONT_SERIF	32	0	
FONT_SANSERIF	32	0	
FONT_CURSIVE	32	0	
FONT_FANTASY	32	0	
FONT_MONOSPAC E	32	0	
BRUSH	32	0	
SIMPLEX	32	0	
COMPLEX	32	0	
SWISS	32	0	
ITALIC	32	0	
DATE_FORMAT	512	0	LC_TIME
DATE_SHORT_FOR MAT	512	0	LC_TIME
DATETIME_AMP M_FORMAT	512	0	LC_TIME
DATETIME_FOR MAT	512	0	LC_TIME
DATETIME_SHORT _FORMAT	512	0	LC_TIME
DATETIME_WEEK_ FORMAT	512	0	LC_TIME
DATETIME_WEEK_ SHORT_FORMAT	512	0	LC_TIME
TIME_AMP M_FORMAT	512	0	LC_TIME
TIME_FORMAT	512	0	LC_TIME

ロケール要素キー	最大長	タイプ	カテゴリ
DATE_WEEK_FOR MAT	512	0	LC_TIME
DATE_WEEK_SHO RT_FORMAT	512	0	LC_TIME
DATE_YYMM_FOR MAT	512	0	LC_TIME
DATE_YYMM_SHO RT_FORMAT	512	0	LC_TIME
DATE_MMDD_FOR MAT	512	0	LC_TIME
DATE_MMDD_SHO RT_FORMAT	512	0	LC_TIME
DATE_YEAR_FOR MAT	512	0	LC_TIME
DATE_YEAR_SHOR T_FORMAT	512	0	LC_TIME
DATE_YYQQ_FOR MAT	512	0	LC_TIME
DATE_YYQQ_SHO RT_FORMAT	512	0	LC_TIME
DATE_YYWW_FOR MAT	512	0	LC_TIME
DATE_YYWW_SHO RT_FORMAT	512	0	LC_TIME
DATE_SEP	8	0	LC_TIME
ABMON01	512	0	LC_TIME
ABMON02	512	0	LC_TIME
ABMON03	512	0	LC_TIME
ABMON04	512	0	LC_TIME
ABMON05	512	0	LC_TIME
ABMON06	512	0	LC_TIME
ABMON07	512	0	LC_TIME
ABMON08	512	0	LC_TIME
ABMON09	512	0	LC_TIME
ABMON10	512	0	LC_TIME

ロケール要素キー	最大長	タイプ	カテゴリ
ABMON11	512	0	LC_TIME
ABMON12	512	0	LC_TIME
MON01	512	0	LC_TIME
MON02	512	0	LC_TIME
MON03	512	0	LC_TIME
MON04	512	0	LC_TIME
MON05	512	0	LC_TIME
MON06	512	0	LC_TIME
MON07	512	0	LC_TIME
MON08	512	0	LC_TIME
MON09	512	0	LC_TIME
MON10	512	0	LC_TIME
MON11	512	0	LC_TIME
MON12	512	0	LC_TIME
ABDAY1	512	0	LC_TIME
ABDAY2	512	0	LC_TIME
ABDAY3	512	0	LC_TIME
ABDAY4	512	0	LC_TIME
ABDAY5	512	0	LC_TIME
ABDAY6	512	0	LC_TIME
ABDAY7	512	0	LC_TIME
DAY1	512	0	LC_TIME
DAY2	512	0	LC_TIME
DAY3	512	0	LC_TIME
DAY4	512	0	LC_TIME
DAY5	512	0	LC_TIME
DAY6	512	0	LC_TIME
DAY7	512	0	LC_TIME
AM	512	0	LC_TIME

ロケール要素キー	最大長	タイプ	カテゴリ
PM	512	0	LC_TIME
ABQTR1	512	0	LC_TIME
ABQTR2	512	0	LC_TIME
ABQTR3	512	0	LC_TIME
ABQTR4	512	0	LC_TIME
QTR1	512	0	LC_TIME
QTR2	512	0	LC_TIME
QTR3	512	0	LC_TIME
QTR4	512	0	LC_TIME
INT_CURRENCY_SYMBOL	3	0	LC_MONETARY
CURRENCY_SYMBOL	32	0	LC_MONETARY
MON_DECIMAL_POINT	8	0	LC_MONETARY
MON_THOUSANDS_SEP	8	0	LC_MONETARY
MON_GROUPING	3	1	LC_MONETARY
MON_POSITIVE_SIGN	8	0	LC_MONETARY
MON_NEGATIVE_SIGN	8	0	LC_MONETARY
MON_INT_FRAC_DIGITS	3	1	LC_MONETARY
MON_FRAC_DIGITS	3	1	LC_MONETARY
MON_P_CS_PRECEDES	3	1	LC_MONETARY
MON_P_SEP_BY_SPACE	3	1	LC_MONETARY
MON_P_SIGN_POSITION	3	1	LC_MONETARY
MON_N_SIGN_POSITION	3	1	LC_MONETARY
DECIMAL_POINT	1	0	LC_NUMERIC

ロケール要素キー	最大長	タイプ	カテゴリ
THOUSANDS_SEP	1	0	LC_NUMERIC
GROUPING	3	1	LC_NUMERIC
POSITIVE_SIGN	8	0	LC_NUMERIC
NEGATIVE_SIGN	8	0	LC_NUMERIC
P_CS_PRECEDES	3	1	LC_NUMERIC
P_SEP_BY_SPACE	3	1	LC_NUMERIC
N_CS_PRECEDES	3	1	LC_NUMERIC
P_SEP_BY_SPACE	3	1	LC_NUMERIC
N_CS_PRECEDES	3	1	LC_NUMERIC
N_SEP_BY_SPACE	3	1	LC_NUMERIC
P_SIGN_POSN	3	1	LC_NUMERIC
N_SIGN_POSN	3	1	LC_NUMERIC
HEIGHT	3	1	
WIDTH	3	1	

例

例 1

次の例では、SETLOCALE 関数は日本語(jp_JP)のロケールを指定します。SETLOCALE 関数は以前のロケールを返します。この例では、以前のロケールは English_United States です。

ステートメント	結果
<pre>data_null; x=setlocale('ja_JP'); put x=; run;</pre>	<pre>x=English_UnitedStates</pre>

例 2

次の例では、SETLOCALE 関数は要素値が変更されているロケール名を返します。

ステートメント	結果
<pre>data_null_; x = setlocale("LC_MONETARY", 'zh_CN'); put x=; run;</pre>	<pre>x=Japanese_Japan</pre>

例 3

次の例では、SETLOCALE 関数は指定された DATE_YEAR_FORMAT キーの値を変更します。

ステートメント	結果
<pre>data null; x=setlocale('DATE_YEAR_FORMAT', '%Y'); put x=; run;</pre>	<pre>x=%Y%</pre>

TRANTAB 関数

指定された変換テーブルを使用してデータをトランスコードします。

カテゴリ: 文字

構文

TRANTAB(*string*,*trantab_name*)

必須引数*string*

トランスコードされる入力データ。

trantab_name

変換テーブル。変換テーブルは、各国語の要件をサポートするために SAS 6 で導入されました。SAS 8.2 では、変換テーブルの直接使用より優れた、LOCALE=システムオプションが導入されました。SAS 9.2 では、TRANTAB 関数が下位互換性を維持するためにサポートされています。ただし、SAS 8.2 以降のリリースでは、LOCALE=システムオプションを使用することをお勧めします。

詳細

TRANTAB 関数では、変換テーブルを使用して 1 つの内部表示形式から別の内部表示形式に文字を再マッピングすることによって、データ文字列をトランスコードします。入力文字列のデータのエンコーディングは、変換テーブルのテーブル 1 のエンコーディングと一致する必要があります。TRANTAB 関数では、テーブル 1 を使用してエンコーディングからデータを再マッピングします。

注意:

TRANTAB 関数は、経験を積んだ SAS ユーザーのみが使用してください。

例

次の例では、Latin2 でエンコードされたデータを大文字の Latin2 エンコーディングにトランスコードする変換テーブルを使用します。

ステートメント	結果
<pre>teststrg=trantab('testing','lat2_ucs'); put teststrg;</pre>	TESTING

関連項目:

プロシジャ:

- [17 章, “TRANTAB プロシジャ,” \(521 ページ\)](#)

UNICODE 関数

Unicode 文字を現在の SAS セッションエンコーディングに変換します。

カテゴリ: 文字

構文

STR=UNICODE(<instr> (<Unicode type>))

必須引数

str

現在の SAS セッションエンコーディングに変換されたデータ文字列。

instr

入力データ文字列。

Unicode type

Unicode 文字出力形式

ESC	Unicode エスケープ(たとえば、\u0042) ESC はデフォルトの出力形式です。
NCR	数値文字表現(たとえば、大 や±)
PAREN	Unicode カッコエスケープ(たとえば、<u0061>)
UCS2	ネイティブエンディアンの UCS2 エンコーディング
UCS2B	ビッグエンディアンの UCS2 エンコーディング
UCS2L	リトルエンディアンの UCS2 エンコーディング
UCS4	ネイティブエンディアンの UCS4 エンコーディング
UCS4B	ビッグエンディアンの UCS4 エンコーディング
UCS4L	リトルエンディアンの UCS4 エンコーディング
UTF16	ビッグエンディアンの UTF16 エンコーディング
UTF16B	ビッグエンディアンの UTF16 エンコーディング
UTF16L	リトルエンディアンの UTF16 エンコーディング
UTF8	UTF8 エンコーディング

詳細

Unicode 文字を読み込み、現在の SAS セッションエンコーディングに変換します。

例

次に、UNICODE 関数の使用例を示します。

```
Examples: (Submitted under Little endian system.)
str1=unicode(*\u0041\u0042\u0043*);
str2=unicode(*\0041\u0042\uu43*, 'esc');
str3=unicode(*&# 177;* , 'ncr');
str4=unicode(*&# 22823;* , 'ncr');
str5=unicode(*<u0061><u0062>* , 'paren');
str6=unicode('2759'x, 'ucs2');
str7=unicode('5927'x, 'ucs2b');
str8=unicode('2759'x, 'ucs2l');
str9=unicode('27590000'x, 'ucs4');
str10=unicode('00005927'x, 'ucs4b');
str11=unicode('27590000'x, 'ucs4l');
str12=unicode('E5A4A7'x, 'utf8');
str13=unicode('2759'x, 'utf16');
str14=unicode('5927'x, 'utf16b');
str15=unicode('2759'x, 'utf16l');
```

```
Results:
str1=ABC
str1=ABC
str3=±
str4=大
str5=ab
str6=大
str7=大
str8=大
str9=大
str10=大
str11=大
str12=大
str13=大
str14=大
str15=大
```

UNICODEC 関数

現在の SAS セッションエンコーディングの文字を Unicode 文字に変換します。

カテゴリ: 文字

構文

STR=UNICODEC(<instr> (<Unicode type>))

必須引数***str***

Unicode エンコーディングに変換されたデータ文字列。

instr

入力データ文字列。

Unicode type

Unicode 文字出力形式

ESC	Unicode エスケープ(たとえば、\u0042)。ESC はデフォルトの出力形式です。
NCR	数値文字表現(たとえば、大 や±))
PAREN	Unicode カッコエスケープ(たとえば、<u0061>)
UCS2	ネイティブエンディアンの UCS2 エンコーディング
UCS2B	ビッグエンディアンの UCS2 エンコーディング
UCS2L	リトルエンディアンの UCS2 エンコーディング
UCS4	ネイティブエンディアンの UCS4 エンコーディング
UCS4B	ビッグエンディアンの UCS4 エンコーディング
UCS4L	リトルエンディアンの UCS4 エンコーディング
UTF16	ビッグエンディアンの UTF16 エンコーディング
UTF16B	ビッグエンディアンの UTF16 エンコーディング
UTF16L	リトルエンディアンの UTF16 エンコーディング
UTF8	UTF8 エンコーディング

詳細

現在の SAS セッションエンコーディングに含まれる文字を読み込み、Unicode エンコーディングに変換します。

例

次に、UNICODEC 関数の使用例を示します。

```
length str4 $20;
dai=unicode('\u5927');
str1=unicodec("ABC");
str2=unicodec("ABC", 'esc');
str3=unicodec(dai, 'ncr');
str4=unicodec("ab", 'paren');
str5=unicodec(dai, 'ucs2');
str6=unicodec(dai, 'ucs2b');
str7=unicodec(dai, 'ucs2l');
str8=unicodec(dai, 'ucs4');
str9=unicodec(dai, 'ucs4b');
str10=unicodec(dai, 'ucs4l');
str11=unicodec(dai, 'utf8');
str12=unicodec(dai, 'utf16');
str13=unicodec(dai, 'utf16b');
str14=unicodec(dai, 'utf16l');
```

```

Results:
str1=414243
str2=414243
str3=
str4=str5=2759
str6=5927
str7=2759
str8=27590000
str9=00005927
str10=27590000
str11=E5A4A7
str12=2759
str13=5927
str14=2759

```

UNICODELEN 関数

Unicode データの文字単位の長さを指定します。

カテゴリ: 文字

構文

UNICODELEN()

詳細

UNICODELEN 関数では、Unicode データの文字単位の長さを指定します。

例

この例では、UNIX オペレーティングシステムでサポートされている日本語(Shift_JIS)セッションエンコーディングを使用します。

ステートメント	結果
<code>len1=unicodelen("abc 太");</code>	len1=4
<code>len2=unicodelen("\u0041\u0042\u0043\u5927", 'esc ');</code>	len2=4
<code>len3=unicodelen("&#22823;' , 'ncr');</code>	len3=1
<code>len4=unicodelen("<u0061><u0062>", 'paren');</code>	len4=2

関連項目:

関数:

- [“UNICODEWIDTH 関数” \(311 ページ\)](#)

UNICODEWIDTH 関数

Unicode データの表示単位の長さを指定します。

カテゴリ: 文字

構文

UNICODEWIDTH()

詳細

UNICODEWIDTH 関数では、Unicode データの表示単位の長さを指定します。表示単位は、文字を固定幅フォントで表示する時の文字の幅を表示します。0x3000 から 0x303F、0x3400 から 0x4DFF、0x4E00 から 0x9FFF、0xF900 から 0xFAFF までの間の文字は、表示単位が 2 の値になります。他の文字は表示単位が 1 になります。

例

この例では、UNIX オペレーティングシステムでサポートされている日本語(Shift_JIS)セッションエンコーディングを使用します。

ステートメント	結果
<code>len1=unicodewidth("abc太");</code>	len1=5
<code>len2=unicodewidth("\u0041\u0042\u0043\u5927",'esc');</code>	len2=5
<code>len3=unicodewidth("&#22823; ", 'ncr');</code>	len3=2
<code>len4=unicodewidth("<u0061><u0062>", 'paren');</code>	len4=2

関連項目:

関数:

- [“UNICODELEN 関数” \(310 ページ\)](#)

VARTRANSCODE 関数

SAS データセット変数のトランスコーディング属性を返します。

カテゴリ: 変数情報

構文

VARTRANSCODE(*data-set-id*, *var-num*)

必須引数

data-set-id

OPEN 関数が返すデータセット識別子を指定します。

var-num

SAS データセット内での変数の位置を指定します。

ヒント: VARNUM 関数はこの値を返します。

詳細

トランスコーディングとは、1 つのエンコーディングから別のエンコーディングにデータを変換する処理のことです。VARTRANSCODE 関数は、*var-num* 変数とその値をトランスコードしない場合は 0 を、*var-num* 変数とその値をトランスコードする場合は 1 を返します。

変数のトランスコーディングの詳細については、[トランスコーディング \(27 ページ\)](#)のセクション(*SAS 各国語サポート(NLS): リファレンスガイド*)を参照してください。値のエンコーディングおよびデータのトランスコーディングの詳細については、[SAS データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値 \(563 ページ\)](#)のセクション(*SAS 各国語サポート(NLS): リファレンスガイド*)を参照してください。

例

次の例は、文字変数をトランスコードするかどうかの決定方法を示しています。

```

data a;
  attrib x length=$3. transcode=no;
  attrib y length=$3. transcode=yes;
  x='abc';
  y='xyz';
run;
data _null_;
  dsid=open('work.a','i');
  nobs=attrn(dsid,"nobs");
  nvars=attrn(dsid,"nvars");
  do i=1 to nobs;
    xrc=fetch(dsid,1);
    do j=1 to nvars;
      transcode = vartranscode(dsid,j);
      put transcode=;
    end;
  end;
run;

```

SAS は次の出力をログに書き込みます。

```

transcode=0
transcode=1

```

関連項目:

関数:

- “ATTRN Function” in *SAS Functions and CALL Routines: Reference*
- “OPEN Function” in *SAS Functions and CALL Routines: Reference*

- “VARNUM Function” in *SAS Functions and CALL Routines: Reference*
- “VTRANSCODE 関数” (313 ページ)
- “VTRANSCODEX 関数” (314 ページ)

VTRANSCODE 関数

指定された文字変数でトランスコーディングが有効かどうかを示す値を返します。

カテゴリ: 変数情報

構文

VTRANSCODE (*var*)

必須引数

var

スカラまたは配列参照として表される文字変数を指定します。

制限事項: 式は引数として使用できません。

詳細

VTRANSCODE 関数は、トランスコーディングがオフの場合は 0 を、オンの場合は 1 を返します。

デフォルトでは、DATA ステップのすべての文字変数がトランスコードされます。トランスコーディングをオフにするには、ATTRIB ステートメントの TRANSCODE=属性を使用します。

比較

- VTRANSCODE 関数は、指定された変数でトランスコーディングが有効かどうかを示す値を返します。一方、VTRANSCODEX 関数は、変数名を判別するために引数を評価します。その後、VTRANSCODEX 関数はその変数名に関連付けられたトランスコーディングステータス(オンまたはオフ)を返します。
- VTRANSCODE 関数は、引数として式を受け入れません。VTRANSCODEX 関数は式を受け入れますが、指定された式の値で配列参照を示すことはできません。
- 関連する関数が、変数名、種類、出力形式、長さなど他の変数属性の値を返します。変数属性のリストについては、“変数情報”の関数(*SAS 関数と CALL ルーチン: リファレンス*)を参照してください。

例

ステートメント	結果
	-----1-----

ステートメント	結果
<pre>attrib x transcode = yes; attrib y transcode = no; rc1 = vtranscode(y); put rc1=;</pre>	rc1=0

関連項目:

関数:

- [“VTRANSCODEX 関数” \(314 ページ\)](#)

ステートメント:

- ATTRIB

VTRANSCODEX 関数

指定された引数でトランスコーディングが有効かどうかを示す値を返します。

カテゴリ: 変数情報

構文

VTRANSCODEX (*var*)

必須引数

var

文字変数名に対して評価する SAS 文字式を指定します。

制限事項: 指定された式の値で配列参照を示すことはできません。

詳細

VTRANSCODEX 関数は、トランスコーディングがオフの場合は 0 を、オンの場合は 1 を返します。

デフォルトでは、DATA ステップのすべての文字変数がトランスコードされます。トランスコーディングをオフにするには、ATTRIB ステートメントの TRANSCODE=属性を使用します。

比較

- VTRANSCODE 関数は、指定された変数でトランスコーディングが有効かどうかを示す値を返します。一方、VTRANSCODEX 関数は、変数名を判別するために引数を実評価します。その後、VTRANSCODEX 関数はその変数名に関連付けられたトランスコーディングステータス(オンまたはオフ)を返します。
- VTRANSCODE 関数は、引数として式を受け入れません。VTRANSCODEX 関数は式を受け入れますが、指定された式の値で配列参照を示すことはできません。

- 関連する関数が、変数名、種類、出力形式、長さなど他の変数属性の値を返しません。変数属性のリストについては、“変数情報”の関数(*SAS 関数と CALL ルーチン: リファレンス*)を参照してください。

例

ステートメント	結果
	----+-----1-----+
<pre>attrib x transcode = yes; attrib y transcode = no; rc1 = vtranscodex('y'); put rc1=;</pre>	rc1=0

関連項目:

関数:

- [“VTRANSCODE 関数” \(313 ページ\)](#)

ステートメント:

- ATTRIB

6 部

各国語サポート関連の入力形式

12 章	
入力形式エントリ	319

12 章

入力形式エントリ

カテゴリ別の入力形式	321
ディクショナリ	328
\$CPTDWw. 入力形式	328
\$CPTWDw. 入力形式	329
EUROw.d 入力形式	330
EUROXw.d 入力形式	332
\$KANJIw. 入力形式	334
\$KANJIXw. 入力形式	334
\$LOGVSw. 入力形式	335
\$LOGVSRw. 入力形式	336
MINGUOw. 入力形式	337
NENGOw. 入力形式	339
NLDATEw. 入力形式	340
NLDATEw. 入力形式	341
NLMNIAEDw.d 入力形式	342
NLMNIAUDw.d 入力形式	343
NLMNIBGNw.d 入力形式	344
NLMNIBRLw.d 入力形式	345
NLMNICADw.d 入力形式	345
NLMNICHFw.d 入力形式	346
NLMNICNYw.d 入力形式	347
NLMNICZKw.d 入力形式	348
NLMNIDKKw.d 入力形式	349
NLMNIEEKw.d 入力形式	350
NLMNIEGPw.d 入力形式	351
NLMNIEURw.d 入力形式	352
NLMNIGBPw.d 入力形式	353
NLMNIHKDw.d 入力形式	354
NLMNIHRKw.d 入力形式	354
NLMNIHUFw.d 入力形式	355
NLMNIIDRw.d 入力形式	356
NLMNIILSw.d 入力形式	357
NLMNIINRw.d 入力形式	358
NLMNIJPYw.d 入力形式	359
NLMNIKRWw.d 入力形式	360
NLMNILTLw.d 入力形式	361
NLMNILVLw.d 入力形式	362
NLMNIMOPw.d 入力形式	363
NLMNIMXNw.d 入力形式	363
NLMNIMYRw.d 入力形式	364
NLMNINOKw.d 入力形式	365

NLMNINZDw.d 入力形式	366
NLMNIPLNw.d 入力形式	367
NLMNIRUBw.d 入力形式	368
NLMNISEKw.d 入力形式	369
NLMNISGDw.d 入力形式	370
NLMNITHBw.d 入力形式	371
NLMNITRYw.d 入力形式	372
NLMNITWDw.d 入力形式	372
NLMNIUSDw.d 入力形式	373
NLMNIZARw.d 入力形式	374
NLMNLAEDw.d 入力形式	375
NLMNLAUDw.d 入力形式	376
NLMNLBGNw.d 入力形式	377
NLMNLBRLw.d 入力形式	378
NLMNLCADw.d 入力形式	379
NLMNLCHFw.d 入力形式	380
NLMNLCNYw.d 入力形式	381
NLMNLCZKw.d 入力形式	381
NLMNLDKKw.d 入力形式	382
NLMNLEEKw.d 入力形式	383
NLMNLEGPw.d 入力形式	384
NLMNLEURw.d 入力形式	385
NLMNLGBPw.d 入力形式	386
NLMNLHKDw.d 入力形式	387
NLMNLHRKw.d 入力形式	388
NLMNLHUFw.d 入力形式	389
NLMNLIDRw.d 入力形式	390
NLMNLILSw.d 入力形式	390
NLMNLINRw.d 入力形式	391
NLMNLJPYw.d 入力形式	392
NLMNLKRWw.d 入力形式	393
NLMNLLTLw.d 入力形式	394
NLMNLLVLw.d 入力形式	395
NLMNLMOPw.d 入力形式	396
NLMNLMXNw.d 入力形式	397
NLMNLMYRw.d 入力形式	398
NLMNLNOKw.d 入力形式	399
NLMNLNZDw.d 入力形式	399
NLMNLPLNw.d 入力形式	400
NLMNLRUBw.d 入力形式	401
NLMNLSEKw.d 入力形式	402
NLMNLSGDw.d 入力形式	403
NLMNLTHBw.d 入力形式	404
NLMNLTRYw.d 入力形式	405
NLMNLTWDw.d 入力形式	406
NLMNLUSDw.d 入力形式	407
NLMNLZARw.d 入力形式	408
NLMNYw.d 入力形式	408
NLMNYIw.d 入力形式	410
NLNUMw.d 入力形式	411
NLNUMIw.d 入力形式	412
NLPCTw.d 入力形式	413
NLPCTIw.d 入力形式	414
NLTIMAPw. 入力形式	415
NLTIMEw. 入力形式	416
\$REVERJw. 入力形式	417

\$REVERSw. 入力形式	418
\$UCS2Bw. 入力形式	419
\$UCS2BEw. 入力形式	420
\$UCS2Lw. 入力形式	421
\$UCS2LEw. 入力形式	422
\$UCS2Xw. 入力形式	423
\$UCS2XEw. 入力形式	424
\$UCS4Bw. 入力形式	425
\$UCS4Lw. 入力形式	426
\$UCS4Xw. 入力形式	427
\$UCS4XEw. 入力形式	428
\$UESCw. 入力形式	429
\$UESCEw. 入力形式	430
\$UNCRw. 入力形式	431
\$UNCREw. 入力形式	432
\$UPARENw. 入力形式	433
\$UPARENEw. 入力形式	434
\$UPARENpw. 入力形式	435
\$UTF8Xw. 入力形式	436
\$VSLOGw. 入力形式	437
\$VSLOGRw. 入力形式	438
YENw.d 入力形式	439

カテゴリ別の入力形式

各国語サポート関連の入力形式には、次の 6 つのカテゴリがあります。

カテゴリ	説明
BIDI テキストの操作	データ変数から双方向のデータ値を読み込むように指示します。
文字	文字変数に文字データ値を読み込むように指示します。
DBCS	さまざまなアジア言語を管理するように指示します。
日付と時間	日付、時刻および日時を表す変数にデータ値を読み込むように指示します。
ヘブライ語のテキストの操作	データ変数からヘブライ語のデータを読み込むように指示します。
数値	数値変数に数値データ値を読み込むように指示します。

次の表に、SAS 入力形式の概要を示します。詳細については、それぞれの各国語サポート関連の入力形式に関する情報を参照してください。

カテゴリ	言語要素	説明
BIDI テキストの操作	\$LOGVSw. 入力形式 (p. 335)	左から右に論理的順序で並ぶ文字列を読み込み、その文字列を視覚的順序に変換します。
	\$LOGVSRw. 入力形式 (p. 336)	右から左に論理的順序で並ぶ文字列を読み込み、その文字列を視覚的順序に変換します。
	\$VSLOGw. 入力形式 (p. 437)	視覚的順序で並ぶ文字列を読み込み、その文字列を左から右の論理的順序に変換します。
	\$VSLOGRw. 入力形式 (p. 438)	視覚的順序で並ぶ文字列を読み込み、その文字列を右から左の論理的順序に変換します。
DBCS	\$KANJIw. 入力形式 (p. 334)	DBCS データからシフトコードデータを削除します。
	\$KANJIXw. 入力形式 (p. 334)	シフトコードデータを DBCS データに追加します。
数値	EUROw.d 入力形式 (p. 330)	数値を読み込み、ヨーロッパ通貨の埋め込み文字を取り除いて、カンマと小数点を逆にします。
	EUROXw.d 入力形式 (p. 332)	数値を読み込み、ヨーロッパ通貨の埋め込み文字を削除します。
	NLMNIAEDw.d 入力形式 (p. 342)	アラブ首長国連邦の国際表示形式で通貨データを読み込みます。
	NLMNIAUDw.d 入力形式 (p. 343)	オーストラリアの国際表示形式で通貨データを読み込みます。
	NLMNIBGNw.d 入力形式 (p. 344)	ブルガリアの国際表示形式で通貨データを読み込みます。
	NLMNIBRLw.d 入力形式 (p. 345)	ブラジルの国際表示形式で通貨データを読み込みます。
	NLMNICADw.d 入力形式 (p. 345)	カナダの国際表示形式で通貨データを読み込みます。
	NLMNICHfw.d 入力形式 (p. 346)	リヒテンシュタインとスイスの国際表示形式で通貨データを読み込みます。
	NLMNICNYw.d 入力形式 (p. 347)	中国の国際表示形式で通貨データを読み込みます。
	NLMNICZKw.d 入力形式 (p. 348)	チェコ共和国の国際表示形式で通貨データを読み込みます。
	NLMNIDKKw.d 入力形式 (p. 349)	デンマーク、フェロー諸島およびグリーンランドの国際表示形式で通貨データを読み込みます。
	NLMNIEEKw.d 入力形式 (p. 350)	エストニアの国際表示形式で通貨データを読み込みます。
NLMNIEGPw.d 入力形式 (p. 351)	エジプトの国際表示形式で通貨データを読み込みます。	

カテゴリ	言語要素	説明
	NLMNIEURw.d 入力形式 (p. 352)	オーストリア、ベルギー、フィンランド、フランス、ドイツ、ギリシャ、アイルランド、イタリア、ルクセンブルク、マルタ、オランダ、ポルトガル、スロベニア、スペインの国際表示形式で通貨データを読み込みます。
	NLMNIGBPw.d 入力形式 (p. 353)	英国の国際表示形式で通貨データを読み込みます。
	NLMNIHKDw.d 入力形式 (p. 354)	香港の国際表示形式で通貨データを読み込みます。
	NLMNIHRKw.d 入力形式 (p. 354)	クロアチアの国際表示形式で通貨データを読み込みます。
	NLMNIHUFw.d 入力形式 (p. 355)	ハンガリーの国際表示形式で通貨データを読み込みます。
	NLMNIIDRw.d 入力形式 (p. 356)	インドネシアの国際表示形式で通貨データを読み込みます。
	NLMNIILSw.d 入力形式 (p. 357)	イスラエルの国際表示形式で通貨データを読み込みます。
	NLMNIINRw.d 入力形式 (p. 358)	インドの国際表示形式で通貨データを読み込みます。
	NLMNIJPYw.d 入力形式 (p. 359)	日本の国際表示形式で通貨データを読み込みます。
	NLMNIKRWw.d 入力形式 (p. 360)	韓国の国際表示形式で通貨データを読み込みます。
	NLMNITLw.d 入力形式 (p. 361)	リトアニアの国際表示形式で通貨データを読み込みます。
	NLMNIVLw.d 入力形式 (p. 362)	ラトビアの国際表示形式で通貨データを読み込みます。
	NLMNIMOPw.d 入力形式 (p. 363)	マカオの国際表示形式で通貨データを読み込みます。
	NLMNIMXNw.d 入力形式 (p. 363)	メキシコの国際表示形式で通貨データを読み込みます。
	NLMNIMYRw.d 入力形式 (p. 364)	マレーシアの国際表示形式で通貨データを読み込みます。
	NLMNINOKw.d 入力形式 (p. 365)	ノルウェーの国際表示形式で通貨データを読み込みます。
	NLMNINZDw.d 入力形式 (p. 366)	ニュージーランドの国際表示形式で通貨データを読み込みます。

カテゴリ	言語要素	説明
	NLMNIPLNw.d 入力形式 (p. 367)	ポーランドの国際表示形式で通貨データを読み込みます。
	NLMNIRUBw.d 入力形式 (p. 368)	ロシアの国際表示形式で通貨データを読み込みます。
	NLMNISEKw.d 入力形式 (p. 369)	スウェーデンの国際表示形式で通貨データを読み込みます。
	NLMNISGDw.d 入力形式 (p. 370)	シンガポールの国際表示形式で通貨データを読み込みます。
	NLMNITHBw.d 入力形式 (p. 371)	タイの国際表示形式で通貨データを読み込みます。
	NLMNITRYw.d 入力形式 (p. 372)	トルコの国際表示形式で通貨データを読み込みます。
	NLMNITWDw.d 入力形式 (p. 372)	台湾の国際表示形式で通貨データを読み込みます。
	NLMNIUSDw.d 入力形式 (p. 373)	プエルトリコと米国の国際表示形式で通貨データを読み込みます。
	NLMNIZARw.d 入力形式 (p. 374)	南アフリカの国際表示形式で通貨データを読み込みます。
	NLMNLAEDw.d 入力形式 (p. 375)	アラブ首長国連邦のローカル表示形式で通貨データを読み込みます。
	NLMNLAUDw.d 入力形式 (p. 376)	オーストラリアのローカル表示形式で通貨データを読み込みます。
	NLMNLBGWw.d 入力形式 (p. 377)	ブルガリアのローカル表示形式で通貨データを読み込みます。
	NLMNLBRLw.d 入力形式 (p. 378)	ブラジルのローカル表示形式で通貨データを読み込みます。
	NLMNLCADw.d 入力形式 (p. 379)	カナダのローカル表示形式で通貨データを読み込みます。
	NLMNLCHFw.d 入力形式 (p. 380)	リヒテンシュタインとスイスのローカル表示形式で通貨データを読み込みます。
	NLMNLCNYw.d 入力形式 (p. 381)	中国のローカル表示形式で通貨データを読み込みます。
	NLMNLCZKw.d 入力形式 (p. 381)	チェコ共和国のローカル表示形式で通貨データを読み込みます。
	NLMNLDDKw.d 入力形式 (p. 382)	デンマーク、フェロー諸島およびグリーンランドのローカル表示形式で通貨データを読み込みます。

カテゴリ	言語要素	説明
	NLMNLEEKw.d 入力形式 (p. 383)	エストニアのローカル表示形式で通貨データを読み込みます。
	NLMNLEGPw.d 入力形式 (p. 384)	エジプトのローカル表示形式で通貨データを読み込みます。
	NLMNLEURw.d 入力形式 (p. 385)	オーストリア、ベルギー、フィンランド、フランス、ドイツ、ギリシャ、アイルランド、イタリア、ルクセンブルク、マルタ、オランダ、ポルトガル、スロベニア、スペインのローカル表示形式で通貨データを読み込みます。
	NLMNLGBPw.d 入力形式 (p. 386)	英国のローカル表示形式で通貨データを読み込みます。
	NLMNLHKDw.d 入力形式 (p. 387)	香港のローカル表示形式で通貨データを読み込みます。
	NLMNLHRKw.d 入力形式 (p. 388)	クロアチアのローカル表示形式で通貨データを読み込みます。
	NLMNLHUFw.d 入力形式 (p. 389)	ハンガリーのローカル表示形式で通貨データを読み込みます。
	NLMNLIDRw.d 入力形式 (p. 390)	インドネシアのローカル表示形式で通貨データを読み込みます。
	NLMNLILSw.d 入力形式 (p. 390)	イスラエルのローカル表示形式で通貨データを読み込みます。
	NLMNLINRw.d 入力形式 (p. 391)	インドのローカル表示形式で通貨データを読み込みます。
	NLMNLJPYw.d 入力形式 (p. 392)	日本のローカル表示形式で通貨データを読み込みます。
	NLMNLKRWw.d 入力形式 (p. 393)	韓国のローカル表示形式で通貨データを読み込みます。
	NLMNLLTLw.d 入力形式 (p. 394)	リトアニアのローカル表示形式で通貨データを読み込みます。
	NLMNLLVLw.d 入力形式 (p. 395)	ラトビアのローカル表示形式で通貨データを読み込みます。
	NLMNLMOPw.d 入力形式 (p. 396)	マカオのローカル表示形式で通貨データを読み込みます。
	NLMNLMXNw.d 入力形式 (p. 397)	メキシコのローカル表示形式で通貨データを読み込みます。
	NLMNLMYRw.d 入力形式 (p. 398)	マレーシアのローカル表示形式で通貨データを読み込みます。

カテゴリ	言語要素	説明
	NLMNLNOKw.d 入力形式 (p. 399)	ノルウェーのローカル表示形式で通貨データを読み込みます。
	NLMNLNZDw.d 入力形式 (p. 399)	ニュージーランドのローカル表示形式で通貨データを読み込みます。
	NLMNLPLNw.d 入力形式 (p. 400)	ポーランドのローカル表示形式で通貨データを読み込みます。
	NLMNLRUBw.d 入力形式 (p. 401)	ロシアのローカル表示形式で通貨データを読み込みます。
	NLMNLSEKw.d 入力形式 (p. 402)	スウェーデンのローカル表示形式で通貨データを読み込みます。
	NLMNLSGDw.d 入力形式 (p. 403)	シンガポールのローカル表示形式で通貨データを読み込みます。
	NLMNLTHBw.d 入力形式 (p. 404)	タイのローカル表示形式で通貨データを読み込みます。
	NLMNLTRYw.d 入力形式 (p. 405)	トルコのローカル表示形式で通貨データを読み込みます。
	NLMNLTWDw.d 入力形式 (p. 406)	台湾のローカル表示形式で通貨データを読み込みます。
	NLMNLUSDw.d 入力形式 (p. 407)	プエルトリコと米国のローカル表示形式で通貨データを読み込みます。
	NLMNLZARw.d 入力形式 (p. 408)	南アフリカのローカル表示形式で通貨データを読み込みます。
	NLMNYw.d 入力形式 (p. 408)	指定されたロケールの通貨データをローカル表示形式で読み込み、そのデータを数値に変換します。
	NLMNYIw.d 入力形式 (p. 410)	指定されたロケールの通貨データを国際表示形式で読み込み、そのデータを数値に変換します。
	NLNUMw.d 入力形式 (p. 411)	指定されたロケールの数値データをローカル表示形式で読み込み、そのデータを数値に変換します。
	NLNUMIw.d 入力形式 (p. 412)	指定されたロケールの数値データを国際表示形式で読み込み、そのデータを数値に変換します。
	NLPCTw.d 入力形式 (p. 413)	指定されたロケールの百分率データをローカル表示形式で読み込み、そのデータを数値に変換します。
	NLPCTIw.d 入力形式 (p. 414)	指定されたロケールの百分率データを国際表示形式で読み込み、そのデータを数値に変換します。
	YENw.d 入力形式 (p. 439)	埋め込まれた円記号、カンマおよび小数点を削除します。
日付と時間	MINGUOW. 入力形式 (p. 337)	台湾の形式の日付を読み込みます。

カテゴリ	言語要素	説明
	NENGOw. 入力形式 (p. 339)	eyymmdd 形式の日本の日付値を読み込みます。
	NLDATEw. 入力形式 (p. 340)	指定されたロケールの日付値を読み込み、その日付値をローカルの SAS 日付値に変換します。
	NLDATMw. 入力形式 (p. 341)	指定されたロケールの日時値を読み込み、その日時値をローカルの SAS 日時値に変換します。
	NLTIMAPw. 入力形式 (p. 415)	指定されたロケールの時間値を a.m.および p.m.表示を使用して読み込み、その時間値をローカルの SAS 時間値に変換します。
	NLTIMEw. 入力形式 (p. 416)	指定されたロケールの時間値を読み込み、その時間値をローカルの SAS 時間値に変換します。
ヘブライ語のテキストの操作	\$CPTDWw. 入力形式 (p. 328)	ヘブライ語 DOS (cp862)エンコーディングの文字列を読み込み、その文字列を Windows (cp1255)エンコーディングに変換します。
	\$CPTWDw. 入力形式 (p. 329)	Windows (cp1255)エンコーディングの文字列を読み込み、その文字列をヘブライ語 DOS (cp862)エンコーディングに変換します。
文字	\$REVERJw. 入力形式 (p. 417)	文字データを右から左の順に読み込み、空白を保持します。
	\$REVERSw. 入力形式 (p. 418)	文字データを右から左の順に左詰めで読み込みます。
	\$UCS2Bw. 入力形式 (p. 419)	ビッグエンディアン、16 ビット、UCS2 形式の UNICODE(ユニコード)文字列を読み取り、その文字列を現在の SAS セッションのエンコーディングに変換します。
	\$UCS2BEw. 入力形式 (p. 420)	現在の SAS セッションのエンコーディングの文字列を読み込み、その文字列をビッグエンディアン、16 ビット、UCS2 形式の UNICODE(ユニコード)に変換します。
	\$UCS2Lw. 入力形式 (p. 421)	リトルエンディアン、16 ビット、UCS2 形式の UNICODE(ユニコード)文字列を読み取り、その文字列を現在の SAS セッションのエンコーディングに変換します。
	\$UCS2LEw. 入力形式 (p. 422)	現在の SAS セッションのエンコーディングにある文字列を読み込み、その文字列をリトルエンディアン、16 ビット、UCS2 形式の UNICODE(ユニコード)に変換します。
	\$UCS2Xw. 入力形式 (p. 423)	16 ビット、UCS2 形式の UNICODE(ユニコード)文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換します。
	\$UCS2XEw. 入力形式 (p. 424)	現在の SAS セッションのエンコーディングの文字列を読み込み、その文字列を、16 ビット、UCS2 形式の UNICODE(ユニコード)に変換します。

カテゴリ	言語要素	説明
	\$UCS4Bw. 入力形式 (p. 425)	ビッグエンディアン、32 ビット、UCS4 形式の UNICODE(ユニコード)文字列を読み取り、その文字列を現在の SAS セッションのエンコーディングに変換します。
	\$UCS4Lw. 入力形式 (p. 426)	リトルエンディアン、32 ビット、UCS4 形式の UNICODE(ユニコード)文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換します。
	\$UCS4Xw. 入力形式 (p. 427)	32 ビット、UCS4 形式の UNICODE(ユニコード)文字列を読み取り、その文字列を現在の SAS セッションのエンコーディングに変換します。
	\$UCS4XEw. 入力形式 (p. 428)	現在の SAS セッションのエンコーディングの文字列を読み込み、その文字列を、32 ビット、UCS4 形式の UNICODE(ユニコード)に変換します。
	\$UESCw. 入力形式 (p. 429)	UESC 表記でエンコードされた文字列を読み取り、その文字列を現在の SAS セッションのエンコーディングに変換します。
	\$UESCEw. 入力形式 (p. 430)	現在の SAS セッションのエンコーディングを使用する文字列を読み取り、その文字列を UESC 表記に変換します。
	\$UNCRw. 入力形式 (p. 431)	NCR 文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換します。
	\$UNCREw. 入力形式 (p. 432)	現在の SAS セッションのエンコーディングを使用する文字列を読み込み、その文字列を NCR に変換します。
	\$UPARENw. 入力形式 (p. 433)	UPAREN 表記でエンコードされた文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換します。
	\$UPARENEw. 入力形式 (p. 434)	現在の SAS セッションのエンコーディングを使用する文字列を読み込み、その文字列を UPAREN 表記に変換します。
	\$UPARENpw. 入力形式 (p. 435)	UPAREN 表記でエンコードされた文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換しますが、各国語文字は UPAREN 表記のエンコーディングのまま残ります。
	\$UTF8Xw. 入力形式 (p. 436)	UTF-8 でエンコードされた文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換します。

ディクショナリ

\$CPTDWw. 入力形式

ヘブライ語 DOS (cp862)エンコーディングの文字列を読み込み、その文字列を Windows (cp1255)エンコーディングに変換します。

カテゴリ: ヘブライ語のテキストの操作

構文

`$CPTDWw.`

説明

`w`
 入力幅を指定します。
 デフォルト: 200
 範囲: 1–32000

比較

`$CPTDWw.`入力形式では、`$CPTWDw.`入力形式とは逆の処理を行います。

例

次の例では、入力値として 808182 を使用します。

ステートメント	結果
	----+-----1-----+
<pre>x=input('808182',\$cptdw6.); put x;</pre>	אבא

関連項目:

出力形式:

- “[\\$CPTDWw. 出力形式](#)” (85 ページ)
- “[\\$CPTWDw. 出力形式](#)” (85 ページ)

入力形式:

- “[\\$CPTWDw. 入力形式](#)” (329 ページ)

\$CPTWDw. 入力形式

Windows (cp1255)エンコーディングの文字列を読み込み、その文字列をヘブライ語 DOS (cp862)エンコーディングに変換します。

カテゴリ: ヘブライ語のテキストの操作

構文

`$CPTWDw.`

説明

w
 入力幅を指定します。
デフォルト: 200
範囲: 1–32000

比較

\$CPTDw.入力形式では、\$CPTDw.入力形式とは逆の処理を行います。

例

次の例では、入力値として1100を使用します。

ステートメント	結果
	----+----1-----+
<code>x=input (' 1100 ', \$cptwd6.);</code> <code>put x;</code>	€□,

関連項目:**出力形式:**

- “\$CPTDw. 出力形式” (85 ページ)
- “\$CPTDw. 出力形式” (85 ページ)

入力形式:

- “\$CPTDw. 入力形式” (328 ページ)

EUROw.d 入力形式

数値を読み込み、ヨーロッパ通貨の埋め込み文字を取り除いて、カンマと小数点を逆にします。

カテゴリ: 数値

構文

EUROw.d

説明

w
 入力幅を指定します。
デフォルト: 6
範囲: 1–32

d

値を除算する 10 のべき乗を指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

詳細

EUROw.d 入力形式では数値を読み込み、埋め込みユーロ記号(E)、カンマ、空白、パーセント記号、ダッシュ、閉じかっこを入力データから削除します。小数点は、実数部と小数部を分ける区切り文字と見なされます。EUROw.d 入力形式では、フィールドの先頭にある開きかっこがマイナス記号に変換されます。

比較

- EUROw.d 入力形式は EUROXwd 入力形式に似ていますが、EUROXwd では小数点とカンマの役割が逆になっています。この規則はヨーロッパの国で共通です。
- 入力にカンマもピリオドもない場合、EUROw.d 入力形式と EUROXwd 入力形式は相互に交換可能です。

例

次の表に、ユーロでの通貨の入力値、適用される SAS ステートメントおよび結果を示します。

```
data _null_;
input x euro10.;
put x=;
datalines;
E1
E1.23
1.23
1,234.56
;
run;
SAS Log:
x=1
x=1.23
x=1.23
x=1234.56
```

値	ステートメント	結果
		----+----1----2
E1	input x euro10.; put x=;	1
E1.23	input x euro10.; put x=;	1.23
1.23	input x euro10.; put x=;	1.23

値	ステートメント	結果
1,234.56	input x euro10. ; put x;	1234.56

関連項目:

出力形式:

- “EUROw.d 出力形式” (86 ページ)
- “EUROXw.d 出力形式” (89 ページ)

入力形式:

- “EUROXw.d 入力形式” (332 ページ)

EUROXw.d 入力形式

数値を読み込み、ヨーロッパ通貨の埋め込み文字を削除します。

カテゴリ: 数値

構文

EUROXw.d

説明

w

入力幅を指定します。

デフォルト: 6

範囲: 1–32

d

値を除算する 10 のべき乗を指定します。データに小数点を表すカンマが含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

詳細

EUROXw.d 入力形式では数値を読み込み、埋め込みユーロ記号(E)、カンマ、空白、パーセント記号、ダッシュ、閉じかっこを入力データから削除します。カンマは、実数部と小数部を分ける区切り文字と見なされます。EUROXw.d 入力形式では、フィールドの先頭にある開きかっこがマイナス記号に変換されます。

比較

- EUROXw.d 入力形式は EUROw.d 入力形式に似ていますが、EUROw.d では、カンマと小数点の役割が逆になっています。この規則は英語圏の国で共通です。

- 入力にカンマもピリオドもない場合、EUROXw.d 入力形式と EUROw.d 入力形式は相互に交換可能です。

例

次の表に、ユーロでの通貨の入力値、適用される SAS ステートメントおよび結果を示します。

```
data _null_;
input x eurox10.;
put x=;
datalines;
E1
E1.23
1.23
1,234.56
; run;
SAS Log:
7 input x eurox10.;
8 put x=;
9 datalines;
x=1
x=123
x=123
x=1.23456
```

値	ステートメント	結果
		-----1-----2
E1	input x eurox10.; put x=;	1
E1.23	input x eurox10.; put x=;	123
1.23	input x eurox10.; put x=;	123
1,234.56	input x eurox10.; put x=;	1.23456

関連項目:

出力形式:

- [“EUROw.d 出力形式” \(86 ページ\)](#)
- [“EUROXw.d 出力形式” \(89 ページ\)](#)

入力形式:

- [“EUROw.d 入力形式” \(330 ページ\)](#)

\$KANJIw. 入力形式

DBCS データからシフトコードデータを削除します。

カテゴリ: DBCS

構文

\$KANJIw.

説明

w

入力幅を指定します。

範囲: 入力形式の最小幅は 2 です。

制限事項: 幅は偶数幅(2 バイト単位)で指定します。奇数幅を指定すると幅が切り捨てられます。この幅は、シフトコードデータの長さ以上にする必要があります。

詳細

\$KANJI 入力形式では、DBCS データからシフトコードデータが削除されます。
\$KANJI 入力形式では、ホストメインフレームデータが処理されます。\$KANJI は、他のプラットフォームで使用できます。\$KANJI 入力形式を非 EBCDIC (モーダルエンコーディングではない)ホストで使用する場合、データは変化しません。

SBCS の空データが返された場合を除き、データは SO で開始し、SI で終了する必要があります。入力データ長は、 $2 + (SO/SI \text{ 長}) * 2$ である必要があります。

関連項目:

出力形式:

- “\$KANJIw. 出力形式” (93 ページ)
- “\$KANJIIXw. 出力形式” (94 ページ)

入力形式:

- “\$KANJIIXw. 入力形式” (334 ページ)

\$KANJIIXw. 入力形式

シフトコードデータを DBCS データに追加します。

カテゴリ: DBCS

構文

\$KANJIIXw.

説明

w

入力幅を指定します。

範囲: 入力形式の最小幅は、 $2 + (\text{現在の DBCSTYPE} = \text{設定で使用されるシフトコードの長さ}) * 2$ です。

制限事項: 幅は偶数幅(2 バイト単位)で指定します。奇数幅を指定すると幅が切り捨てられます。この幅は、シフトコードデータの長さ以上にする必要があります。

詳細

\$KANJIX 入力形式では、シフトコードデータが、シフトコードデータのない DBCS データに追加されます。入力データが空白の場合、シフトコードデータは追加されません。\$KANJIX 入力形式で処理されるのはホストメインフレームデータですが、\$KANJIX は他のプラットフォームで使用できます。\$KANJIX 入力形式を非 EBCDIC (モデルエンコーディングではない)ホストで使用する場合、データは変化しません。

関連項目:

出力形式:

- “\$KANJIw. 出力形式” (93 ページ)
- “\$KANJIXw. 出力形式” (94 ページ)

入力形式:

- “\$KANJIw. 入力形式” (334 ページ)

\$LOGVSw. 入力形式

左から右に論理的順序で並ぶ文字列を読み込み、その文字列を視覚的順序に変換します。

カテゴリ: BIDI テキストの操作

構文

\$LOGVSw.

説明

w

入力幅を指定します。

デフォルト: 200

範囲: 1–32000

比較

\$LOGVSw.入力形式では、LOGVSRw.入力形式とは逆の処理を行います。

例

次の例では、“flight”のヘブライ語の入力値 תִּירוּסִי を使用します。

ステートメント	結果
	-----1-----+
x=input(' תִּירוּסִי flight',\$logvs12.); put x;	תִּירוּסִי flight

次の例では、“computer”のアラビア語の入力値 حاسوب を使用します。

ステートメント	結果
	-----1-----+
x=input(' حاسوب computer',\$logvs12.); put x;	حاسوب computer

関連項目:

出力形式:

- “\$LOGVSRw. 出力形式” (96 ページ)
- “\$LOGVSw. 出力形式” (95 ページ)

入力形式:

- “\$LOGVSRw. 入力形式” (336 ページ)

\$LOGVSRw. 入力形式

右から左に論理的順序で並ぶ文字列を読み込み、その文字列を視覚的順序に変換します。

カテゴリ: BIDI テキストの操作

構文

\$LOGVSRw.

説明

w
入力幅を指定します。
デフォルト: 200

範囲: 1–32000

比較

\$LOGVSRw.入力形式では、\$LOGVSw 入力形式とは逆の処理を行います。

例

次の例では、“flight”のヘブライ語の入力値 תִּירוֹט を使用します。

ステートメント	結果
	-----1-----+
x=input(' תִּירוֹט flight',\$logvsr12.); put x;	flight תִּירוֹט

次の例では、“computer”のアラビア語の入力値 تﺎﺗﺎﻧﺎ を使用します。

ステートメント	結果
	-----1-----+
x=input(' تﺎﺗﺎﻧﺎ computer',\$logvsr12.); put x;	تﺎﺗﺎﻧﺎ computer

関連項目:

出力形式:

- “\$LOGVSw. 出力形式” (95 ページ)
- “\$LOGVSRw. 出力形式” (96 ページ)

入力形式:

- “\$LOGVSw. 入力形式” (335 ページ)

MINGUOw. 入力形式

台湾の形式の日付を読み込みます。

カテゴリ: 日付と時間

構文

MINGUOw.

説明

w
 入力幅を指定します。
デフォルト: 6
範囲: 6–10

詳細

台湾の日付の通常の形式は、*yyyymmdd* です。

yyyy
 年を表す整数です。

mm
 月を表す 01 から 12 の整数です。

dd
 月の日を表す 01 から 31 の整数です。

台湾のカレンダーでは、1912 年を基準年(1912 年 1 月 1 日が 01/01/01)として使用します。1912 年より前の日付は無効です。年の値が 100 年周期で繰り返されることはありません。年の値は、継続して増加していきます。

年、月、日の値は、*YYMMDDw* 入力形式で使用可能な、空白、スラッシュ、ダッシュなどの区切り文字で区切ることができます。区切り文字を使用する場合は、すべての値の間に挿入します。区切り文字を省略する場合は、値が 10 未満の日または月の先頭に 0 を使用してください。

例

次の例では、入力値に異なる日付を使用します。

```
input date minguo10.;
put date date9.;

data _null_;
input date minguo10.;
put date date9.;
datalines;
49/01/01
891215
03-01-01
;
```

値	結果
	----+----1----+
49/01/01	01JAN1960
891215	15DEC2000
103-01-01	01JAN2014

関連項目:**出力形式:**

- “MINGUOW. 出力形式” (97 ページ)

入力形式:

- “YYMMDDw. Informat” in *SAS Formats and Informats: Reference*

NENGOW. 入力形式

eyymmdd 形式の日本の日付値を読み込みます。

カテゴリ: 日付と時間

構文

NENGOW.

説明

w

入力幅を指定します。

デフォルト: 10

範囲: 7–32

詳細

日本の日付の通常の形式は、eyymmdd です。

e

元号(明治、大正、昭和、平成)の先頭の文字(M、T、S、H)です。

yy

年を表す整数です。

mm

月を表す 01 から 12 の整数です。

dd

月の日を表す 01 から 31 の整数です。

e の値と整数はピリオドで区切ることができます。*e* を省略すると、現在の元号が使用されます。年、月、日の値は、空白や数値以外の文字で区切ることができます。ただし、区切り文字を使用する場合は、すべての値の間に挿入します。区切り文字を省略する場合は、値が 10 未満の日または月の先頭に 0 を使用してください。

例

次の例では、異なる入力値を使用します。

```
input nengo_date nengo8.;
put nengo_date date9.;
```

```

data _null_;
input nengo_date nengo8.;
put nengo_date date9.;
put nengo_date= ;
datalines;
h11108
h.11108
11/10/08
;

```

値	結果
	----+----1----+
h11108	08OCT1999
h.11108	08OCT1999
11/10/08	08OCT1999

関連項目:

出力形式:

- [“NENGOW. 出力形式” \(99 ページ\)](#)

NLDATEw. 入力形式

指定されたロケールの日付値を読み込み、その日付値をローカルの SAS 日付値に変換します。

カテゴリ: 日付と時間

構文

NLDATEw.

説明

w
 入力幅を指定します。
デフォルト: 20
範囲: 10–200

例

次の例では、入力値として February 24, 2003 を使用します。

ステートメント	結果
	----+----1----+

ステートメント	結果
<pre>options locale=English_UnitedStates; dy='February 24, 2003'; y=input('dy,nldate200.); put y;</pre>	15760
<pre>options locale=German_Germany; dy='24. Februar 2003'; y=input(dy,nldate16.); put y;</pre>	15760

関連項目:

出力形式:

- [“NLDATEw. 出力形式” \(102 ページ\)](#)

NLDATMw. 入力形式

指定されたロケールの日時値を読み込み、その日時値をローカルの SAS 日時値に変換します。

カテゴリ: 日付と時間

構文

NLDATM w .

説明

w
 入力幅を指定します。
デフォルト: 30
範囲: 10–200

例

次の例では、入力値として February 24, 2003 12:39:43 を使用します。

ステートメント	結果
<pre>options locale=English_UnitedStates; y=input('24.Feb03:12:39:43', nldatm.); put y;</pre>	<pre>-----+-----1-----+ 1361709583</pre>

ステートメント	結果
<pre>options locale=German_Germany; y=input('24.Februar 2003 12.39 Uhr', nldatm.); put y=;</pre>	1330171200

関連項目:**出力形式:**

- “[NLDATMw. 出力形式](#)” (111 ページ)

NLMNIAEDw.d 入力形式

アラブ首長国連邦の国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIAED $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67',nlmniaed32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	-----1-----+

ステートメント	結果
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- “NLMNLAEDw.d 入力形式” (375 ページ)

NLMNIAUDw.d 入力形式

オーストラリアの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIAUD $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67',nlmniaud32.2);
```

```
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**入力形式:**

- [“NLMNLAUDw.d 入力形式” \(376 ページ\)](#)

NLMNIBGNw.d 入力形式

ブルガリアの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIBGNw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=put (' (-1234.56789) ',nlmnibgn32.2);
```

```
y=put (' (-1234.56789) ',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**入力形式:**

- [“NLMNLBGNw.d 入力形式” \(377 ページ\)](#)

NLMNIBRLw.d 入力形式

ブラジルの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIBRLw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d
10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnibr132.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNLBRLw.d 入力形式” \(378 ページ\)](#)

NLMNICADw.d 入力形式

カナダの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNICAD $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnicad32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNICAD \$w.d\$ 出力形式” \(127 ページ\)](#)

NLMNICHF $w.d$ 入力形式

リヒテンシュタインとスイスの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNICHF $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input'($12,345.67)',nlmnicf32.2);
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNICHFw.d 出力形式” \(128 ページ\)](#)

NLMNICNYw.d 入力形式

中国の国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNICNY $w.d$

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input'($12,345.67)',nlmnicny32.2);
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNICNYw.d 出力形式” (129 ページ)

NLMNICZKw.d 入力形式

チェコ共和国の国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNICZKw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmniczk32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- “NLMNLCZKw.d 入力形式” (381 ページ)

NLMNIDKKw.d 入力形式

デンマーク、フェロー諸島およびグリーンランドの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIDKKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmndkk32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNIDKKw.d 出力形式” \(131 ページ\)](#)

NLMNIEEKw.d 入力形式

エストニアの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIEEKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnieek32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNLEEKw.d 入力形式” \(383 ページ\)](#)

NLMNIEGPw.d 入力形式

エジプトの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIEGPw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmniegp32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- “NLMNLEGPw.d 入力形式” (384 ページ)

NLMNIEURw.d 入力形式

オーストリア、ベルギー、フィンランド、フランス、ドイツ、ギリシャ、アイルランド、イタリア、ルクセンブルク、マルタ、オランダ、ポルトガル、スロベニア、スペインの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIEURw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input'($12,345.67)',nlmnieur32.2);
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNIEURw.d 出力形式” (133 ページ)

NLMNIGBPw.d 入力形式

英国の国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIGBPw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67','nlmnigbp32.2);
y=input('$12,345.67','dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNIGBPw.d 出力形式” (134 ページ)

NLMNIHKDw.d 入力形式

香港の国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIHKDw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d
 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input'($12,345.67)',nlmnikd32.2);
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- “NLMNIHKDw.d 出力形式” (135 ページ)

NLMNIHRKw.d 入力形式

クロアチアの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIHRKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnihrk32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNLHRKw.d 入力形式” \(388 ページ\)](#)

NLMNIHUFw.d 入力形式

ハンガリーの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIHUF $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnihof32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNLHUF \$w.d\$ 入力形式” \(389 ページ\)](#)

NLMNIIDR $w.d$ 入力形式

インドネシアの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIIDR $w.d$

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67',nlmniidr32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**入力形式:**

- “NLMNLIDRw.d 入力形式” (390 ページ)

NLMNIILSw.d 入力形式

イスラエルの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIILSw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67','nlmniils32.2);
y=input('$12,345.67')dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- “NLMNIILSw.d 出力形式” (139 ページ)

NLMNIINR*w.d* 入力形式

インドの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIINR*w.d*

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmniinr32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNLINRw.d 入力形式” \(391 ページ\)](#)

NLMNIJPYw.d 入力形式

日本の国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIJPYw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmniipy32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNIJPYw.d 出力形式” \(141 ページ\)](#)

NLMNIKRWw.d 入力形式

韓国の国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIKRWw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input'($12,345.67)',nlmnikrw32.2);
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNLKRWw.d 入力形式” \(393 ページ\)](#)

NLMNLTlw.d 入力形式

リトアニアの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLTlw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67',nlmniltl32.2);
```

```
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**入力形式:**

- [“NLMNLLTLw.d 入力形式” \(394 ページ\)](#)

NLMNILVLw.d 入力形式

ラトビアの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNILVLw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnilv132.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**入力形式:**

- [“NLMNLLVLw.d 入力形式” \(395 ページ\)](#)

NLMNIMOPw.d 入力形式

マカオの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIMOPw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d
10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnimop32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNLMOPw.d 入力形式” \(396 ページ\)](#)

NLMNIMXNw.d 入力形式

メキシコの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIMXN $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnmixn32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNLMXN \$w.d\$ 入力形式” \(397 ページ\)](#)

NLMNIMYR $w.d$ 入力形式

マレーシアの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIMYRw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input{ '$12,345.67' ,nlmnimyr32.2};
y=input( '$12,345.67' ,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNIMYRw.d 出力形式” \(146 ページ\)](#)

NLMNINOKw.d 入力形式

ノルウェーの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNINOKw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input'($12,345.67)',nlmminok32.2);
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- [“NLMNINOKw.d 出力形式” \(147 ページ\)](#)

NLMNINZDw.d 入力形式

ニュージーランドの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNINZDw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmminzd32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- “NLMNINZDw.d 出力形式” (148 ページ)

NLMNIPLNw.d 入力形式

ポーランドの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIPLNw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input'($12,345.67)',nlmnipln32.2);
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- “[NLMNIPLNw.d 出力形式](#)” (149 ページ)

NLMNIRUBw.d 入力形式

ロシアの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIRUBw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input'($12,345.67)',nlmnirub32.2);
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNIRUBw.d 出力形式” \(150 ページ\)](#)

NLMNISEKw.d 入力形式

スウェーデンの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNISEKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input'($12,345.67)',nlmnisek32.2);
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- “[NLMNISEKw.d 出力形式](#)” (151 ページ)

NLMNISGDw.d 入力形式

シンガポールの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNISGDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input'($12,345.67)',nlmnisgd32.2);
```

```
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNISGDw.d 出力形式” (151 ページ)

NLMNITHBw.d 入力形式

タイの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNITHBw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlthb32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**入力形式:**

- “NLMNLTHBw.d 入力形式” (404 ページ)

NLMNITRYw.d 入力形式

トルコの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNITRYw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d
必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnitry32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNLTRYw.d 入力形式” \(405 ページ\)](#)

NLMNITWDw.d 入力形式

台湾の国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNITWD $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67','nlmnitwd32.2');
y=input('$12,345.67','dollar32.2');
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNITWDw.d 出力形式” \(154 ページ\)](#)

NLMNIUSDw.d 入力形式

プエルトリコと米国の国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIUSD $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67','nlmniusd32.2');
y=input('$12,345.67','dollar32.2');
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- “NLMNIUSD $w.d$ 出力形式” (155 ページ)

NLMNIZAR $w.d$ 入力形式

南アフリカの国際表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNIZAR $w.d$

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input'($12,345.67)',nlmnizar32.2);
y=input'($12,345.67)'dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNIZARw.d 出力形式” (156 ページ)

NLMNLAEDw.d 入力形式

アラブ首長国連邦のローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLAEDw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlad32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- “NLMNIAEDw.d 入力形式” (342 ページ)

NLMNLAUDw.d 入力形式

オーストラリアのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLAUDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlaid32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNLAUDw.d 出力形式” \(158 ページ\)](#)

NLMNLBGNw.d 入力形式

ブルガリアのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLBGNw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input(-12345.67,nlmnlbgn32.2);
y=input(-12345.67,dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNIBGNw.d 入力形式” \(344 ページ\)](#)

NLMNLBRLw.d 入力形式

ブラジルのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLBRLw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlbrl32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- “NLMNIBRLw.d 入力形式” (345 ページ)

NLMNLCADw.d 入力形式

カナダのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLCADw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('($12,345.67)',nlmnlcad32.2);
```

```
y=input('($12,345.67)',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNLCADw.d 出力形式” (160 ページ)

NLMNLCHFw.d 入力形式

リヒテンシュタインとスイスのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLCHFw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlchf32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNLCHFw.d 出力形式” (161 ページ)

NLMNLCNYw.d 入力形式

中国のローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLCNYw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlcny32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- “NLMNLCNYw.d 出力形式” (162 ページ)

NLMNLCZKw.d 入力形式

チェコ共和国のローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLCZK $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlczk32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNICZK \$w.d\$ 入力形式” \(348 ページ\)](#)

NLMNLDKK $w.d$ 入力形式

デンマーク、フェロー諸島およびグリーンランドのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLDKKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmldkk32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNLDKKw.d 出力形式” \(164 ページ\)](#)

NLMNLEEKw.d 入力形式

エストニアのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLEEKw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67',nlmleek32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**入力形式:**

- “NLMNIEEKw.d 入力形式” (350 ページ)

NLMNLEGPw.d 入力形式

エジプトのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLEGPw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlegp32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNIEGPw.d 入力形式” \(351 ページ\)](#)

NLMNLEURw.d 入力形式

オーストリア、ベルギー、フィンランド、フランス、ドイツ、ギリシャ、アイルランド、イタリア、ルクセンブルク、マルタ、オランダ、ポルトガル、スロベニア、スペインのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLEURw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlleur32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNLEURw.d 出力形式” \(167 ページ\)](#)

NLMNLGBPw.d 入力形式

英国のローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLGBPw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlgbp32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNLGBPw.d 出力形式” \(168 ページ\)](#)

NLMNLHKDw.d 入力形式

香港のローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLHKDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlhkd32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- “NLMNLHKDw.d 出力形式” (169 ページ)

NLMNLHRKw.d 入力形式

クロアチアのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLHRKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67',nlmnlhrk32.2);
```

```
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-1,234.57

関連項目:**入力形式:**

- “NLMNIHRKw.d 入力形式” (354 ページ)

NLMNLHUFw.d 入力形式

ハンガリーのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLHUFw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlhuf32.2);
```

```
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**入力形式:**

- “NLMNIHUFw.d 入力形式” (355 ページ)

NLMNLIDR*w.d* 入力形式

インドネシアのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLIDR*w.d*

説明

w
出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d
必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlidr32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- “NLMNIIDR*w.d* 入力形式” (356 ページ)

NLMNLILSw*d* 入力形式

イスラエルのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLILSw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlils32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNLILSw.d 出力形式” \(172 ページ\)](#)

NLMNLINRw.d 入力形式

インドのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLINR $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlr32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- “NLMNIINR $w.d$ 入力形式” (358 ページ)

NLMNLJPY $w.d$ 入力形式

日本のローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLJPY $w.d$

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67',nlmnljpy32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNLJPYw.d 出力形式” (174 ページ)

NLMNLKRWw.d 入力形式

韓国のローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLKRWw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlkrw32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNIKRWw.d 入力形式” \(360 ページ\)](#)

NLMNLLTLw.d 入力形式

リトアニアのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLLTLw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnl1t132.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNLLVLw.d 入力形式” \(361 ページ\)](#)

NLMNLLVLw.d 入力形式

ラトビアのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLLVLw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnl1v132.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- “NLMNMLVLw.d 入力形式” (362 ページ)

NLMNLMOPw.d 入力形式

マカオのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLMOPw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlmop32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- “NLMNIMOPw.d 入力形式” (363 ページ)

NLMNLMXNw.d 入力形式

メキシコのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLMXNw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('($12,345.67)',nlmnlmxn32.2);
```

```
y=input('($12,345.67)',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**入力形式:**

- “NLMNIMXNw.d 入力形式” (363 ページ)

NLMNLMYRw.d 入力形式

マレーシアのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLMYRw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlmyr32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNLMYRw.d 出力形式” (179 ページ)

NLMNZNOKw.d 入力形式

ノルウェーのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNZNOKw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d
必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlnok32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNZNOKw.d 出力形式” \(180 ページ\)](#)

NLMNZNZDw.d 入力形式

ニュージーランドのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLNZD $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlz32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNLNZD \$w.d\$ 出力形式” \(181 ページ\)](#)

NLMNLPLN $w.d$ 入力形式

ポーランドのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLRUBw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67',nlmnlpln32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNLRUBw.d 出力形式” \(182 ページ\)](#)

NLMNLRUBw.d 入力形式

ロシアのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLRUBw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67',nlmnlrub32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNLRUBw.d 出力形式” (183 ページ)

NLMNLSEKw.d 入力形式

スウェーデンのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLSEKw.d

説明**w**

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlsek32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNLSSEKw.d 出力形式” \(184 ページ\)](#)

NLMNLSGDw.d 入力形式

シンガポールのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLSGDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlsgd32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNLSGDw.d 出力形式” \(185 ページ\)](#)

NLMNLTHBw.d 入力形式

タイのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLTHBw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlthb32.2);
y=input('$12,345.67',dollar32.2);
```


ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- [“NLMNITHBw.d 入力形式” \(371 ページ\)](#)

NLMNLTRYw.d 入力形式

トルコのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLTRYw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnltry32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+

ステートメント	結果
put x=;	-12345.67
put y=;	-12345.67

関連項目:

入力形式:

- “NLMNITRYw.d 入力形式” (372 ページ)

NLMNLTWDw.d 入力形式

台湾のローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLTWDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されていません。

```
x=input('$12,345.67',nlmnltd32.2);
```

```
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNLUSDw.d 出力形式” (187 ページ)

NLMNLUSDw.d 入力形式

プエルトリコと米国のローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLUSDw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlusd32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+-----1-----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:**出力形式:**

- “NLMNLUSDw.d 出力形式” (188 ページ)

NLMNLZARw.d 入力形式

南アフリカのローカル表示形式で通貨データを読み込みます。

カテゴリ: 数値

配置: 左

構文

NLMNLZARw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 9

範囲: 1–32

d
必要に応じて、 10^d で数値を除算するよう指定します。データに小数点が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

例

次の例では、LOCALE=システムオプションが English_UnitedStates に設定されています。

```
x=input('$12,345.67',nlmnlzar32.2);
y=input('$12,345.67',dollar32.2);
```

ステートメント	結果
	----+----1----+
put x=;	-12345.67
put y=;	-12345.67

関連項目:

出力形式:

- [“NLMNLZARw.d 出力形式” \(189 ページ\)](#)

NLMNYw.d 入力形式

指定されたロケールの通貨データをローカル表示形式で読み込み、そのデータを数値に変換します。

カテゴリ: 数値

構文

NLMNYw.d

説明

w

入力幅を指定します。

デフォルト: 9

範囲: 1–32

d

必要に応じて、数値を 10^d で除算するかどうかを指定します。データに小数点区切り文字が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0–31

詳細

NLMNYw.d 入力形式では、指定されたロケールの通貨データをローカル表示形式で読み込み、そのデータを数値に変換します。入力データから、桁区切り文字、小数点区切り文字、空白、通貨記号、閉じかっこを削除します。

比較

NLMNYw.d 入力形式では、NLMNYIw.d 入力形式とは逆の処理を行います。

NLMNYw.d 入力形式は、DOLLARw.d 入力形式と似ていますが、NLMNYw.d 入力形式はロケール固有である点が違います。

例

次の例では、入力値として\$12,345.67を使用します。

ステートメント	結果
	-----1-----+
options LOCALE=English_UnitedStates;	-12345.67
x=input('(\$12,345.67)',nlmny32.2);	-12345.67
y=input('(\$12,345.67)',dollar32.2);	
put x=;	
put y=;	

関連項目:

出力形式:

- “NLMNYw.d 出力形式” (190 ページ)
- “NLMNYIw.d 出力形式” (191 ページ)

入力形式:

- “NLMNYIw.d 入力形式” (410 ページ)

NLMNYIw.d 入力形式

指定されたロケールの通貨データを国際表示形式で読み込み、そのデータを数値に変換します。

カテゴリ: 数値

構文

NLMNYIw.d

説明

w

入力幅を指定します。

デフォルト: 9

範囲: 1-32

d

必要に応じて、数値を 10^d で除算するかどうかを指定します。データに小数点区切り文字が含まれる場合、d 値は無視されます。

デフォルト: 0

範囲: 0-31

詳細

NLMNYIw.d 入力形式では、指定されたロケールの通貨データを国際表示形式で読み込み、そのデータを数値に変換します。入力データから、桁区切り文字、小数点区切り文字、空白、通貨記号、閉じかっこを削除します。

比較

NLMNYIw.d 入力形式では、NLMNYw.d 入力形式とは逆の処理を行います。

例

次の例では、入力値として 12,345.67 を使用します。

ステートメント	結果
	----+----1-----+
options LOCALE=English_UnitedStates;	-12345.67
x=input(' (USD12,345.67) ',nlmnyi32.2);	-12345.67
y=input('\$-12,345.67)',dollar32.2);	
put x=;	
put y=;	

関連項目:**出力形式:**

- “NLMNYw.d 出力形式” (190 ページ)
- “NLMNYIw.d 出力形式” (191 ページ)

入力形式:

- “NLMNYw.d 入力形式” (408 ページ)

NLNUMw.d 入力形式

指定されたロケールの数値データをローカル表示形式で読み込み、そのデータを数値に変換します。

カテゴリ: 数値

構文

NLNUMw.d

説明

w

入力幅を指定します。

デフォルト: 6

範囲: 1–32

d

必要に応じて、数値を 10^d で除算するかどうかを指定します。データに小数点区切り文字が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

詳細

NLNUMw.d 入力形式 では、指定されたロケールの数値データをローカル表示形式で読み込み、そのデータを数値に変換します。入力データから、桁区切り文字、小数点区切り文字、空白、通貨記号、閉じかっこを削除します。

比較

NLNUMw.d 入力形式 では、NLNUMIw.d 入力形式とは逆の処理を行います。

例

次の例では、入力値として-1234356.78 を使用します。

ステートメント	結果
	-----1-----

ステートメント	結果
<pre>options locale=English_UnitedStates; x=input ('-1,234,356.78',nlum32.2); put x=;</pre>	-1234356.78

関連項目:

出力形式:

- [“NLNUMw.d 出力形式” \(193 ページ\)](#)
- [“NLMNYIw.d 出力形式” \(191 ページ\)](#)

入力形式:

- [“NLNUMIw.d 入力形式” \(412 ページ\)](#)

NLNUMIw.d 入力形式

指定されたロケールの数値データを国際表示形式で読み込み、そのデータを数値に変換します。

カテゴリ: 数値

構文

NLNUMIw.d

説明

w

入力幅を指定します。

デフォルト: 6

範囲: 1–32

d

必要に応じて、 10^d で数値を除算するよう指定します。データに小数点区切り文字が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0–31

詳細

NLNUMIw.d 入力形式 では、指定されたロケールの数値データを国際表示形式で読み込み、そのデータを数値に変換します。入力データから、桁区切り文字、小数点区切り文字、空白、通貨記号、閉じかっこを削除します。

比較

NLNUMIw.d 入力形式 では、NLNUMw.d 入力形式とは逆の処理を行います。

例

次の例では、入力値として-1,234,356.78 を使用します。

ステートメント	結果
	----+-----1-----+
options locale=English_UnitedStates;	-1234356.78
x=input('-1,234,356.78', nlnumi32.2);	
put x=;	

関連項目:

出力形式:

- [“NLNUMw.d 出力形式” \(193 ページ\)](#)
- [“NLNUMIw.d 出力形式” \(194 ページ\)](#)
- [“NLNUMw.d 入力形式” \(411 ページ\)](#)

NLPCTw.d 入力形式

指定されたロケールの百分率データをローカル表示形式で読み込み、そのデータを数値に変換します。

カテゴリ: 数値

構文

NLPCTw.d

説明

w

入力幅を指定します。

デフォルト: 6

範囲: 1-32

d

必要に応じて、数値を 10^d で除算するかどうかを指定します。データに小数点区切り文字が含まれる場合、*d* 値は無視されます。

デフォルト: 0

範囲: 0-31

詳細

NLPCTw.d 入力形式では、指定されたロケールの百分率データをローカル表示形式で読み込み、そのデータを数値に変換します。値を 100 で除算し、桁区切り文字、小数点区切り文字、空白、パーセント記号、閉じかっこを入力データから削除します。

比較

NLPCT $w.d$ 入力形式 では、NLPCT $w.d$ 入力形式とは逆の処理を行います。
NLPCT $w.d$ 入力形式 は、PERCENT $w.d$ 入力形式と似ていますが、NLPCT $w.d$ 入力形式は ロケール固有である点が違います。

例

次の例では、入力値として-12,345.67% を使用します。

ステートメント	結果
	----+----1-----+
options LOCALE=English_UnitedStates;	-123.4567
x=input ('-12,345.67%', nlpct32.2);	-123.4567
y=input ('(12,345.67%)', percent32.2);	
put x=;	
put y=;	

関連項目:

出力形式:

- “NLPCT $w.d$ 出力形式” (195 ページ)
- “NLPCT $w.d$ 出力形式” (197 ページ)
- “NLPCT $w.d$ 入力形式” (414 ページ)

NLPCT $w.d$ 入力形式

指定されたロケールの百分率データを国際表示形式で読み込み、そのデータを数値に変換します。

カテゴリ: 数値

構文

NLPCT $w.d$

説明

w

入力幅を指定します。

デフォルト: 6

範囲: 1-32

d

必要に応じて、数値を 10^d で除算するかどうかを指定します。データに小数点区切り文字が含まれる場合、 d 値は無視されます。

デフォルト: 0

範囲: 0-31

詳細

NLPCTIw.d 入力形式では、指定されたロケールの百分率データを国際表示形式で読み込み、そのデータを数値に変換します。値を 100 で除算し、桁区切り文字、小数点区切り文字、空白、パーセント記号、閉じかっこを入力データから削除します。

比較

NLPCTIw.d 入力形式では、NLPCTw.d 入力形式とは逆の処理を行います。

例

次の例では、入力値として-12,345.67%を使用します。

ステートメント	結果
	-----1-----+
options LOCALE=English_UnitedStates;	-123.4567
x=input('-12,345.67%',nlpct32.2);	-123.4567
y=input('(12,345.67%)',percent32.2);	
put x=;	
put y=;	

関連項目:

出力形式:

- “NLPCTw.d 出力形式” (195 ページ)
- “NLPCTIw.d 出力形式” (197 ページ)

入力形式:

- “NLPCTw.d 入力形式” (413 ページ)

NLTIMAPw. 入力形式

指定されたロケールの時間値を a.m.および p.m.表示を使用して読み込み、その時間値をローカルの SAS 時間値に変換します。

カテゴリ: 日付と時間

構文

NLTIMAPw.

説明

w
 入力幅を指定します。
デフォルト: 10
範囲: 4–200

例

次の例では、入力値として 04:24:43 p.m.を使用します。

ステートメント	結果
	-----1-----
options locale=English_UnitedStates; y=input('04:24:43 PM',nltimap11.); put y time.;	16:24:43
options locale=German_Germany; y=input('16.24 Uhr',nltimap11.); put y time.;	16:24:00

関連項目:**出力形式:**

- [“NLTIMAPw. 出力形式” \(204 ページ\)](#)

NLTIMEW. 入力形式

指定されたロケールの時間値を読み込み、その時間値をローカルの SAS 時間値に変換します。

カテゴリ: 日付と時間

構文

NLTIMEW.

説明

w
 入力幅を指定します。
デフォルト: 20
範囲: 10–200

例

次の例では、入力値として 16:24:43 を使用します。

ステートメント	結果
	-----1-----
options locale=English_UnitedStates; y=input('16:24:43',nltime.); put y time.;	16:24:43
options locale=German_Germany; y=input('16.24 Uhr',nltime.); put y time.;	16:24:00

関連項目:

出力形式:

- “NLTIMEw. 出力形式” (205 ページ)

\$REVERJw. 入力形式

文字データを右から左の順に読み込み、空白を保持します。

カテゴリ: 文字

構文

\$REVERJ w .

説明

w

入力幅を指定します。

デフォルト: w が指定されていない場合は 1

範囲: 1–32767

比較

\$REVERJ w . 入力形式は \$REVER s_w . 入力形式とほぼ同じですが、\$REVER s_w . 入力形式では先頭の空白がすべて削除され、結果を左詰めにします。

例

次の例では、入力値として ABCD を使用します。

```
input @1 name $reverj7.;
```

値	結果
	-----1

値	結果
ABCD	###DCBA
ABCD	DCBA###*

* #文字は空白を表します。

関連項目:

入力形式:

- “\$REVERSw. 入力形式” (418 ページ)

\$REVERSw. 入力形式

文字データを右から左の順に左詰めで読み込みます。

カテゴリ: 文字

構文

\$REVERSw.

説明

w

入力幅を指定します。

デフォルト: *w* が指定されていない場合は 1

範囲: 1–32767

比較

\$REVERSw. 入力形式は \$REVERJw. 入力形式とほぼ同じですが、\$REVERJw. 入力形式では先頭と末尾の空白がすべて保持されます。

例

次の例では、入力値として ABCD を使用します。

```
input @1 name $revers7.;
```

値	結果
	----+-----1
ABCD	DCBA###
ABCD	DCBA###*

* #文字は空白を表します。

関連項目:**入力形式:**

- “\$REVERJw. 入力形式” (417 ページ)

\$UCS2Bw. 入力形式

ビッグエンディアン、16 ビット、UCS2 形式の UNICODE(ユニコード)文字列を読み取り、その文字列を現在の SAS セッションのエンコーディングに変換します。

カテゴリ: 文字

構文

\$UCS2Bw.

説明

w

入力幅を指定します。16 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 2–32000

比較

\$UCS2Bw.入力形式では、\$UCS2BEw.入力形式とは逆の処理を行います。同一の動作環境内のデータを処理する場合は、\$UCS2Xw.入力形式を使用します。異なる動作環境のデータを処理する場合は、\$UCS2Bw.入力形式や\$UCS2Lw.入力形式を使用します。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
x=input('5927'x,\$ucs2b.); put x=\$hex4.;	x=91e5

関連項目:**出力形式:**

- “\$UCS2Bw. 出力形式” (206 ページ)
- “\$UCS2Lw. 出力形式” (208 ページ)

- “\$UCS2Xw. 出力形式” (211 ページ)
- “\$UTF8Xw. 出力形式” (226 ページ)

入力形式:

- “\$UCS2Lw. 入力形式” (421 ページ)
- “\$UCS2Xw. 入力形式” (423 ページ)
- “\$UTF8Xw. 入力形式” (436 ページ)

\$UCS2BEw. 入力形式

現在の SAS セッションのエンコーディングの文字列を読み込み、その文字列をビッグエンディアン、16 ビット、UCS2 形式の UNICODE(ユニコード)に変換します。

カテゴリ: 文字

構文

\$UCS2BEw.

説明

w

入力幅を指定します。16 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 1–32000

比較

\$UCS2BEw.入力形式では、\$UCS2BEw.入力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
<pre>ucs2str=input ('大', \$ucs2be2.); put ucs2str=\$hex4.;</pre>	ucs2str=2020

関連項目:

出力形式:

- “\$UCS2Bw. 出力形式” (206 ページ)
- “\$UCS2BEw. 出力形式” (207 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)

\$UCS2Lw. 入力形式

リトルエンディアン、16 ビット、UCS2 形式の UNICODE(ユニコード)文字列を読み取り、その文字列を現在の SAS セッションのエンコーディングに変換します。

カテゴリ: 文字

構文

\$UCS2Lw.

説明

w

入力幅を指定します。16 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 2–32000

比較

\$UCS2Lw.入力形式では、\$UCS2LEw.入力形式とは逆の処理を行います。同一の動作環境内のデータを処理する場合は、\$UCS2Xw.入力形式を使用します。異なる動作環境のデータを処理する場合は、\$UCS2Bw.入力形式や\$UCS2Lw.入力形式を使用します。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
x=input('2759'x,\$ucs2l.);	x=91e5
put x=\$hex4.;	

関連項目:**出力形式:**

- “\$UCS2Bw. 出力形式” (206 ページ)

- “\$UCS2Lw. 出力形式” (208 ページ)
- “\$UCS2Xw. 出力形式” (211 ページ)
- “\$UTF8Xw. 出力形式” (226 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2Xw. 入力形式” (423 ページ)
- “\$UTF8Xw. 入力形式” (436 ページ)

\$UCS2LEw. 入力形式

現在の SAS セッションのエンコーディングにある文字列を読み込み、その文字列をリトルエンディアン、16 ビット、UCS2 形式の UNICODE(ユニコード)に変換します。

カテゴリ: 文字

構文

\$UCS2LEw.

説明

w

入力幅を指定します。16 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 1–32000

比較

\$UCS2LEw.入力形式では、\$UCS2LEw.入力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
<pre>ucs2str=input ('大', \$ ucs2le2.); put ucs2str=\$hex4;</pre>	ucs2str=2759

関連項目:

出力形式:

- “\$UCS2Lw. 出力形式” (208 ページ)
- “\$UCS2LEw. 出力形式” (209 ページ)

入力形式:

- “\$UCS2Lw. 入力形式” (421 ページ)

\$UCS2Xw. 入力形式

16 ビット、UCS2 形式の UNICODE(ユニコード)文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換します。

カテゴリ: 文字

構文

\$UCS2Xw.

説明

w

出力フィールドの幅を指定します。16 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 2–32000

比較

\$UCS2Xw.入力形式では、\$UCS2XEw.入力形式とは逆の処理を行います。同一の動作環境内のデータを処理する場合は、\$UCS2Xw.入力形式を使用します。異なる動作環境のデータを処理する場合は、\$UCS2Bw.入力形式や\$UCS2Lw.入力形式を使用します。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。リトルエンディアン形式を使用します。

ステートメント	結果
	----+-----1-----+
x=input('5927'x,\$ucs2x.);	x=91e5
put x=\$hex4.;	

関連項目:**出力形式:**

- “\$UCS2Bw. 出力形式” (206 ページ)

- “\$UCS2Lw. 出力形式” (208 ページ)
- “\$UCS2Xw. 出力形式” (211 ページ)
- “\$UTF8Xw. 出力形式” (226 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2Lw. 入力形式” (421 ページ)
- “\$UTF8Xw. 入力形式” (436 ページ)

\$UCS2XEw. 入力形式

現在の SAS セッションのエンコーディングの文字列を読み込み、その文字列を、16 ビット、UCS2 形式の UNICODE(ユニコード)に変換します。

カテゴリ: 文字

構文

\$UCS2XEw.

説明

w

入力幅を指定します。16 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 1-32000

比較

\$UCS2XEw.入力形式では、\$UCS2Xw.入力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
<pre>ucs2str=input ('大', \$ ucs2xe2.); put ucs2str=\$hex6;</pre>	ucs2str=5927

関連項目:

出力形式:

- “\$UCS2Xw. 出力形式” (211 ページ)
- “\$UCS2XEw. 出力形式” (212 ページ)

入力形式:

- “\$UCS2Xw. 入力形式” (423 ページ)

\$UCS4Bw. 入力形式

ビッグエンディアン、32 ビット、UCS4 形式の UNICODE(ユニコード)文字列を読み取り、その文字列を現在の SAS セッションのエンコーディングに変換します。

カテゴリ: 文字

構文

\$UCS4Bw.

説明

w

入力幅を指定します。32 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 4

範囲: 4–32000

比較

同一の動作環境内のデータを処理する場合は、\$UCS4Xw.入力形式を使用します。異なる動作環境のデータを処理する場合は、\$UCS4Bw.入力形式や\$UCS4Lw.入力形式を使用します。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
z=put('Zero1',\$UCS4B20.);	Zero1
x=input(z,\$UCS4B20.);	
put x;	

関連項目:**出力形式:**

- “\$UCS4Bw. 出力形式” (213 ページ)

入力形式:

- “\$UCS4Lw. 入力形式” (426 ページ)
- “\$UCS4Xw. 入力形式” (427 ページ)

\$UCS4Lw. 入力形式

リトルエンディアン、32 ビット、UCS4 形式の UNICODE(ユニコード)文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換します。

カテゴリ: 文字

構文

\$UCS4Lw.

説明

w

入力幅を指定します。32 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 4

範囲: 4–32000

比較

同一の動作環境内のデータを処理する場合は、\$UCS4Xw.入力形式を使用します。異なる動作環境のデータを処理する場合は、\$UCS4Bw.入力形式や\$UCS4Lw.入力形式を使用します。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+----1----+----2----+----3----+
z=put(' .com', \$UCS4L16.);	2E000000630000006F0000006D000000
put z \$hex32.;	

関連項目:**出力形式:**

- “\$UCS4Lw. 出力形式” (215 ページ)

入力形式:

- “\$UCS4Bw. 入力形式” (425 ページ)

- “\$UCS4Xw. 入力形式” (427 ページ)

\$UCS4Xw. 入力形式

32 ビット、UCS4 形式の UNICODE(ユニコード)文字列を読み取り、その文字列を現在の SAS セッションのエンコーディングに変換します。

カテゴリ: 文字

構文

\$UCS4Xw.

説明

w

入力幅を指定します。32 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 4

範囲: 4–32000

比較

\$UCS4Xw.入力形式では、\$UCS4Xw.入力形式とは逆の処理を行います。同一の動作環境内のデータを処理する場合は、\$UCS4Xw.入力形式を使用します。異なる動作環境のデータを処理する場合は、\$UCS4Bw.入力形式と\$UCS4Lw.入力形式を使用します。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。リトルエンディアン形式を使用します。

ステートメント	結果
	----+-----1-----+
ucs4=put('91e5'x,\$ucs4x.);	ucs4=27590000
sjis=input(ucs4,\$ucs4x.);	sjis=91E52020
put ucs4=\$hex8. sjis=\$hex8.;	
run;	

関連項目:

出力形式:

- “\$UCS2Xw. 出力形式” (211 ページ)
- “\$UCS2Bw. 出力形式” (206 ページ)
- “\$UCS2Lw. 出力形式” (208 ページ)

- “\$UCS4Xw. 出力形式” (218 ページ)
- “\$UTF8Xw. 出力形式” (226 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2Lw. 入力形式” (421 ページ)
- “\$UTF8Xw. 入力形式” (436 ページ)

\$UCS4Xw. 入力形式

現在の SAS セッションのエンコーディングの文字列を読み込み、その文字列を、32 ビット、UCS4 形式の UNICODE(ユニコード)に変換します。

カテゴリ: 文字

構文

`$UCS4Xw.`

説明

`w`

入力幅を指定します。32 ビットサイズの UNICODE 文字が入る十分な幅を指定します。

デフォルト: 8

範囲: 1–32000

比較

`$UCS4Xw.`入力形式では、`$UCS4Xw.`入力形式とは逆の処理を行います。

例

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+----1----
<code>ucs4str=input ('大', \$ucs4xe2.);</code>	<code>ucs4str=00005927</code>
<code>put ucs4str=\$hex8;</code>	

関連項目:

出力形式:

- “\$UCS4Xw. 出力形式” (218 ページ)
- “\$UCS4XEw. 出力形式” (219 ページ)

入力形式:

- “\$UCS4Xw. 入力形式” (427 ページ)

\$UESCW. 入力形式

UESC 表記でエンコードされた文字列を読み取り、その文字列を現在の SAS セッションのエンコーディングに変換します。

カテゴリ: 文字

構文

\$UESCW.

説明

w

出力フィールドの幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

0 から 9、a から z、A から Z など、すべての動作環境で使用可能でない文字は、UESC 表記で表す必要があります。\$UESCW. 入力形式はネストできます。

比較

\$UESCW.入力形式では、\$UESCEw.入力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
x=input('¥u5927', \$uesc10.);	大
y=input('¥uu5927', \$uesc10.);	¥u5927
z=input('¥uuu5927', \$uesc10.);	¥uu5927
put x;	
put y;	
put z;	

関連項目:**出力形式:**

- “\$UESCW. 出力形式” (220 ページ)
- “\$UESCEW. 出力形式” (221 ページ)

入力形式:

- “\$UESCEW. 入力形式” (430 ページ)

\$UESCEw. 入力形式

現在の SAS セッションのエンコーディングを使用する文字列を読み取り、その文字列を UESC 表記に変換します。

カテゴリ: 文字

構文

\$UESCEw.

説明

w

入力幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

\$UESCEw.入力形式はネストできます。

比較

\$UESCEw.入力形式では、\$UESCW.入力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
<code>x=input('大', \$uesc10.);</code>	¥u5927
<code>y=input('¥u5927', \$uesc10.);</code>	¥uu5927
<code>z=input('¥uu5927', \$uesc10.);</code>	¥uuu5927
<code>put x y z;</code>	

関連項目:**出力形式:**

- “\$UESCw. 出力形式” (220 ページ)
- “\$UESCEw. 出力形式” (221 ページ)

入力形式:

- “\$UESCw. 入力形式” (429 ページ)

\$UNCRw. 入力形式

NCR 文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換します。

カテゴリ: 文字

構文

\$UNCRw.

説明

w

入力幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

入力文字列に含まれるのは、文字と NCR のみにする必要があります。各国語の文字は、NCR で表記する必要があります。

比較

\$UNCRw.入力形式では、\$UNCREw.入力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
<pre>x=input('&#22823;', \$uncr10.); y=input('abc', \$uncr10.); put X; put Y;</pre>	大 abc

関連項目:**出力形式:**

- “\$UNCRw. 出力形式” (222 ページ)
- “\$UNCREw. 出力形式” (223 ページ)

入力形式:

- “\$UNCREw. 入力形式” (432 ページ)

\$UNCREw. 入力形式

現在の SAS セッションのエンコーディングを使用する文字列を読み込み、その文字列を NCR に変換します。

カテゴリ: 文字

構文

\$UNCREw.

説明

w

入力幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

出力文字列は、プレーンテキストと NCR に変換されます。各国語の文字は、NCR に変換されます。

比較

\$UNCREw.入力形式では、\$UNCRw.入力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
<pre>x=input ('大 abc', \$uncre12.); put x;</pre>	大abc

関連項目:**出力形式:**

- “\$UNCRw. 出力形式” (222 ページ)
- “\$UNCREw. 出力形式” (223 ページ)

入力形式:

- “\$UNCRw. 入力形式” (431 ページ)

\$SUPARENw. 入力形式

UPAREN 表記でエンコードされた文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換します。

カテゴリ: 文字

構文

\$SUPARENw.

説明

w

入力幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

SAS セッションのエンコーディングに対応する UNICODE 表記がない場合、その表記は現在の SAS セッションのエンコーディングのまま残ります。

比較

\$SUPARENw. 入力形式では、\$SUPARENw. informat.

例

この例では、UNIX オペレーティングシステムでサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
<code>v=input('<u0061>', \$uparen10.);</code>	a
<code>w=input('<u0062>', \$uparen10.);</code>	b
<code>x=input('<u0063>', \$uparen10.);</code>	c
<code>y=input('<u0033>', \$uparen10.);</code>	3
<code>z=input('<u5927>', \$uparen10.);</code>	太
<code>put v;</code>	
<code>put w;</code>	
<code>put x;</code>	
<code>put y;</code>	
<code>put z;</code>	

関連項目:

出力形式:

- “\$UPARENw. 出力形式” (224 ページ)
- “\$UPARENw. 出力形式” (225 ページ)

入力形式:

- “\$UPARENw. 入力形式” (434 ページ)
- “\$UPARENw. 入力形式” (435 ページ)

\$UPARENw. 入力形式

現在の SAS セッションのエンコーディングを使用する文字列を読み込み、その文字列を UPAREN 表記に変換します。

カテゴリ: 文字

構文

\$UPARENw.

説明

w
 入力幅を指定します。
 デフォルト: 8
 範囲: 1–32000

比較

\$UPARENw.入力形式では、\$UPARENw.入力形式とは逆の処理を行います。

例

この例では、UNIX オペレーティングシステムでサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
<code>v=input('a',\$uparen10.);</code>	<u0061>
<code>w=input('b',\$uparene10.);</code>	<u0062>
<code>x=input('c',\$uparene10.);</code>	<u0063>
<code>y=input('3',\$uparene10.);</code>	<u0033>
<code>z=input('大',\$uparen10.);</code>	<u5927>
<code>put v;</code>	
<code>put w;</code>	
<code>put x;</code>	
<code>put y;</code>	
<code>put z;</code>	

関連項目:

出力形式:

- “\$UPARENw. 出力形式” (224 ページ)
- “\$UPARENw. 出力形式” (225 ページ)

入力形式:

- “\$UPARENw. 入力形式” (433 ページ)
- “\$UPARENpw. 入力形式” (435 ページ)

\$UPARENpw. 入力形式

UPAREN 表記でエンコードされた文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換しますが、各国語文字は UPAREN 表記のエンコーディングのまま残ります。

カテゴリ: 文字

構文

\$UPARENpw.

説明

w

入力幅を指定します。

デフォルト: 8

範囲: 1–32000

詳細

UPAREN 表記に各国語文字が含まれ、その値が UNICODE の 0x00ff より大きい場合、その表記は UPAREN 表記のまま残ります。

例

この例では、UNIX オペレーティングシステムでサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
v=input ('<u0061>', \$uparen10.);	a
w=input ('<u0062>', \$uparenp10.);	b
x=input ('<u0063>', \$uparenp10.);	c
y=input ('<u0033>', \$uparenp10.);	3
z=input ('<u5927>', \$uparepn10.);	<u5927>
put v;	
put w;	
put x;	
put y;	
put z;	

関連項目:

出力形式:

- “\$UPARENw. 出力形式” (224 ページ)
- “\$UPARENEw. 出力形式” (225 ページ)

入力形式:

- “\$UPARENw. 入力形式” (433 ページ)
- “\$UPARENEw. 入力形式” (434 ページ)

\$UTF8Xw. 入力形式

UTF-8 でエンコードされた文字列を読み込み、その文字列を現在の SAS セッションのエンコーディングに変換します。

カテゴリ: 文字

構文

\$UTF8Xw.

説明

- w 入力幅を指定します。
デフォルト: 8
範囲: 1–32000

比較

この例では、UNIX 動作環境でサポートされる日本語(Shift_JIS)エンコーディングを使用します。

ステートメント	結果
	----+-----1-----+
<pre>x=input (' e5a4a7' x, \$utf8x3.); put x;</pre>	大

関連項目:

出力形式:

- “\$UCS2Bw. 出力形式” (206 ページ)
- “\$UCS2Lw. 出力形式” (208 ページ)
- “\$UCS2Xw. 出力形式” (211 ページ)
- “\$UTF8Xw. 出力形式” (226 ページ)

入力形式:

- “\$UCS2Bw. 入力形式” (419 ページ)
- “\$UCS2Lw. 入力形式” (421 ページ)
- “\$UCS2Xw. 入力形式” (423 ページ)

\$VSLOG_w. 入力形式

視覚的順序で並ぶ文字列を読み込み、その文字列を左から右の論理的順序に変換します。

カテゴリ: BIDI テキストの操作

構文

`$VSLOGw.`

説明

`w`

入力幅を指定します。

デフォルト: 200

範囲: 1–32000

比較

\$VSLOG_w.入力形式では、\$VSLOG_R_w.入力形式とは逆の処理を行います。

例

次の例では、“flight”のヘブライ語の入力値 **טוֹר** を使用します。

ステートメント	結果
	-----1-----+
<pre>x=input('טוֹר ','\$vslog12.');</pre>	טוֹר flight
<pre>put x;</pre>	

次の例では、“computer”のアラビア語の入力値 **كأنا** を使用します。

ステートメント	結果
	-----1-----+
<pre>x=input('كأنا computer','\$vslog12.');</pre>	كأنا computer
<pre>put x;</pre>	

関連項目:

出力形式:

- “\$VSLOGRw. 出力形式” (229 ページ)
- “\$VSLOGw. 出力形式” (228 ページ)

入力形式:

- “\$VSLOGRw. 入力形式” (438 ページ)

\$VSLOGRw. 入力形式

視覚的順序で並ぶ文字列を読み込み、その文字列を右から左の論理的順序に変換します。

カテゴリ: BIDI テキストの操作

構文

\$VSLOGRw.

説明

w 入力幅を指定します。

デフォルト: 200

範囲: 1–32000

比較

\$VSLOGRw.入力形式では、\$VSLOGRw.入力形式とは逆の処理を行います。

例

次の例では、“flight”のヘブライ語の入力値 **טוֹרָח** を使用します。

ステートメント	結果
	----+-----1-----+
<pre>x=input('טוֹרָח',\$vslogr12.); put x;</pre>	flight טוֹרָח

次の例では、“computer”のアラビア語の入力値 **كَمْبِيوتَر** を使用します。

ステートメント	結果
	----+-----1-----+
<pre>x=input('كَمْبِيوتَر computer',\$vslogr12.); put x;</pre>	كَمْبِيوتَر computer

関連項目:

出力形式:

- “\$VSLOGw. 出力形式” (228 ページ)
- “\$VSLOGRw. 出力形式” (229 ページ)

入力形式:

- “\$VSLOGw. 入力形式” (437 ページ)

YENw.d 入力形式

埋め込まれた円記号、カンマおよび小数点を削除します。

カテゴリ: 数値

構文

YENw.d

説明**w**

入力幅を指定します。

デフォルト: 1**範囲:** 1–32**d**

値を除算する 10 のべき乗を指定します。

要件 *d* は 0 または 2 でなければなりません。**ヒント:** *d* が 2 の場合、YEN*w.d* は小数点と小数点以下 2 桁を読み込みます。*d* が 0 の場合、YEN*w.d* は整数部分のみの値を読み込みます。**詳細**

円記号文字のコードの 16 進表現は、EBCDIC のシステムでは 5B、ASCII のシステムでは 5C です。これらのコードが表す通貨記号は、他の国では異なる場合があります。

例

次の例では、入力値として円を使用します。

input value yen10.2;

値	結果
	----+-----1-----+
¥1254.71	1254.71

関連項目:**出力形式:**

- “YEN*w.d* 出力形式” (235 ページ)

7 部

各国語サポート関連のマクロ関数

13 章	
マクロ関数エントリ	443

13 章 マクロ関数エントリ

カテゴリ別のマクロ	443
ディクショナリ	444
%KCOMPRES マクロ関数	444
%KINDEX マクロ関数	444
%KLEFT と%QKLEFT マクロ関数	445
%KLENGTH マクロ関数	445
%KSCAN と%QKSCAN 関数	446
%KSUBSTR と%QKSUBSTR マクロ関数	450
%KUPCASE と%QKUPCASE マクロ関数	452

カテゴリ別のマクロ

次の表に、SAS 各国語サポートマクロ関数の概要を示します。詳細については、それぞれの各国語サポート関連のマクロ関数に関する情報を参照してください。

カテゴリ	言語要素	説明
DBCS	%KCOMPRES マクロ関数 (p. 444)	複数の空白を圧縮し、先頭と末尾の空白を削除します。
	%KINDEX マクロ関数 (p. 444)	文字列の先頭文字の位置を返します。
	%KLEFT と%QKLEFT マクロ関数 (p. 445)	先頭の空白を削除し、引数を左詰めにします。
	%KLENGTH マクロ関数 (p. 445)	文字列の長さを返します。
	%KSCAN と%QKSCAN 関数 (p. 446)	文字列内の単語を位置指定して検索します。
	%KSUBSTR と%QKSUBSTR マクロ関数 (p. 450)	文字列の部分文字列を生成します。
	%KUPCASE と%QKUPCASE マクロ関数 (p. 452)	値を大文字に変換します。

ディクショナリ

%KCMPRES マクロ関数

複数の空白を圧縮し、先頭と末尾の空白を削除します。

カテゴリ: DBCS

種類: NLS マクロ関数

構文

`%KCMPRES (text | text expression)`

詳細

%KCMPRES マクロ関数は、複数の空白を圧縮し、先頭と末尾の空白を削除します。

%KINDEX マクロ関数

文字列の先頭文字の位置を返します。

カテゴリ: DBCS

種類: NLS マクロ関数

構文

`%KINDEX (source, string)`

必須引数

source

文字列またはテキスト式です。

string

文字列またはテキスト式です。

詳細

%KINDEX 関数は、*source* を検索して、最初に出現する *string* の先頭文字の位置を返します。*string* が見つからない場合、関数は 0 を返します。

例: 文字の検索

次のステートメントは、文字列で最初に出現する文字 *v* を検索します。

```
%let a=a very long value;
%let b=%kindex(&a,v);
%put v appears at position &b..;
```


ステートメントが実行されると、SAS ログに次の行が出力されます。

```
V appears at position 3.
```

%KLEFT と%QKLEFT マクロ関数

先頭の空白を削除し、引数を左詰めにします。

カテゴリ: DBCS
要件 MAUTOSOURCE システムオプション

構文

%KLEFT (*text* | *text expression*)

%QKLEFT (*text* | *text expression*)

詳細

%KLEFT と%QKLEFT マクロ関数は、先頭の空白を削除し、引数を左詰めにします。引数が特殊文字またはニーモニック演算子を含む場合は、%KQLEFT を使用します。

%KLEFT は、引数が引用符で囲まれている場合でも、引用符で囲まれていない結果を返します。%KQLEFT は、次の特殊文字とニーモニック演算子がマスクされた結果を生成します。そのためマクロプロセッサは、それらの結果を、マクロ言語の要素ではなくテキストとして解釈します。

```
& % ' " ( ) + - * / < > = ~ ^ ~ ; , # blank  
AND OR NOT EQ NE LE LT GE GT IN
```

%KLENGTH マクロ関数

文字列の長さを返します。

カテゴリ: DBCS
種類: NLS マクロ関数

構文

%KLENGTH (*character string* | *text expression*)

詳細

引数が文字列の場合、%KLENGTH は文字列の長さを返します。引数がテキスト式の場合、%KLENGTH は変換された値の長さを返します。引数が NULL 値の場合、%KLENGTH は 0 を返します。

例: 文字列の長さを返す

次のステートメントは、文字列とテキスト式の長さを返します。

```
%let a=Happy;  
%let b=Birthday;
```

```
%put The length of &a is %klength(&a).;
%put The length of &b is %klength(&b).;
%put The length of &a &b To You is %klength(&a &b to you).;
```

ステートメントが実行されると、SAS ログに次の行が出力されます。:

```
The length of Happy is 5.
The length of Birthday is 8.
The length of Happy Birthday To You is 21.
```

%KSCAN と%QKSCAN 関数

文字列内の単語を位置指定して検索します。

カテゴリ: DBCS

種類: NLS マクロ関数

構文

%KSCAN (*argument*, *n*<,charlist<,modifiers>>)

%QKSCAN (*argument*, *n*<,charlist<,modifiers>>)

必須引数

argument

文字列またはテキスト式です。*argument* が後述する特殊文字またはニーモニック演算子を含む場合は、%QKSCAN を使用します。

n

整数または整数を生成するテキスト式です。この整数は、単語の位置を表します。*n* が *argument* にある単語数より大きい場合、関数は NULL 文字列を返します。*n* が負の場合、%KSCAN は文字列の最後の単語から逆方向に検索して、単語を選択します。

charlist

文字リストを初期設定する任意の文字式を指定します。このリストによって、単語の区切り文字として使用される文字が特定されます。次の規則が適用されます。

- デフォルトでは、*charlist* のすべての文字が区切り文字として使用されます。
- *modifier* 引数に K 修飾子を指定した場合、*charlist* に含まれないすべての文字が区切り文字として使用されます。

ヒント: その他の修飾子を使用して、*charlist* に文字を追加できます。

modifier

文字定数、変数、式に、空白以外の文字を指定して、%KSCAN 関数の処理を変更できます。空白は無視されます。次の文字を修飾子として使用できます。

a または A 文字リストに英字を追加します。

b または B *count* 引数の符号に関係なく、逆方向(左から右方向ではなく、右から左方向)にスキャンします。

c または C 文字リストにコントロール文字を追加します。

- d または D 文字リストに数字を追加します。
- f または F 文字リストにアンダースコアと英字(`VALIDVARNAME=V7` を使用した SAS 変数名の有効先頭文字)を追加します。
- g または G 文字リストにグラフィカル文字を追加します。グラフィカル文字とは、印刷用紙にイメージを生成する文字です。
- h または H 文字リストに水平タブを追加します。
- i または I 文字の大文字小文字を無視します。
- k または K 文字リストに含まれないすべての文字を区切り文字として処理します。K を指定した場合、文字リストの文字は削除されずに、返される値に含まれます。K を指定しない場合、文字リストのすべての文字は区切り文字として処理されます。
- l または L 文字リストに小文字を追加します。
- m または M 連続する複数の区切り文字、および *string* 引数の先頭または末尾の区切り文字を長さゼロの単語として処理します。M 修飾子を指定しない場合、連続する複数の区切り文字は 1 つの区切り文字として処理され、*string* 引数の先頭または末尾の区切り文字は無視されます。
- n または N 文字リストに、数字、アンダースコア、英字(`VALIDVARNAME=V7` を使用した SAS 変数名に表示可能な文字)を追加します。
- o または O *charlist* と *modifier* 引数を、%KSCAN 関数の呼び出しごとに処理するのではなく、1 回のみ処理します。DATA ステップ(WHERE 句を除く)または SQL プロシジャで O 修飾子を使用した場合、*charlist* と *modifier* 引数が変更されないループ内で呼び出される %KSCAN をより高速に実行できます。O 修飾子は、SAS コード内の %KSCAN 関数のインスタンスごとに適用されます。このため、%KSCAN 関数のすべてのインスタンスで、同じ区切り文字と修飾子が使用されることはありません。
- p または P 文字リストに句読点を追加します。
- q または Q 引用符に囲まれた部分文字列内の区切り文字を無視します。*string* 引数の値がペアでない引用符を含む場合、左から右方向のスキャンと右から左方向のスキャンでは生成される単語が異なります。
- r または R %KSCAN は、単語から先頭と末尾の空白を削除して返します。Q と R の両方の修飾子を指定した場合、%KSCAN 関数は、最初に単語の先頭と末尾の空白を削除します。次に、引用符で始まる単語の場合は、この引用符のペアを単語から削除します。
- s または S 文字リストに空白文字を追加します(空白、水平タブ、垂直タブ、キャリッジリターン、ラインフィード、フォームフィード)。
- t または T *string* と *charlist* 引数から末尾の空白を取り除きます。末尾の空白を、両方の文字引数ではなく一方の文字引数からのみ削除するには、%KSCAN 関数に T 修飾子を指定するのではなく、TRIM 関数を使用してください。

u また 文字リストに小文字を追加します。
は U

w また 文字リストに印刷可能(書き込み可能)文字を追加します。
は W

x また 文字リストに 16 進文字を追加します。
は X

ヒント: 文字定数の *modifier* 引数は、引用符で囲みます。一組の引用符内に複数の修飾子を指定します。*modifier* 引数は、文字変数または文字式でも表すことができます。

詳細

%KSCAN と %QKSCAN 関数は、*argument* を検索して、*n* 番目の単語を返します。単語とは、1 つ以上の区切り文字によって区切られた 1 つ以上の文字のことです。

%KSCAN では、引数がマクロ引用関数によりマスクされていても、特殊文字とニーモニック演算子をマスクせずに値を返します。%QKSCAN は、次の特殊文字とニーモニック演算子をマスクした値を返します。

```
& % ' " ( ) + - * / < > = ~ ^ ~ ; , # blank
AND OR NOT EQ NE LE LT GE GT IN
```

区切り文字とは、単語区切りに使用される文字のことです。*charlist* と *modifier* 引数に区切り文字を指定できます。

Q 修飾子を指定した場合、引用符に囲まれた部分文字列内の区切り文字は無視されます。

%KSCAN 関数では、次の条件をすべて満たす部分文字列を単語とします。

- 左境界が、区切り文字または文字列の先頭である
- 右境界が、区切り文字または文字列の末尾である
- 区切り文字を含まない

単語は、その文字列の先頭または末尾に区切り文字がある場合、またはその文字列に 2 つ以上の連続した区切り文字が含まれる場合には長さが 0 となります。ただし、%KSCAN 関数に M 修飾子を指定しない場合、長さがゼロの単語は無視されます。

%KSCAN 関数に指定する引数が 2 つのみの場合、デフォルトの区切り文字は、コンピュータの使用文字が ASCII であるか EBCDIC であるかによって異なります。

- コンピュータの使用文字が ASCII の場合、次の文字がデフォルトの区切り文字です。

```
blank ! $ % & ( ) * + , - . / ; < ^ |
```

^文字が含まれない ASCII 環境では、%KSCAN 関数は代わりに~文字を使用します。

- コンピュータの使用文字が EBCDIC の場合、次の文字がデフォルトの区切り文字です。

```
blank ! $ % & ( ) * + , - . / ; < ~ | ¢|
```

区切り文字を指定せずに、*modifier* 引数を使用した場合、*modifier* 引数によって定義される区切り文字のみ使用されます。この場合、ASCII と EBCDIC 環境のデフォルトの区切り文字のリストは使用されません。つまり、修飾子は、*charlist* 引数により指定された区切り文字リストに追加されます。修飾子は、デフォルトの修飾子のリストには追加されません。

M 修飾子を指定した場合、文字列の単語数は、文字列の区切り文字数プラス 1 として定義されます。ただし、Q 修飾子を指定した場合、引用符内の区切り文字は無視されます。

M 修飾子を指定して、次のいずれかの条件が真の場合、%KSCAN 関数は長さがゼロの単語を返します。

- 文字列の開始文字が区切り文字、かつ、先頭の単語を要求した場合
- 文字列の終了文字が区切り文字、かつ、末尾の単語を要求した場合
- 2 つの連続する区切り文字が文字列に含まれ、かつ、その 2 つの区切り文字の間の単語を要求した場合

M 修飾子を指定しない場合、文字列の単語数は、連続する非区切り文字で構成される部分文字列の最大数として定義されます。ただし、Q 修飾子を指定した場合、引用符内の区切り文字は無視されます。

M 修飾子を指定しない場合、%KSCAN 関数は次の処理を行います。

- 文字列の開始または終了位置の区切り文字を無視します。
- 2 つ以上の連続する区切り文字を単一の区切り文字として処理します。

文字列に区切り文字のみ含まれる、または指定した count の絶対値が文字列の単語数より大きい場合、%KSCAN 関数は次のいずれかを返します。

- DATA ステップから%KSCAN 関数を呼び出す場合、単一の空白を返します。
- マクロプロセッサから%KSCAN 関数を呼び出す場合、長さがゼロの文字列を返します。

%KSCAN 関数では、文字引数に NULL を指定できます。NULL 引数は、長さがゼロの文字列として処理されます。数値引数には NULL を指定できません。

例: %KSCAN と%QKSCAN の処理の比較

次の例に、%KSCAN と%QKSCAN の処理を示します。

```
%macro a;
aaaaaa
%mend a;
%macro b;
bbbbbb
%mend b;
%macro c;
cccccc
%mend c;
%let x=%nrstr(%a*%b*%c);
%put X: &x;
%put The third word in X, with KSCAN: %kscan(&x,3,*);
%put The third word in X, with QKSCAN: %qkscan(&x,3,*);
```

%PUT ステートメントは、ログに次の行を出力します。

```
X: %a*%b*%c
The third word in X, with KSCAN: ccccc
The third word in X, with QKSCAN: %c
```

%KSUBSTR と%QKSUBSTR マクロ関数

文字列の部分文字列を生成します。

カテゴリ: DBCS

種類: NLS マクロ関数

構文

`%KSUBSTR (argument, position<, length>)`

`%QKSUBSTR (argument, position<, length>)`

必須引数

argument

文字列またはテキスト式です。*argument* が後述する特殊文字またはニーモニック演算子を含む場合は、%QKSUBSTR を使用します。

position

整数または整数を生成する(テキスト、論理、算術)式です。この整数は、部分文字列の先頭文字の位置を表します。*position* が文字列の文字数より大きい場合、%KSUBSTR と%QKSUBSTR は警告メッセージを発行して、NULL 値を返します。

length

整数または整数を生成する(テキスト、論理、算術)式です(オプション)。この整数は、部分文字列の文字数を表します。*length* が *argument* 内の *position* 以降の文字数より大きい場合、%KSUBSTR と%QKSUBSTR は警告メッセージを発行して、*position* から文字列の末尾までの文字で構成される部分文字列を返します。デフォルトで、%KSUBSTR と%QKSUBSTR は *position* から文字列の末尾までの文字で構成される文字列を作成します。

詳細

%KSUBSTR と%QKSUBSTR 関数は、*argument* 内の *position* 位置から *length* の文字数分の文字で構成される部分文字列を生成します。

%KSUBSTR は、特殊文字とニーモニック演算子をマスクせずに値を返します。

%QKSUBSTR は、次の特殊文字とニーモニック演算子をマスクします。

```
& % ' " ( ) + - * / < > = ~ ^ ~ ; , # blank
AND OR NOT EQ NE LE LT GE GT IN
```

例

例 1: ファイル参照名を 8 文字に制約する

マクロ MAKEFREF は%KSUBSTR を使用して、ユーザーが割り当てたパラメータが 8 文字より長い場合、先頭の 8 文字をファイル参照名に割り当てます。

```
%macro makefref(fileref,file);
%if %length(&fileref) gt 8 %then
%let fileref = %substr(&fileref,1,8);
filename &fileref "&file";
```

```
%mend makehref;
%makehref (humanresource, /dept/humanresource/report96)
```

SAS は、次のステートメントを読み込みます。

```
FILENAME HUMANRES "/dept/humanresource/report96";
```

例 2: セグメントの長いマクロ変数値の保存

マクロ SEPMSG は、マクロ変数 MSG の値を 40 文字単位に分割して、単位ごとに異なる変数に保存します。

```
%macro sepmsg(msg);
%let i=1;
%let start=1;
%if %length(&msg)>40 %then
%do;
%do %until(%klength(&&msg&i)<40);
%let msg&i=%qksubstr(&msg,&start,40);
%put Message &i is: &&msg&i;
%let i=%eval(&i+1);
%let start=%eval(&start+40);
%let msg&i=%qksubstr(&msg,&start);
%end;
%put Message &i is: &&msg&i;
%end;
%else %put No subdivision was needed.;
%mend sepmsg;
%sepmsg(%nrstr(A character operand was found in the %EVAL function
or %IF condition where a numeric operand is required. A character
operand was found in the %EVAL function or %IF condition where a
numeric operand is required.));
```

このプログラムが実行されると、次の行が SAS ログに出力されます。

```
Message 1 is: A character operand was found in the %EV
Message 2 is: AL function or %IF condition where a nu
Message 3 is: meric operand is required. A character
Message 4 is: operand was found in the %EVAL function
Message 5 is: or %IF condition where a numeric operan
Message 6 is: d is required.
```

例 3: %KSUBSTR と%QKSUBSTR の処理の比較

%KSUBSTR は、C 言語で特殊文字とニーモニック演算子をマスクせずに結果を処理するため、名前が解決された結果を生成します。

```
%let a=one;
%let b=two;
%let c=%nrstr(&a &b);
%put C: &c;
%put With KSUBSTR: %ksubstr(&c,1,2);
%put With QKSUBSTR: %qksubstr(&c,1,2);
```

これらのステートメントが実行されると、次の行が SAS ログに出力されます。

```
C: &a &b
With KSUBSTR: one
With QKSUBSTR: &a
```

%KUPCASE と%QKUPCASE マクロ関数

値を大文字に変換します。

カテゴリ: DBCS

種類: NLS マクロ関数

構文

`%KUPCASE` (*character string* | *text expression*)

`%QKUPCASE` (*character string* | *text expression*)

詳細

`%KUPCASE` と `%QKUPCASE` 関数は、引数内の小文字を大文字に変換します。
`%KUPCASE` は、特殊文字とニーモニック演算子をマスクせずに値を返します。

引数が次の特殊文字またはニーモニック演算子を含む場合は、`%QKUPCASE` を使用します。`%QKUPCASE` は、次の特殊文字とニーモニック演算子をマスクして値を返します。

```
& % ' " ( ) + - * / < > = ~ ^ ~ ; , # blank
AND OR NOT EQ NE LE LT GE GT IN
```

値を比較する場合、マクロ機能では比較前に小文字から大文字に自動的に変換しないため、`%KUPCASE` と `%QKUPCASE` が役に立ちます。

例

例 1: 比較する値を大文字に変更する

この例では、マクロ `RUNREPT` は、マクロ変数 `MONTH` の入力値と文字列 `DEC` を比較します。大文字に変換された値が `DEC` の場合、`PROC FSVIEW` はデータセット `REPORTS.ENDYEAR` で実行されます。DEC ではない場合、`PROC FSVIEW` は、`REPORTS` データライブラリの月名のデータセットで実行されます。

```
%macro runrept(month);
%if %kupcase(&month)=DEC %then
%str(proc fsview data=reports.endyear; run;);
%else %str(proc fsview data=reports.&month; run;);
%mend runrept;
```

%IF 条件を満たすには、次のいずれかの方法でマクロを起動します。

```
%runrept(DEC)
%runrept(Dec)
%runrept(dec)
```

例 2: %KUPCASE と%QKUPCASE の比較

次のステートメントは、`%KUPCASE` と `%QKUPCASE` により生成される結果を示します。

```
%let a=begin;
%let b=%nrstr(&a);
```



```
%put KUPCASE produces: %kupcase(&b);  
%put QKUPCASE produces: %qkupcase(&b);
```

ステートメントが実行されると、SAS ログに次の行が出力されます。

```
KUPCASE produces: begin  
QKUPCASE produces: &A
```


8 部

各国語サポート関連のシステムオプション

14 章	
システムオプションエントリ.....	457

14 章

システムオプションエントリ

カテゴリ別のシステムオプションエントリ	457
ディクショナリ	459
BOMFILE システムオプション	459
DATESTYLE=システムオプション	460
DBCS システムオプション: UNIX、Windows および z/OS	460
DBCSLANG システムオプション: UNIX、Windows および z/OS	461
DBCSTYPE システムオプション: UNIX、Windows および z/OS	462
DFLANG= システムオプション: UNIX、Windows および z/OS	464
ENCODING システムオプション: UNIX、Windows、および z/OS	465
FSDBTYPE システムオプション: UNIX	467
FSIMM システムオプション: UNIX	468
FSIMMOPT システムオプション: UNIX	469
LOCALE システムオプション	470
LOCALELANGCHG システムオプション	471
NLSCOMPATMODE システムオプション: z/OS	473
PAPERSIZE=システムオプション	474
RSASIOTRANSERROR システムオプション	474
SORTSEQ=システムオプション: UNIX、Windows および z/OS	474
TRANTAB=システムオプション	476
URLENCODING=システムオプション	478
VALIDMEMNAME システムオプション	478
VALIDVARNAME=システムオプション	478

カテゴリ別のシステムオプションエントリ

SAS システムオプションの言語コントロールカテゴリは、各国語サポートの影響を受けます。次の表に、SAS システムオプションの概要を示します。詳細については、各 SAS システムオプションの辞書エントリを参照してください。

カテゴリ	言語要素	説明
SAS ファイル	VALIDMEMNAME システムオプション (p. 478)	SAS データセット、ビュー、およびアイテムストアの命名規則を指定します。
	VALIDVARNAME=システムオプション (p. 478)	SAS セッション中に作成および処理できる有効な SAS 変数名のルールを指定します。

カテゴリ	言語要素	説明
環境コントロール: 言語コントロール	DATESTYLE=システムオプション (p. 460)	ANYDTDTM、ANYDTDTE または ANYDTTME 入力形式で、年、月、日の区別があいまいな場合に、月、日、年の順序を識別します。
	DBCS システムオプション: UNIX、Windows および z/OS (p. 460)	2 バイト文字セット(DBCS)を認識します。
	DBCSLANG システムオプション: UNIX、Windows および z/OS (p. 461)	2 バイト文字セット(DBCS)言語を指定します。
	DBCSTYPE システムオプション: UNIX、Windows および z/OS (p. 462)	2 バイト文字セット(DBCS)言語に使用するエンコーディング方法を指定します。
	DFLANG= システムオプション: UNIX、Windows および z/OS (p. 464)	国際化対応日付入力形式および出力形式の言語を指定します。
	ENCODING システムオプション: UNIX、Windows、および z/OS (p. 465)	SAS セッションのデフォルト文字セットエンコーディングを指定します。
	FSDBTYPE システムオプション: UNIX (p. 467)	フルスクリーンの 2 バイト文字セット(DBCS)エンコーディング方法を指定します。
	FSIMM システムオプション: UNIX (p. 468)	フルスクリーンの 2 バイト文字セット(DBCS)の入力方法モジュール(IMM)を指定します。
	FSIMMOPT システムオプション: UNIX (p. 469)	フルスクリーンの 2 バイト文字セット(DBCS)で使用される入力方法モジュール(IMM)のオプションを指定します。
	LOCALE システムオプション (p. 470)	地域の言語、地域の規則および文化を反映する、SAS セッションの属性一式を指定します。
	LOCALELANGCHG システムオプション (p. 471)	ODS 出力のテキストの言語が変更可能かどうかを判断します。
	NLSCOMPATMODE システムオプション: z/OS (p. 473)	各国語に以前の SAS リリースとの互換性を提供します。
	PAPERSIZE=システムオプション (p. 474)	使用するプリンタの用紙サイズを指定します。
	TRANTAB=システムオプション (p. 476)	SAS のさまざまな構成要素で使用される変換テーブルを指定します。
	URLENCODING=システムオプション (p. 478)	URLENCODE 関数と URLDECODE 関数の引数が SAS セッションのエンコーディングまたは UTF-8 エンコーディングを使って解釈されるかを指定します。

カテゴリ	言語要素	説明
並べ替え: プロシ ジャオプション	SORTSEQ=システムオプシ ョン: UNIX、Windows および z/ OS (p. 474)	SORT および SQL プロシジャが現在の SAS セッションで使用す る、言語固有の照合順序を指定します。
入力コントロール: データ処理	DATESTYLE=システムオプシ ョン (p. 460)	ANYDTDTM、ANYDTDTE または ANYDTTME 入力形式で、 年、月、日の区別があいまいな場合に、月、日、年の順序を識別 します。
ファイル: SAS ファ イル	RSASIOTRANSERROR シス テムオプション (p. 474)	リモートアプリケーションから不正なデータが読み込まれたとき にトランスコーディングエラーを表示します。
ファイル: 外部ファ イル	BOMFILE システムオプション (p. 459)	Unicode でエンコードされた外部ファイルにバイトオーダーマーク (BOM)接頭辞を書き出すかどうかを指定します。

ディクショナリ

BOMFILE システムオプション

Unicode でエンコードされた外部ファイルにバイトオーダーマーク(BOM)接頭辞を書き出すかどうかを指定します。

該当要素: 構成ファイル、SAS 起動時、OPTIONS ステートメント、**SAS システムオプション**ウィンドウ

カテゴリ: ファイル: 外部ファイル

**PROC OPTIONS
GROUP=** EXTFILES

構文

BOMFILE | NOBOMFILE

説明

BOMFILE

Unicode でエンコードされたファイルを外部ファイルに書き出すときに、バイトオー
ダーマーク(BOM)接頭辞を書き出すように指定します。

NOBOMFILE

Unicode でエンコードされたファイルを外部ファイルに書き出すときに、BOM 接頭
辞を書き出さないように指定します。

詳細

BOMFILE システムオプションは、Unicode でエンコードされた外部ファイルを読み込
むときには適用されません。

BOM は、Unicode データストリームの先頭の署名です。BOM のサイズはエンコーデ
ィングにより異なります。

DATESTYLE=システムオプション

ANYDTDTM、ANYDTDE または ANYDTTME 入力形式で、年、月、日の区別があいまいな場合に、月、日、年の順序を識別します。

該当要素: 構成ファイル、SAS 起動時、OPTIONS ステートメント、**SAS システムオプション**ウィンドウ

カテゴリ: 環境コントロール: 言語コントロール
入力コントロール: データ処理

PROC OPTIONS GROUP= INPUTCONTROL, LANGUAGECONTROL

参照項目: DATESTYLE= システムオプション(SAS システムオプション: リファレンス)

DBCS システムオプション: UNIX、Windows および z/OS

2 バイト文字セット(DBCS)を認識します。

該当要素: 構成ファイル、SAS 起動時

カテゴリ: 環境コントロール: 言語コントロール

PROC OPTIONS GROUP= LANGUAGECONTROL

デフォルト: NODBCS

UNIX 固有: SASV9_OPTIONS 環境変数を用いて指定することも可能

構文

-DBCS | -NODBCS (UNIX および Windows)

DBCS | NODBCS (z/OS)

必須引数

DBCS

値のエンコーディングで 2 バイト文字セット(DBCS)を認識します。DBCS エンコーディングは、東アジア言語のサポートに使用されます。

NODBCS

値のエンコーディングで DBCS を認識しません。かわりに、1 バイト文字セット(SBCS)が値のエンコーディングに使用されます。1 バイトで文字セットの各文字が表記されます。

詳細

DBCS システムオプションは、中国語、日本語、韓国語、台湾語のような東アジア各国の言語をサポートするために使用されます。

関連項目:

概念的な情報:

- “DBCS サポート” (37 ページ)
- “SAS セッションの DBCS 値” (561 ページ)
- “SAS 言語要素中のエンコーディング値” (563 ページ)

システムオプション:

- “DBCSLANG システムオプション: UNIX、Windows および z/OS” (461 ページ)
- “DBCSTYPE システムオプション: UNIX、Windows および z/OS” (462 ページ)

DBCSLANG システムオプション: UNIX、Windows および z/OS

2 バイト文字セット(DBCS)言語を指定します。

該当要素: 構成ファイル、SAS 起動時

カテゴリ: 環境コントロール: 言語コントロール

PROC OPTIONS LANGUAGECONTROL

GROUP=

デフォルト: なし

UNIX 固有: SASV9_OPTIONS 環境変数を用いて指定することも可能

構文

-DBCSLANG *language* (UNIX および Windows)

DBCSLANG = *language* (z/OS)

必須引数

language

動作環境に依存します。有効な言語値は、次の表を参照してください。

表 14.1 各動作環境でサポートされる DBCS 言語

言語	z/OS	UNIX	Windows
CHINESE (簡体字 中国語)	はい**	はい	はい
JAPANESE	はい	はい	はい
KOREAN	はい	はい	はい
TAIWANESE (繁体字中国語)	はい	はい	はい
NONE	はい	いいえ	はい
UNKNOWN	はい	いいえ	いいえ

* z/OS の場合のみ、CHINESE の有効なエイリアスは HANGUL および HANZI です。

詳細

DBCSSLANG システムオプションの適切な設定は、DBCSTYPE システムオプションに使用された設定に応じて異なります。一部の DBCSTYPE 設定では、DBCSSLANG 言語のすべてがサポートされますが、それ以外の DBCSTYPE 設定では日本語のみがサポートされます。

CHINESE では、簡体字中国語とも呼ばれる、中華人民共和国で 사용되는言語が指定されます。TAIWANESE では、繁体字中国語とも呼ばれる、台湾で 사용되는中国語が指定されます。

関連項目:

- “DBCS サポート” (37 ページ)
- “SAS セッションの DBCS 値” (561 ページ)
- “SAS 言語要素中のエンコーディング値” (563 ページ)

システムオプション:

- “DBCS システムオプション: UNIX、Windows および z/OS” (460 ページ)
- “DBCSTYPE システムオプション: UNIX、Windows および z/OS” (462 ページ)

DBCSTYPE システムオプション: UNIX、Windows および z/OS

2 バイト文字セット(DBCS)言語に使用するエンコーディング方法を指定します。

該当要素: 構成ファイル、SAS 起動時

カテゴリ: 環境コントロール: 言語コントロール

PROC OPTIONS GROUP= LANGUAGECONTROL

z/OS 固有: IBM

UNIX 固有: 各マシンに依存します。
SASV9_OPTIONS 環境変数を用いて指定することも可能

Windows 固有: PCMS

構文

-DBCSTYPE *encoding-method* (UNIX および Windows)

DBCSTYPE = *encoding-method* (z/OS)

必須引数

encoding-method

2 バイト文字セット(DBCS)言語のエンコードに使用する方法を指定します。
encoding-method の有効値は、コンピュータのハードウェアメーカーが動作環境に適用した標準に依存します。

詳細

DBCS エンコーディング方法は、コンピュータのハードウェアメーカーと標準組織によって異なります。

DBCSLANG=システムオプションは、エンコーディング方法が適用される言語を指定します。DBCSTYPE=は、DBCS および DBCSLANG=システムオプションも指定した場合にのみ指定する必要があります。

z/OS の DBCSTYPE=では、DBCSTYPE=値として IBM がサポートされます。

比較

表 14.2 z/OS の DBCS エンコーディング方法

DBCSTYPE=値	説明
IBM	IBM PC エンコーディング方法

表 14.3 UNIX の DBCS エンコーディング方法

DBCSTYPE=値	説明
DEC	DEC エンコーディング方法
EUC	拡張 UNIX コードエンコーディング方法
HP15	Hewlett Packard エンコーディング方法
PCIBM	IBM PC エンコーディング方法
PCMS	Microsoft PC エンコーディング方法
SJIS	日本語にのみ有効な Shift-JIS エンコーディング方法
NONE	DBCS 処理の無効化

表 14.4 Windows の DBCS エンコーディング方法

DBCSTYPE=値	説明
PCMS	Microsoft PC エンコーディング方法
windows	PCMS のエイリアス
SJIS	日本語にのみ有効な Shift-JIS エンコーディング方法

関連項目:

概念的な情報:

- [“DBCS サポート” \(37 ページ\)](#)

- “SAS セッションの DBCS 値” (561 ページ)
- “SAS 言語要素中のエンコーディング値” (563 ページ)

システムオプション:

- “DBCS システムオプション: UNIX、Windows および z/OS” (460 ページ)
- “DBCSLANG システムオプション: UNIX、Windows および z/OS” (461 ページ)

DFLANG= システムオプション: UNIX、Windows および z/OS

国際化対応日付入力形式および出力形式の言語を指定します。

該当要素:	構成ファイル、SAS 起動時、OPTIONS ステートメント、 SAS システムオプション ウィンドウ
カテゴリ:	環境コントロール: 言語コントロール
PROC OPTIONS GROUP=	LANGUAGECONTROL
デフォルト:	English

構文

DFLANG=*language*, locale

説明

language

国際化対応日付入力形式および出力形式に使用する言語を指定します。

language の値として有効なのは、次の言語です。

- Afrikaans
- Catalan
- Croatian
- Czech
- Danish
- Dutch
- English
- Finnish
- French
- German
- Hungarian
- Italian
- Japanese
- Macedonian
- Norwegian
- Polish

- Portuguese
- Russian
- Slovenian
- Spanish
- Swedish
- Swiss_French
- Swiss_German

locale

LOCALE システムオプションで指定されたロケールがアクティブなロケールになります。

詳細

DFLANG システムオプションの値は、SAS セッション中に変更できますが、一度に使用できる言語は 1 つのみです。language の値では、大文字と小文字は区別されません。

`dflang=locale` を指定すると、ロケールステートメントシステムオプションで指定されたロケールがアクティブなロケールになります。locale/language は DFLANG システムオプションでサポートされていない限りなりません。

次の例では、国際化対応日付入力形式および出力形式はドイツ語(German)になります。ドイツ語ロケールの POSIX 名は `de_DE` になります。ドイツ語ロケールは DFLANG システムオプションでサポートされています。

```
option locale=de_DE; /* German locale */
option DFLANG=locale;
```

次の例では、国際化対応日付入力形式および出力形式は英語(English)になります。マルタ語(Maltese)は `dflang` でサポートされていないので、デフォルトロケールは English になります。

```
option locale=mt_MT; /* Maltese locale */
option DFLANG=locale;
```

`dflang=locale` を指定すると、日付はロケールステートメントのシステムオプションで指定されたロケールを使用して表示されます。出力の日付形式をコントロールするために、DFLANG のロケールとして起動時に設定された LOCALE システムオプションに基づく値を使用します。DFLANG が有効な言語に設定されていない場合、出力の日付形式はデフォルトでは英語になります。次の例では、ロケールはフランス語に設定され、リスト出力が指定されています。

```
Sas.exe -locale French
Proc print data=sashelp.class ; run ;
```

```
mercredi 09 mars 2011 14 h 25
```

ENCODING システムオプション: UNIX、Windows、および z/OS

SAS セッションのデフォルト文字セットエンコーディングを指定します。

該当要素: 構成ファイル、SAS 起動時

カテゴリ: 環境コントロール: 言語コントロール

PROC OPTIONS	LANGUAGECONTROL
GROUP=	
OpenVMS 固有:	latin1
z/OS 固有:	OPEN_ED-1047
Windows 固有:	wlatin1

構文

-ENCODING= ASCIIANY | EBCDICANY | *encoding-value* (UNIX と Windows)

ENCODING= *encoding-value* (UNIX、Windows および z/OS)

必須引数

ASCIIANY

トランスコーディングは、通常は SAS によりセッションエンコーディングとデータセットエンコーディングが異なることが検出されたときに発生します。ASCIIANY では、データセットにアクセスする SAS セッションに ASCII の値をエンコードするセッションがある場合に、SAS でトランスコードされないデータセットを作成できます。EBCDIC エンコーディングを使用するマシンにデータセットを移動すると、トランスコーディングが発生します。

注: ANY は BINARY の指定と同等です。データはバイナリであるため、実際のエンコーディングには意味がありません。

EBCDICANY

z/OS でのみ有効です。トランスコーディングは、通常は SAS によりセッションエンコーディングとデータセットエンコーディングが異なることが検出されたときに発生します。EBCDICANY では、データセットにアクセスする SAS セッションに EBCDIC の値をエンコードするセッションがある場合に、SAS でトランスコードされないデータセットを作成できます。ASCII エンコーディングを使用するマシンにデータセットを移動すると、トランスコーディングが発生します。

encoding-value

すべての動作環境の有効な値については、“[SAS セッションのエンコーディング値](#)” (573 ページ)を参照してください。

詳細

文字セットエンコーディングは、コードポイントと呼ばれる数値にマップされた文字のセットです。

SAS セッションのエンコーディングは、次のように ENCODING=、LOCALE=、DBCSTYPE=、DBCSLANG=システムオプションの値で決まります。

- ENCODING=と LOCALE=システムオプションが指定されていない場合、デフォルト値は ENCODING=です。UNIX の場合、デフォルト値は latin1 です。Windows の場合、デフォルト値は wlatin1 です。z/OS の場合、デフォルトは OPEN_ED-1047 です。
- ENCODING オプションが指定されていない場合、エンコーディング値は、LOCALE の値と SAS を起動しているオペレーティングシステムから判別されます。また、LOCALE が設定されていない場合、デフォルト LOCALE は en_US になります。
- LOCALE=と ENCODING=の両方が指定された場合、セッションエンコーディングは ENCODING=オプションで指定された値です。

- LOCALE=が指定されていて ENCODING=が指定されていない場合、SAS により LOCALE=値から適切なエンコーディング値が推測されます。
- DBCS オプションが設定された場合、DBCSSLANG=と DBCSTYPE=システムオプションの値により ENCODING=と LOCALE=の値が決まります。

関連項目:

概念的な情報:

- “各国語サポート関連のロケール概念の概要” (5 ページ)
- “概要: 各国語サポート関連のエンコーディング” (9 ページ)
- “トランスコーディングの概要” (27 ページ)
- 表 18.1 (549 ページ)
- “DBCS データ処理に関連する SAS システムオプション” (561 ページ)
- “SAS 言語要素中のエンコーディング値” (563 ページ)

FSDBTYPE システムオプション: UNIX

フルスクリーンの 2 バイト文字セット(DBCS)エンコーディング方法を指定します。

該当要素: 構成ファイル、SAS 起動時、SASV9_OPTIONS 環境変数

カテゴリ: 環境コントロール: 言語コントロール

PROC OPTIONS GROUP= LANGUAGECONTROL

デフォルト: DEFAULT

UNIX 固有: すべて

構文

-FSDBTYPE *encoding-method*

詳細

FSDBTYPE=システムオプションでは、フルスクリーン DBCS を可能にする方法に適切なエンコーディング方法を指定します。フルスクリーン DBCS エンコーディング方法は、コンピュータのハードウェアメーカーと標準組織により異なります。

表 14.5 フルスクリーン DBCS エンコーディング方法

FSDBTYPE=エンコーディング方法	説明
dec	Digital Equipment Corporation (DEC)エンコーディング方法
euc	拡張 UNIX エンコーディング方法
hp15	HP-UX エンコーディング方法

FSDBTYPE=エンコーディング方法	説明
jis7	日本語の X Windows 環境でのみ使用される 7 ビット Shift-JIS エンコーディング方法
pcibm	IBM PC エンコーディング方法
SJIS	日本語にのみ有効な Shift-JIS エンコーディング方法
デフォルト	特定のホストで使用されるデフォルトの方法

関連項目:

概念的な情報:

- “[DBCS サポート](#)” (37 ページ)
- “[SAS セッションの DBCS 値](#)” (561 ページ)
- “[SAS 言語要素中のエンコーディング値](#)” (563 ページ)

FSIMM システムオプション: UNIX

フルスクリーンの 2 バイト文字セット(DBCS)の入力方法モジュール(IMM)を指定します。

該当要素: 構成ファイル、SAS 起動時、SASV9_OPTIONS 環境変数

カテゴリ: 環境コントロール: 言語コントロール

PROC OPTIONS GROUP= LANGUAGECONTROL

デフォルト: なし

UNIX 固有: すべて

構文

```
-FSIMM fsdevice_name=IMM-name1<,fsdevice_name=IMM-name2> ...
```

詳細

IMM-name に次の値を指定できます。

TTY | SASWUJT

`/dev/tty` のインターフェイスを提供します。この IMM では、DBCS 入力機能のある端末エミュレータ経由で DBCS 文字列を入力できます。

PIPE | SASWUJP

パイプインターフェイスを提供します。このインターフェイスでは、DBCS 入力サーバープロセスを分岐します。デフォルトのサーバー名は `saswujms` で、ベンダで提供される MOTIF ツールキットを使用します。

たとえば、X11 ドライバに PIPE 入力方法モジュールを使用するには、次のように指定します。

-FSIMM X11=PIPE

注: サーバーは FSIMMOPT オプションを使用して指定します。

関連項目:

概念的な情報:

- [“DBCS サポート” \(37 ページ\)](#)

システムオプション:

- [“FSIMMOPT システムオプション: UNIX ” \(469 ページ\)](#)

FSIMMOPT システムオプション: UNIX

フルスクリーンの 2 バイト文字セット(DBCS)で使用される入力方法モジュール(IMM)のオプションを指定します。

該当要素: 構成ファイル、SAS 起動時、SASV9_OPTIONS 環境変数

カテゴリ: 環境コントロール: 言語コントロール

PROC OPTIONS GROUP= LANGUAGECONTROL

デフォルト: なし

UNIX 固有: すべて

構文

-FSIMMOPT *fullscreen-IMM:IMM-option*

詳細

FSIMMOPT システムオプションでは、各フルスクリーン IMM (入力方法モジュール) のオプションを指定します。各 IMM に 1 つの FSIMMOPT オプションのみ指定できます。同じ IMM に複数の FSIMMOPT オプションを指定した場合、最後の指定のみが使用されます。

各 IMM のオプション値については、SAS Technical Report J-121DBCS *Support Usage Guide* (日本語)を参照してください。

たとえば、PIPE IMM で使用するサーバー名 MOTIF の指定に、FSIMMOPT オプションを使用できます。

-fsimmopt PIPE:MOTIF

関連項目:

概念的な情報:

- [“DBCS サポート” \(37 ページ\)](#)

システムオプション:

- [“FSIMM システムオプション: UNIX ” \(468 ページ\)](#)

LOCALE システムオプション

地域の言語、地域の規則および文化を反映する、SAS セッションの属性一式を指定します。

該当要素: 構成ファイル、SAS 起動時、OPTIONS ステートメント、**SAS システムオプション**ウィンドウ

カテゴリ: 環境コントロール: 言語コントロール

**PROC OPTIONS
GROUP=** LANGUAGECONTROL

デフォルト: English_UnitedStates

UNIX 固有: SASV9_OPTIONS 環境変数を用いて指定することも可能

構文

-LOCALE *locale-name* (UNIX および Windows)

LOCALE=*locale-name* (UNIX、Windows および z/OS)

必須引数

locale-name

全ロケール値(SAS 名および POSIX 名)のリストについては、“[ENCODING、PAPERSIZE、DFLANG、DATESTYLE オプションの LOCALE=値とデフォルト設定](#)” (549 ページ)を参照してください。

詳細

LOCALE=システムオプションは、地域の規則、言語および文化を反映するロケールの指定に使用します。

LOCALE=システムオプションの値に ENCODING=システムオプションの値との互換性がない場合、文字セットエンコーディングは、ENCODING=システムオプションの値によって決まります。

DBCS=システムオプションがアクティブな場合、DBCSTYPE=および DBCSLANG=システムオプションの値によってロケールと文字セットエンコーディングが決まります。

LOCALE=の値を設定すると、次のシステムオプションの値は、明示的に値を指定しない限り、変更されます。

ENCODING=

設定したロケールには、SAS が実行する動作環境の多くで使用される一般的なエンコーディング値があります。ENCODING= オプションが構成ファイルやコマンドラインに明示的に指定されていない場合、SAS は LOCALE とオペレーションシステムのデフォルトのエンコーディングを使用します。LOCALE を明示的に指定できませんが、しない場合はデフォルトになります。ENCODING=システムオプションが設定されると、TRANTAB=システムオプションも設定されます。

DATESTYLE=

LOCALE=が設定されると、DATESTYLE=システムオプションには選択されたロケールに対応する値が使用されます。

DFLANG=

LOCALE=が設定されると、DFLANG=システムオプションは選択されたロケールに対応する値に設定されます。

PAPERSIZE=

LOCALE=が設定されると、PAPERSIZE=システムオプションは選択されたロケールに対応する値に設定され、ODS プリンタがそのロケールの優先測定単位(インチまたはセンチメートル)に設定されます。

注意:

Windows オペレーティングシステムのみ: LOCALE=オプションを使用して PAPERSIZE=を指定できるのは、UNIVERSALPRINT および UPRINTMENUSWITCH システムオプションも指定された場合のみです。詳細については UNIVERSALPRINT [kIsError - link not found - The element n1qvbs96exn1d8n19c9yjf28bu4p was not found in the link database](#) システムオプション(*SAS システムオプション: リファレンス*)と UPRINTMENUSWITCH [kIsError - link not found - The element n06plvzgzolpsxqn1rkmoknl0c4nc was not found in the link database](#) システムオプション(*Windows 版 SAS*)を参照してください。

関連項目:

概念的な情報:

- “[各国語サポート関連のロケール](#)” (5 ページ)
- “[ENCODING、PAPERSIZE、DFLANG、DATESTYLE オプションの LOCALE=値とデフォルト設定](#)” (549 ページ)

システムオプション:

- “[ENCODING システムオプション: UNIX、Windows、および z/OS](#)” (465 ページ)
- “DATESTYLE= System Option” in *SAS System Options: Reference*
- “[DFLANG= システムオプション: UNIX、Windows および z/OS](#)” (464 ページ)
- “PAPERSIZE= System Option” in *SAS System Options: Reference*
- “[TRANTAB=システムオプション](#)” (476 ページ)

LOCALELANGCHG システムオプション

ODS 出力のテキストの言語が変更可能かどうかを判断します。

該当要素: 構成ファイル、SAS 起動時

カテゴリ: 環境コントロール: 言語コントロール

PROC OPTIONS GROUP= LANGUAGECONTROL

デフォルト: LOCALELANGCHG は、Unicode サーバー以外のすべてのサーバーでオフに設定されています。

ヒント: LOCALELANGCHG オプションを使用する言語の切り替え機能は、Unicode サーバー (UTF-8、ENCODING=utf8 のセッションエンコーディングを使用する SAS サーバー)でサポートされます。

構文

LOCALELANGCHG | NOLOCALELANGCHG

説明

LOCALELANGCHG

起動後に LOCALE オプションが設定された場合に、ODS 出力の SAS メッセージテキストの言語を変更できるように指定します。

NOLOCALELANGCHG

起動後に LOCALE オプションが設定された場合に、ODS 出力の SAS メッセージテキストの言語が変更できないように指定します。

詳細

言語の切り替え機能により、起動後に SAS メッセージの言語を変更できます。この機能を使用するには、LOCALELANGCHG を有効にする必要があります。

起動中に、構成ファイルと LOCALE オプションにより SAS メッセージの言語が決まります。起動後に LOCALE オプションと LOCALELANGCHG オプションが設定されると、地域化が可能な場合はメッセージと ODS テンプレートの言語が LOCALE 設定にあわせて変更されます。

LOCALELANGCHG を有効にした場合でも、出力でそのロケールの言語には翻訳されません。たとえば、LOCALELANGCHG を有効にし、フランス語の SAS セッションを開始してロケールをギリシャ語に設定した場合、NLDATE はギリシャ語で表示されます。出力はフランス語で表示されます。出力がフランス語で表示されるのは、ギリシャ語に翻訳されていないためです。

比較

LOCALELANGCHG を起動時に有効にして LOCALE をセッション中に変更すると、新規ロケールの言語に翻訳されたテンプレートアイテムストアが存在する場合はそれを含めるように ODS PATH が更新されます。SAS ログに表示されないメッセージは、新規ロケールの言語で表示されます。ログメッセージは、セッションロケールの元の言語で表示されます。

LOCALELANGCHG を起動時に有効にせずに LOCALE をセッション中に変更すると、ODS 出力は起動時に設定された言語で表示されます。

例

例 1 は、LOCALELANGCHG を有効にしていない(NOLOCALELANGCHG)フランス語のサーバーです。

フランス語のクライアントアプリケーションがサーバーに接続すると、出力はフランス語で表示され、NLDATE 出力形式が適用された日付もフランス語で表示されます。ドイツ語のクライアントアプリケーションがフランス語のサーバーに接続し、サーバーでロケールがドイツ語に変更された場合、出力メッセージはフランス語で表示され、NLDATE を使った日付形式はドイツ語で表示されます。

例 2 は、LOCALELANGCHG を有効にした(LOCALELANGCHG)フランス語のサーバーです。

フランス語のクライアントアプリケーションがサーバーに接続すると、出力はフランス語で表示され、NLDATE 出力形式が適用された日付もフランス語で表示されます。ドイツ語のクライアントアプリケーションがフランス語のサーバーに接続し、サーバーでロケールがドイツ語に変更された場合、出力メッセージはドイツ語で表示され、NLDATE を使った日付形式もドイツ語で表示されます。

NLSCOMPATMODE システムオプション: z/OS

各国語に以前の SAS リリースとの互換性を提供します。

該当要素:	構成ファイル、SAS 起動時
カテゴリ:	環境コントロール: 言語コントロール
PROC OPTIONS GROUP=	LANGUAGECONTROL
デフォルト:	NONLSCOMPATMODE

構文

NLSCOMPATMODE | NONLSCOMPATMODE

説明

NLSCOMPATMODE

デフォルト言語である英語以外の言語のデータを処理するために、以前の SAS リリースとの互換性を提供します。NLSCOMPATMODE が設定されると、以前の SAS リリースで実行されていたプログラムが引き続き動作します。

注: NLSCOMPATMODE は、ODS を使用して作成された出力の形式に影響を与える場合があります。ODS を使用している場合は、オプション値を NONLSCOMPATMODE に設定します。

NONLSCOMPATMODE

英語以外の言語でネイティブの文字を使用してデータを処理するためのサポートを提供します。NONLSCOMPATMODE を設定すると、文字データは SAS セッションに指定されたエンコーディングを使用して処理されます。

NONLSCOMPATMODE が有効な場合、SAS 構文で代替文字はサポートされません。NONLSCOMPATMODE を設定して SAS を実行する場合、代替文字ではなく各国語文字を使用するように既存のプログラムを更新する必要があります。たとえば、既存の SAS プログラムで 'À' を '\$' 文字で代替するデンマーク語の顧客は、'\$' を顧客の環境で使用するように SAS 構文を更新する必要があります。

詳細

NONLSCOMPATMODE システムオプションは、英語以外のエンコーディングを使用し、新規アプリケーションのコーディングに際して新しい業界標準の利用を考える顧客向けに提供されています。

NLSCOMPATMODE または NONLSCOMPATMODE 設定によって、LOCALE または ENCODING システムオプションの値が変更されることはありません。

NONLSCOMPATMODE が有効な場合、SAS は ENCODING または LOCALE オプションで設定されたエンコーディングを文字データの処理に使用します。コンパイラとセッションエンコーディングの文字は、引き続き別個に扱われます。

注: NLSCOMPATMODE オプションは廃止される予定のため、NLSCOMPATMODE が設定された場合、次の警告が SAS ログに出力されます: SAS has been started in NLS compatibility mode with the NLSCOMPATMODE option. このオプションは今後の SAS リリースでは廃止され、NLS 互換性モードはサポートされなくなります。詳細については、SAS Institute のテクニカルサポートにお問い合わせください。

PAPERSIZE=システムオプション

使用するプリンタの用紙サイズを指定します。

該当要素: 構成ファイル、SAS 起動時、OPTIONS ステートメント、**SAS システムオプション**ウィンドウ

カテゴリ: 環境コントロール: 言語コントロール

PROC OPTIONS GROUP= LANGUAGECONTROL

参照項目: PAPERSIZE=システムオプション in SAS システムオプション: リファレンス

RSASIOTRANSERROR システムオプション

リモートアプリケーションから不正なデータが読み込まれたときにトランスコーディングエラーを表示します。

該当要素: 構成ファイル、SAS 起動時、OPTIONS ステートメント、**SAS システムオプション**

カテゴリ: ファイル: SAS ファイル

PROC OPTIONS GROUP= SASFILES

デフォルト: RSASIOTRANSERROR

構文

RSASIOTRANSERROR | NORSASIOTRANSERROR

説明

RSASIOTRANSERROR

リモートアプリケーションから不正な値が読み込まれた場合に、トランスコーディングエラーを表示するように指定します。

NORSASIOTRANSERROR

リモートアプリケーションから不正な値が読み込まれた場合に、トランスコーディングエラーを表示しないように指定します。

詳細

RSASIOTRANSERROR システムオプションにより、SAS Enterprise Guide や SAS Enterprise Miner などの SASIO のリモートユーザーは、不正なデータ値を無視できません。通常は不正なデータ値がリモートアプリケーションから読み込まれると、トランスコーディングエラーが発生します。

SORTSEQ=システムオプション: UNIX、Windows および z/OS

SORT および SQL プロシジャが現在の SAS セッションで使用する、言語固有の照合順序を指定します。

該当要素: 構成ファイル、SAS 起動時、OPTIONS ステートメント、**SAS システムオプション**ウィンドウ

カテゴリ: 並べ替え: プロシジャオプション

PROC OPTIONS
GROUP=

SORT

構文SORTSEQ=*collating-sequence***説明***collating-sequence*

SORT プロシジャが現在の SAS セッションで使用する照合順序を指定します。有効な値をユーザーが指定するか、次のいずれかを選択できます。

- ASCII
- DANISH (エイリアスは NORWEGIAN)
- EBCDIC
- FINNISH
- ITALIAN
- NATIONAL
- POLISH
- REVERSE
- SPANISH
- SWEDISH

詳細

照合順序を作成または変更するには、TRANTAB プロシジャを使用して変換テーブルを作成または変更します。独自の変換テーブルを作成すると、作成したテーブルは PROFILE カタログに保存されます。これは HOST カタログに同じ名前で作成された変換テーブルより優先されます。

注: システムマネージャは、新しく作成したテーブルを PROFILE カタログから HOST カタログにコピーすることで、HOST カタログを変更できます。新しい変換テーブルまたは変更された変換テーブルには、すべてのユーザーがアクセスできます。

ウィンドウ環境の場合は、**エクスプローラ**ウィンドウを使用して SASHELP HOST カタログを表示します。HOST カタログでは、タイプ TRANTAB のエントリに、エントリ名で識別される照合順序が含まれています。

ウィンドウ環境でない場合は、次のステートメントを発行して、HOST カタログのコンテンツのリストを生成します。タイプ TRANTAB のエントリが照合順序です。

```
proc catalog catalog=sashelp.host;
contents;
run;
```

特定の変換テーブルのコンテンツを表示するには、次のステートメントを使用します。

```
proc trantab table=translation-table-name;
list;
run;
```

照合順序のコンテンツは、SAS ログに表示されます。

例

この例は、SORTSEQ と一緒に PROC SORT および PROC SQL を使用した場合の機能を示しています。

```
options sortseq=reverse;
proc sort data=sashelp.class out=fool;
by name;
run;
proc sql;
create table foo2 as select * from sashelp.class order by name;
quit;
run;
```

関連項目:

- “照合順序” (16 ページ)

システムオプション:

- “TRANTAB=システムオプション” (476 ページ)

TRANTAB=システムオプション

SAS のさまざまな構成要素で使用される変換テーブルを指定します。

該当要素: 構成ファイル、SAS 起動時、OPTIONS ステートメント、**SAS システムオプション**ウィンドウ

カテゴリ: 環境コントロール: 言語コントロール

**PROC OPTIONS
GROUP=** LANGUAGECONTROL

操作: TRANTAB=システムオプションでは、ファイル転送を含む、SAS セッションで使用される変換テーブルを指定します。TRANTAB ステートメントでは、エクスポートまたは転送される SAS ファイルの文字セットに適用する、カスタマイズした変換テーブル(たとえば、EBCDIC 文字の ASCII 文字へのマッピング)を指定します。

構文

TRANTAB=(*catalog-entries*)

説明

catalog-entries

変換テーブルを含む SAS カタログエントリを指定します。*entry-name.type* を指定すると、最初に SASUSER.PROFILE が検索され、次に SASUSER.HOST が検索されます。

詳細

TRANTAB=は各国語の要件をサポートするために SAS 6 で導入されました。SAS 8.2 では、TRANTAB=の機能が改善された LOCALE=システムオプションが導入されました。SAS 9.2 では、TRANTAB=が下位互換性のためにサポートされています。ただし、SAS 8.2 以降のリリースでは、LOCALE=システムオプションを使用することをお勧めします。

変換テーブルは、10 個の位置を含むかっこで囲まれたリストで指定します。リスト内でのテーブルの位置によって指定する変換テーブルの種類が決まります。リストの各エントリはカンマで区切られます。次の位置と種類のリストを参照してください。

位置	変換テーブルの種類
1 番目	local-to-transport-format
2 番目	transport-to-local-format
3 番目	lowercase-to-uppercase
4 番目	uppercase-to-lowercase
5 番目	文字の分類
6 番目	スキャナ変換
7 番目	デルタ文字
8 番目	スキャナ文字の分類
9 番目	未使用
10 番目	DBCS ユーザーテーブル

注意:

利用方法がよくわからない場合は変換テーブルを変更しないでください。変換テーブルは SAS のシステムにより各国語サポートを提供するために内部的に利用されます。変換テーブルの利用方法がよくわからない場合や、技術的な支援がない場合は設定を変更しないでください。

1 つのテーブルを変更するには、その他のテーブルに NULL エントリを指定します。たとえば、リストの 3 番目にある lowercase-to-uppercase テーブルを変更するには、uppercase を次のように指定します。

```
options trantab = ( , , new-uppercase-table);
```

その他のテーブルは変更されません。OPTIONS プロシジャからの出力では、TRANTAB=オプションの複数の指定ではなく最後の指定のみが反映されます。次に、例を示します。

```
options trantab = ( , , new-uppercase-table);
options trantab = ( , , , new-lowercase-table);
```

PROC OPTIONS では、TRANTAB=の値は

(, , , new-lowercase-table)と表示されますが、new-uppercase と new-lowercase の両方のテーブルに効力があります。

関連項目:

[17 章, “TRANTAB プロシジャ,” \(521 ページ\)](#)

URLENCODING=システムオプション

URLENCODE 関数と URLDECODE 関数の引数が SAS セッションのエンコーディングまたは UTF-8 エンコーディングを使って解釈されるかを指定します。

該当要素: 構成ファイル、SAS 起動時、OPTIONS ステートメント、

カテゴリ: 環境コントロール: 言語コントロール

**PROC OPTIONS
GROUP=** LANGUAGECONTROL

参照項目: “URLENCODING= System Option” in *SAS System Options: Reference*

構文

URLENCODING=[SESSION](#) | [UTF8](#)

VALIDMEMNAME システムオプション

SAS データセット、ビュー、およびアイテムストアの命名規則を指定します。

該当要素: 構成ファイル、SAS 起動時、OPTIONS ステートメント、**SAS システムオプション**ウィンドウ

カテゴリ: SAS ファイル

**PROC OPTIONS
GROUP=** SASFILES

参照項目: VALIDMEMNAME システムオプション in *SAS システムオプション: リファレンス*

構文

VALIDMEMNAME=[COMPAT](#) | [EXTEND](#)

VALIDVARNAME=システムオプション

SAS セッション中に作成および処理できる有効な SAS 変数名のルールを指定します。

該当要素: 構成ファイル、SAS 起動時、OPTIONS ステートメント、SAS システムオプションウィンドウ

カテゴリ: SAS ファイル

**PROC OPTIONS
GROUP=** SASFILES

デフォルト: V7

参照項目: “VALIDVARNAME= System Option” in *SAS System Options: Reference*

構文

VALIDVARNAME=[V7](#) | [UPCASE](#) | [ANY](#)

9 部

各国語サポート関連のコマンド、ステートメント、プロシジャオプション

15 章	
コマンド、ステートメント、プロシジャオプションのエントリ.....	481

15 章

コマンド、ステートメント、プロシジャ
オプションのエントリ

カテゴリ別の各国語サポート関連のコマンド、ステートメント、およびプロシジャオプション	481
ディクショナリ	482
CHARSET=オプション	482
照合順序オプション	483
CORRECTENCODING=オプション	489
CVPBYTES=オプション、CVPENGINE=オプション、 CVPMULTIPLIER=オプション	490
ENCODING=オプション	495
INENCODING=/OUTENCODING=オプション	499
ODSCHARSET=オプション	500
ODSTRANTAB=オプション	501
PROC SQL の TRANSCODE=列修飾子	502
RENCODING=オプション	503
TRANSCODE=オプション	505
TRANTAB=オプション	507
XMLENCODING=オプション	508
TRANTAB ステートメント	509

カテゴリ別の各国語サポート関連のコマンド、ステートメント、およびプロシジャオプション

選択した SAS ステートメントのオプションに対するデータアクセスカテゴリとデータセットコントロールカテゴリは、各国語サポートの影響を受けます。次の表に、ステートメントオプションの概要を示します。詳細については、各ステートメントオプションの辞書エントリを参照してください。

カテゴリ	言語要素	説明
ODS: サードパーティの出力形式	CHARSET=オプション (p. 482)	出力の META 宣言で生成される文字セットを指定します。
	TRANTAB=オプション (p. 507)	出力ファイルが適切になるよう、SAS ファイルの文字データをトランスコードする場合に使用する変換テーブルを指定します。
情報	TRANSCODE=オプション (p. 505)	文字変数をトランスコードするかどうかを指示する、ATTRIB ステートメントの属性(1 つ以上の変数に出力形式、入力形式、ラベル、長さに関連付ける)を指定します。

カテゴリ	言語要素	説明
データアクセス	CVPBYTES=オプション、 CVPENGINE=オプション、 CVPMULTIPLIER=オプション (p. 490)	SAS ファイルをトランスコードするために必要とされる文字変数の属性を指定します。
	ENCODING=オプション (p. 495)	外部ファイルの入力または出力処理のエンコーディングを優先指定してトランスコードします。
	INENCODING=/ OUTENCODING=オプション (p. 499)	SAS ライブラリの SAS データセットの読み込みまたは書き出し時のエンコーディングを無効にして変更します。
	ODSCHARSET=オプション (p. 500)	出力の META 宣言で生成される文字セットを指定します。
	ODSTRANTAB=オプション (p. 501)	出力ファイルの XML ドキュメントをトランスコードするときに使用する変換テーブルを指定します。
	RENCODING=オプション (p. 503)	EBCDICANY または ASCIIANY セッションエンコーディングを使用する SAS/SHARE サーバーセッションのデータをトランスコードする場合に使用する、ASCII ベースまたは EBCDIC ベースのエンコーディングを指定します。
	XMLENCODING=オプション (p. 508)	外部ドキュメントのインポートまたはエクスポート時に、XML ドキュメントのエンコーディングより優先される値を指定します。

ディクショナリ

CHARSET=オプション

出力の META 宣言で生成される文字セットを指定します。

該当要素: ODS MARKUP ステートメントと ODS HTML ステートメントの LIBNAME ステートメント

カテゴリ: ODS: サードパーティの出力形式

構文

CHARSET=*character-set* ;

必須引数

character-set

HTML 出力の META タグで使用する文字セットを指定します。

エンコーディングの例は ISO-8859-1 です。インターネットで使用する公式の文字セットは、IANA (Internet Assigned Numbers Authority) によって登録されます。IANA は、ポート、プロトコルとエンタープライズ番号、オプション、コードとタイプなどのさまざまなインターネットプロトコルパラメータの中央レジストリです。

character-set 値の全リストについては、www.unicode.org/reports/tr22/index.html および www.iana.org/assignments/character-sets を参照してください。

character set は、このコンテキストの *encoding-value* に似ています。ただし、*character set* という用語は、インターネットでの使用に適したエンコーディングを識別するために使用されます。

例: ODS MARKUP ステートメントの META 宣言で生成される出力

```
<META http-equiv="Content-Type" content="text/html; charset=iso-8858-1">
```

関連項目:

概念的な情報:

- “各国語サポート関連のエンコーディング” (9 ページ)

ステートメント:

- `xisError - link not found - The element n0onpp2holuauhn1gyb7lzx6knr7 was not found in the link database`
- `xisError - link not found - The element n0f5s1zezthhbrn1u0z71mh3wx64 was not found in the link database`

照合順序オプション

PROC SORT の照合順序を指定します。

該当要素: PROC SORT ステートメント

注: PROC SORT ステートメントは、SAS データセットのオブザベーションを 1 つ以上の文字変数または数値変数を基準に並べ替えます。

構文

```
PROC SORT collating-sequence-option <other option(s)> ;
```

オプション

オプションには、1 つの *collating-sequence-option* と複数の *other options* を含めることができます。2 種類のオプションの順序は重要でなく、両方の種類を同じ PROC SORT ステップに含める必要はありません。PROC SORT *collating-sequence-option* についてのみ次に説明します。

動作環境の情報

DANISH、FINNISH、NORWEGIAN、SWEDISH のいずれかの *collating-sequence-option* に対する動作環境固有の動作の詳細については、現在の動作環境向けの SAS ドキュメントを参照してください。

ASCII

ASCII 照合順序を使用して文字変数を並べ替えます。このオプションは、EBCDIC がネイティブ照合順序のシステムで ASCII 並べ替えを実行する場合にのみ必要です。

DANISH NORWEGIAN

デンマーク語とノルウェー語に従って文字を並べ替えます。

デンマーク語とノルウェー語の照合順序は、[図 15.1 \(485 ページ\)](#)に示されています。

EBCDIC

EBCDIC 照合順序を使用して文字変数を並べ替えます。このオプションは、ASCII がネイティブ照合順序のシステムで EBCDIC 並べ替えを実行する場合にのみ必要です。

POLISH

ポーランド語の規則に従って文字を並べ替えます。

FINNISH SWEDISH

フィンランド語とスウェーデン語の規則に従って文字を並べ替えます。フィンランド語とスウェーデン語の照合順序は、[図 15.1 \(485 ページ\)](#)に示されています。

NATIONAL

各国の使用法の違いを反映するために、インストール時の定義に従い、代替照合順序を使用して文字変数を並べ替えます。このオプションを使用するには、カスタマイズされた各国の並べ替え順序がサイトに定義されている必要があります。カスタマイズされた各国の並べ替え順序が使用可能かどうかを確認するには、サイトの SAS インストール担当者にお問い合わせください。

NORWEGIAN

DANISH を参照してください。

SWEDISH

FINNISH を参照してください。

SORTSEQ=collating-sequence

照合順序を指定します。*collating-sequence* には、*collating-sequence-option*、変換テーブル、エンコーディング、キーワード LINGUISTIC のいずれかを指定できます。1 つの照合順序のみ指定できます。詳細については、“[照合順序](#)” (16 ページ)を参照してください。

照合順序の説明を次に示します。

collating—sequence—option | translation table

SAS で提供される変換テーブルかユーザー定義の変換テーブル、または 1 つの PROC SORT ステートメントの *collating-sequence-option* を指定します。SORTSEQ=とともに PROC TRANTAB と PROC SORT を使用した例については、“[例 6: 並べ替えへのさまざまな変換テーブルの使用](#)” (540 ページ)を参照してください。

使用可能な変換テーブルは、次のとおりです。

- ASCII
- DANISH
- EBCDIC
- FINNISH
- ITALIAN
- NORWEGIAN
- POLISH
- REVERSE
- SPANISH

- SWEDISH

次の図に、各言語で英数字が並べ替えられる方法を示します。

図 15.1 英数字の各言語での並べ替え

Danish:	0123456789ABCDEF GHIJKLMNOPQRSTUVWXYZÆØÅabc defghijklmnopqrstuvwxyzæøå
Finnish:	0123456789ABCDEF GHIJKLMNOPQRSTUVWXYZÄÅÖabc defghijklmnopqrstuvwxyzääö
Italian:	0123456789AÀBÇÇDEÉÈÈFGHIÌJ KLMNOÒPQRSTUÙVWXYZaàbcççdeéèèfghii jklmnoòpqrstuùvwxyz
Norwegian:	0123456789ABCDEF GHIJKLMNOPQRSTUVWXYZÆØÅabc defghijklmnopqrstuvwxyzæøå
Spanish:	0123456789AÁaáBbCcDdEéEeFfGgHhÍíiíJjKkLlMmNnÑñOóoóPpQqRrSsTtUúuúÚúVvWwXxYyZz
Swedish:	0123456789ABCDEF GHIJKLMNOPQRSTUVWXYZÄÅÖabc defghijklmnopqrstuvwxyzääö

制限事項: PROC SORT ステップでは、1 つの collating-sequence-option のみ指定できます。

ヒント: SORTSEQ=照合順序オプションは、かっこを使用せず、引数を関連付けずに指定します。照合順序の指定方法の例を次に示します。`proc sort data=mydata SORTSEQ=ASCII;`

encoding-value

エンコーディング値を指定します。この結果は、指定したエンコーディングで表示される文字データのバイナリ照合と同じです。サポートされるエンコーディング値は“データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値” (563 ページ)を参照してください。

制限事項: SORTSEQ=オプションで指定したエンコーディングを認識する SAS System の一部またはプロシジヤは、PROC SORT のみです。

ヒント: エンコーディング値に英数字またはアンダースコア以外の文字が含まれる場合、その値を引用符で囲む必要があります。

参照項目: “データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値” (563 ページ)で指定可能なエンコーディングのリスト。

LINGUISTIC<(collating—rules)>

指定した言語のルールに従って文字を並べ替える、言語照合を指定します。ルールおよびデフォルトの照合順序オプションは、現在のロケール設定で指定した言語に基づきます。International Components for Unicode (ICU)ライブラリにより実装され、Unicode 照合アルゴリズム(UCA)と大部分で互換性のある結果を作成します。

別名: UCA

制限事項: SORTSEQ=LINGUISTIC オプションは PROC SORT SORTSEQ=オプションでのみ使用可能で、SAS System の SORTSEQ=オプションでは使用できません。

ヒント:

LINGUISTIC 並べ替えは、z/OS メインフレームではより多くのメモリを必要とします。REGION の 50M 以上の設定が必要な場合があります。バッチモードで実行する場合は JCL、対話形式で実行する場合は VERIFY 画面でこのアクションを実行する必要があります。このアクションにより ICU ライブラリが適切に読み込まれ、並べ替えに使用するメモリに影響しません。collating-rules はかっこで囲む必要があります。複数の照合ルールを指定できます。

言語照合で並べ替えたデータセットで BY 処理を実行する場合、データセットが適切に処理されるように NOBYSORTED システムオプションの指定が必要な場合があります。BY 処理は、照合順序処理とは異なる方法で実行されます。

参照項目:

Appendix 4, “ICU License - ICU 1.8.1 and later,” in *Base SAS Procedures Guide*

“照合順序” (16 ページ)を参照してください。

Unicode 照合アルゴリズム(UCA)の仕様については、<http://www.unicode.org> の Web サイトを参照してください。

LINGUISTIC オプションで指定可能な collation-rules を次に示します。これらのルールにより言語照合順序が変更されます。

ALTERNATE HANDLING=SHIFTED

スペース、句読点、記号などの変数文字の処理を制御します。このオプションを指定しない(デフォルト値の Non-Ignorable を使用する)場合、これらの変数文字間の違いは、通常の文字間の違いと同様に重要です。

ALTERNATE_HANDLING オプションを指定する場合、これらの変数文字はあまり重要ではありません。

デフォルト: NON_IGNOREABLE

ヒント: SHIFTED 値は、QUATERNARY に設定された STRENGTH=と組み合わせてよく使用されます。このような場合、スペース文字、句読点、記号が文字列の比較時に考慮されますが、文字列のその他の要素(基本文字、アクセント記号、大文字と小文字)が同一の場合に限定されます。

CASE FIRST=

大文字と小文字の順序を指定します。この引数は、TERTIARY、QUATERNARY または IDENTICAL レベルでのみ有効です。次の表に、CASE_FIRST 引数の値と情報を示します。

値	説明
UPPER	大文字、小文字の順序で並べ替えます。
LOWER	小文字、大文字の順序で並べ替えます。

COLLATION=

次の表に、使用可能な COLLATION=値を示します。照合値を選択しない場合、ユーザーのロケールのデフォルト照合が選択されます。

値	説明
BIG5HAN	ラテン語にピンインの並べ替えを指定し、中国語、日本語、韓国語の文字に big5 文字セットの並べ替えを指定します。
DIRECT	ヒンディー語のバリエントを指定します。
GB2312HAN	ラテン語にピンインの並べ替えを指定し、中国語、日本語、韓国語の文字に gb2312han 文字セットの並べ替えを指定します。
PHONEBOOK	文字の並べ替えに電話帳スタイルを指定します。PHONEBOOK はドイツ語でのみ選択します。

値	説明
PINYIN	1文字ずつピンインに変換する方法を使用した中国語、日本語、韓国語の文字の並べ替えを指定します。この並べ替えは、簡体字中国語でよく使用されます。
POSIX	ポータブルオペレーティングシステムインターフェイスです。このオプションでは、文字の"C"ロケール並べ替えを指定します。
STROKE	英字以外の書体の並べ替えを指定しません。中国語、日本語、韓国語、ベトナム語でSTROKEを選択します。この並べ替えは、繁体字中国語でよく使用されます。
TRADITIONAL	文字の並べ替えに従来のスタイルを指定します。たとえば、スペイン語でTRADITIONALを選択します。

LOCALE=locale_name

POSIX 名形式でロケール名を指定します。たとえば、ja_JP と指定します。PROC SORT でサポートされるロケールと POSIX 値のリストについては、“[ENCODING、PAPERSIZE、DFLANG、DATESTYLE オプションの LOCALE=値とデフォルト設定](#)” (549 ページ)を参照してください。

制限事項: 次のロケールは PROC SORT でサポートされていません。

- Afrikaans_SouthAfrica, af_ZA
- Cornish_UnitedKingdom, kw_GB
- ManxGaelic_UnitedKingdom, gv_GB

NUMERIC_COLLATION=

数を示す文字のかわりに、数値でテキスト内の整数値を並べ替えます。

値	説明
ON	数値で番号を並べ替えます。たとえば、"8 Main St."は"45 Main St."より前になります。
OFF	文字の値で数を並べ替えます。たとえば、"45 Main St."は"8 Main St."より前になります。

デフォルト: OFF

STRENGTH=

強度の値は、照合レベルに関連付けられます。5つの照合レベル値があります。次の表に、5つのレベルの情報を示します。強度のデフォルト値は、ロケールに関連付けられます。

値	照合の種類	説明
PRIMARY または 1	PRIMARY では、基本文字間の差異を指定します("a"<"b"など)。	この差異は最強です。たとえば、辞書は基本文字により異なるセクションに分割されません。

値	照合の種類	説明
SECONDARY または 2	文字のアクセント記号が第 2 差異となります("as"<"às"<"at"など)。	文字列内に第 1 差異がある場合は、第 2 差異は無視されません。文字間のその他の差異も、言語に応じて第 2 差異と考慮される場合があります。
TERTIARY または 3	大文字と小文字の差異は、第 3 レベルで区別されます("ao"<"Ao"<"aò"など)。	文字列内に第 1 または第 2 差異がある場合は、第 3 差異は無視されます。別の例として、通常の仮名と小文字の仮名の違いがあります。
QUATERNARY または 4	レベル 1 から 3 で句読点が無視される場合、句読点の有無で単語を区別するために追加レベルが使用されます("ab"<"a-b"<"aB"など)。	句読点の無視が必須の場合や日本語テキストを処理する場合は、第 4 レベルを使用する必要があります。第 1、第 2 または第 3 差異がある場合は、この差異は無視されます。
IDENTICAL または 5	他のすべてのレベルが等しい場合、最終的な決定をするために同一レベルが使用されます。各文字列の正規化形式 D (NFD) の Unicode コードポイント値がこのレベルで比較され、レベル 1 から 4 で違いがないことが確認されません。	2 つの文字列間でコードポイント値のみが異なることはほとんどないため、このレベルは必要最小限に使用します。たとえば、ヘブライ語の詠唱マークのみがこのレベルで区別されます。

別名: Level=

注意:

データの並べ替えにホストの並べ替えユーティリティを使用する場合、SORTSEQ=オプションで変換テーブルベースの照合順序を指定すると、BY 変数の文字が破損する可能性があります。詳細については、現在の作環境向けのドキュメントの PROC SORT を参照してください。

詳細

PROC SORT ステートメントの照合順序オプションは、SAS データセットのオブザベーションを 1 つ以上の文字または数値変数で並べ替えます。

表 15.1 オプション

タスク	オプション
照合順序の指定	
ASCII の指定	ASCII (483 ページ)
EBCDIC の指定	EBCDIC (484 ページ)
デンマーク語の指定	DANISH (484 ページ)
フィンランド語の指定	FINNISH (484 ページ)

タスク	オプション
ノルウェー語の指定	NORWEGIAN (484 ページ)
ポーランド語の指定	POLISH (484 ページ)
スウェーデン語の指定	SWEDISH (484 ページ)
カスタマイズされた順序の指定	NATIONAL (484 ページ)
前述の照合順序(ASCII、EBCDIC、DANISH、FINNISH、ITALIAN、NORWEGIAN、POLISH、SPANISH、SWEDISH、NATIONAL)、システムで提供されたその他の変換テーブル名(POLISH、SPANISH)、ユーザーが作成した変換テーブル名のいずれかを指定します。エンコーディングを指定できます。また、ロケールに適切な照合順序にするために、キーワード LINGUISTIC または UCA を指定できます。	SORTSEQ= (484 ページ)

関連項目:

- “照合順序” (16 ページ)
- Chapter 56, “SORT Procedure” in *Base SAS Procedures Guide*

システムオプション:

- “SORTSEQ=システムオプション: UNIX、Windows および z/OS” (474 ページ)
- “TRANTAB=システムオプション” (476 ページ)

CORRECTENCODING=オプション

SAS ファイルのエンコーディング属性を、SAS ファイル内のデータのエンコーディングに一致させるように明示的に変更します。

該当要素: DATASETS プロシジャの MODIFY ステートメント

構文

```
MODIFY SAS file </<CORRECTENCODING=encoding-value>> ;
```

オプション引数

```
</ <CORRECTENCODING=encoding-value> >
```

ファイル内のデータの実際のエンコーディングに一致させるため、ファイルのディスクリプタ情報に記録されるエンコーディング指標の変更を可能にします。各 SAS ファイル名の後のこのオプションには、かっこは使用できません。フォワードスラッシュの後に CORRECTENCODING=を指定する必要があります。次に、例を示します。

```
modify mydata / correctencoding=latin2;
```

トランスコーディングの有効なエンコーディング値のリストについては、“データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値” (563 ページ)を参照してください。

制限事項: CORRECTENCODING=は、SAS ファイルがデフォルトのベースエンジンである SAS 9 の V9 を使用する場合にのみ、使用できます。

例: SAS セッションエンコーディングと SAS ファイルエンコーディングを解決するための CORRECTENCODING=オプションの使用

ファイルのエンコーディング指標は、データのエンコーディングとは別にできます。たとえば、SAS 9 より前のバージョンで作成された SAS ファイルには、エンコーディング指標は保存されていません。このようなエンコーディングが記録されていない SAS ファイルを SAS 9 のセッションで開くと、現在のセッションのエンコーディングが割り当てられます。たとえば、データのエンコーディングがデンマーク語 EBCDIC で現在のセッションのエンコーディングが Western Wlatin1 の場合、ファイル内のデータの実際のエンコーディングとファイルのディスクリプタ情報に保存されたエンコーディング指標は一致しません。この場合、データは正常にトランスコードされず、出力が読み込めなくなる可能性があります。次の MODIFY ステートメントを使用し、EBCDIC エンコーディングを明示的に割り当てることでこの問題を解決できます。

注: CEDA では読み取り専用のコピーが作成されます。データを永続的にトランスコードするには、PROC COPY または DATA ステップでデータをコピーする必要があります。

```
proc datasets library=myfiles;
modify olddata / correctencoding=ebcdic1142;
quit;
```

CVPBYTES=オプション、CVPENGINE=オプション、CVPMULTIPLIER=オプション

SAS ファイルをトランスコードするために必要とされる文字変数の属性を指定します。

該当要素: LIBNAME ステートメント

カテゴリ: データアクセス

PROC OPTIONS GROUP= 現在の動作環境(Windows、UNIX、Z/OS) 向けのドキュメント内の LIBNAME ステートメント

参照項目: LIBNAME、SAS/ACCESS

構文

```
LIBNAME libref <CVPBYTES=bytes> <CVPENGINE=engine> <CVPMULTIPLIER=multiplier>
> 'SAS data-library';
```

オプション引数

CVPBYTES=bytes

トランスコーディングを必要とする SAS データファイルを処理するときに、拡張する文字変数の長さをバイト数で指定します。CVP エンジンは、長さを拡張することにより、文字データの切り捨てが発生しないようにします。文字変数の長さは、指定

した値を現在の長さに追加して増やします。0 から 32766 までの値を指定できません。

たとえば、次の LIBNAME ステートメントでは、CVPBYTES=オプションを指定して CVP エンジンを暗黙的に割り当てています。

```
libname expand 'SAS data-library' cvpbytes=5;
```

文字変数の長さは、5 バイト追加して増やします。長さが 10 の文字変数は長さ 15 に、長さが 100 の文字変数は長さ 105 に増加します。

デフォルト: CVPBYTES=を指定すると、指定値に従って文字変数の長さを拡張するために CVP エンジンを自動的に使用します。CVP エンジンを明示的に割り当てても、CVPBYTES=または CVPMULTIPLIER=を指定しない場合は、CVPMULTIPLIER=1.5 を使用して文字変数を長くします。

制限事項:

CVP エンジンは、SAS データファイルのみをサポートしています。つまり、SAS のビュー、カタログおよびアイテムストアなどはサポートしていません。

CVP エンジンは、入力(読み込み)処理でのみ使用できます。

CVP エンジンなどのエンジンが混在するライブラリ連結では、SAS データファイルのみが処理されます。たとえば、COPY プロシジャを実行する場合は、SAS データファイルのみコピーされます。

要件 指定するバイト数は、どのような拡張にも適応できる大きさにする必要があります。この大きさが不十分だと、データが切り捨てられ、その結果 SAS ログにエラーメッセージが書き込まれます。

操作: CVPBYTES=と CVPMULTIPLIER=の両方を指定できません。いずれか 1 つのオプションを指定してください。

参照項目: “CVP エンジンを用いた文字データ切り捨てへの対応” (34 ページ)

CVPENGINE=engine

トランスコーディングを必要とする SAS データファイルを処理するために使用するエンジンを指定します。トランスコードする文字変数の長さが CVP エンジンによって拡張されるため、文字データの切り捨ては発生しません。次に、指定されたエンジンで実際のファイル処理を実行します。

別名: CVPENG

デフォルト: SAS は、デフォルトの SAS エンジンを使用します。

参照項目: “CVP エンジンを用いた文字データ切り捨てへの対応” (34 ページ)

CVPMULTIPLIER=multiplier

トランスコーディングを必要とする SAS データファイルを処理するときに、文字変数の長さを拡張するための乗数値を指定します。CVP エンジンは、長さを拡張することにより、文字データの切り捨てが発生しないようにします。文字変数の長さは、指定した値を現在の長さに乗算して増やします。1 から 5 までの乗数値を指定できます。

たとえば、次の LIBNAME ステートメントでは、CVPMULTIPLIER=オプションを指定して CVP エンジンを暗黙的に割り当てています。

```
libname expand 'SAS data-library' cvpmultiplier=2.5;
```

文字変数の長さは 2.5 倍になります。長さが 10 の文字変数は長さ 25 に、長さが 100 の文字変数は長さ 250 に増加します。

別名: CVPMULT

デフォルト: CVPMULTIPLIER=を指定すると、指定値に従って文字変数の長さを拡張するために CVP エンジンを自動的に使用します。CVP エンジンを明示的に割り当てても、CVPMULTIPLIER=または CVPBYTES=を指定しない場合は、CVPMULTIPLIER=1.5 を使用して文字変数を長くします。

制限事項:

CVP エンジンは、SAS データファイルのみをサポートしています。つまり、SAS のビュー、カタログおよびアイテムストアなどはサポートしていません。

CVP エンジンは、入力(読み込み)処理でのみ使用できます。

CVP エンジンなどのエンジンが混在するライブラリ連結では、SAS データファイルのみが処理されます。たとえば、COPY プロシジャを実行する場合は、SAS データファイルのみコピーされます。

要件 指定するバイト数は、どのような拡張にも適応できる大きさにする必要があります。この大きさが不十分だと、データが切り捨てられ、その結果 SAS ログにエラーが書き込まれます。

操作: CVPMULTIPLIER=と CVPBYTES=の両方を指定できません。いずれか 1 つのオプションを指定してください。

参照項目: “CVP エンジンを用いた文字データ切り捨てへの対応” (34 ページ)

例: CVP (文字変数埋め込み)エンジンの使用

次の例に、CVP エンジンを使用して文字データの切り捨てを回避する方法を示します。例では、Wlatin2 エンコーディングの一部の各国語文字を含む MYFILES.WLATIN2 という名前の SAS データセットを使用します。

```
libname myfiles 'C:\Documents and Settings\sasdxw\My Documents\myfiles';
data myfiles.wlatin2 (encoding=wlatin2);
var1='41'x;
var2='8a'x;
var3='9c'x;
var4='b3'x;
;
proc print data=myfiles.wlatin2;
run;
```

The SAS System

Obs	var1	var2	var3	var4
1	A	Š	œ	’

これは、MYFILES.WLATIN2 での PROC CONTENTS 出力です。エンコーディングが Wlatin2 で、各文字変数の長さが 1 バイトであることを示しています。

アウトプット 15.1 MYFILES.WLATIN2 の PROC CONTENTS 出力

```

The SAS System 1
The CONTENTS Procedure
Data Set Name MYFILES.WLATIN2 Observations 1
Member Type DATA Variables 4
Engine V9 Indexes 0
Created Thursday, November 07, 2003 02:02:36 Observation Length 4
Last Modified Thursday, November 07, 2003 02:02:36 Deleted Observations 0
Protection Compressed NO
Data Set Type Sorted NO
Label
Data Representation WINDOWS_32
Encoding wlatin2 Central Europe (Windows)
Engine/Host Dependent Information
Data Set Page Size 4096
Number of Data Set Pages 1
First Data Page 1
Max Obs per Page 987
Obs in First Data Page 1
Number of Data Set Repairs 0
File Name C:\Documents and Settings\xxxxxx\My
Documents\myfiles\wlatin2.sas7bdat
Release Created 9.0100A0
Host Created XP_PRO
Alphabetic List of Variables and Attributes
# Variable Type Len
1 Var1 Char 1
2 Var2 Char 1
3 Var3 Char 1
4 Var4 Char 1

```

次のコードがセッションエンコーディング Wlatin2 を使用して実行されます。

```

options msglevel=i;
libname myfiles 'SAS data-library';
data myfiles.utf8 (encoding="utf-8");
set myfiles.wlatin2;
run;

```

この DATA ステップでは、新しいデータセット MYFILES.UTF8 を作成し、データを UTF-8 エンコーディングのこの新しいデータセットに読み込む(つまり、Wlatin2 から UTF-8 へのデータのトランスコードが必要)ことを要求します。要求は、トランスコーディングによって起こる文字データの切り捨てが原因でエラー終了します。新規データセット MYFILES.UTF8 が作成されますが、データは何も含まれていません。

ログ15.1 トランスコーディングエラーの SAS ログ

```

1 options msglevel=i;
2 libname myfiles 'C:\Documents and Settings\xxxxxx\My Documents\myfiles';
NOTE: ライブラリ参照名 MYFILES を次のように割り当てました。
エンジン: V9
外部ファイル名: C:\Documents and Settings\xxxxxx\My Documents\myfiles
3 data myfiles.utf8 (encoding="utf-8");
4 set myfiles.wlatin2;
5 run;
INFO: データファイル MYFILES.UTF8.DATA に他のホストにネイティブな形式が使用されているか、またはエンコーディングがセッションエンコーディングと一致していません。
クロス環境データアクセスが使用され、パフォーマンスの低下と追加の CPU リソースが必要となる場合もあります。
ERROR: データセット MYFILES.UTF8 のトランスコード時に文字データが一部損失しました。
NOTE: DATA ステップは異常終了しました。
NOTE: エラーが発生したため、このステップの処理を中止しました。
NOTE: データセット MYFILES.WLATIN2 から 1 オブザベーションを読み込みました。
WARNING: データセット MYFILES.UTF8 は未完成です。このステップは、0 オブザベーション、4 変数で停止しました。

```

次のコードがセッションエンコーディング Wlatin2 を使用して再度実行されます。

```

options msglevel=i;
libname myfiles 'SAS data-library';
libname expand cvp 'SAS data-library' cvpbytes=2;
data myfiles.utf8 (encoding="utf-8");
set expand.wlatin2;
run;

```

この例では、CVP エンジンを使用し、それぞれの長さに 2 バイトを追加して文字変数の長さを拡張します。データは、Wlatin2 から UTF-8 にトランスコードされることにより、UTF-8 エンコーディングで新規ファイルに読み込まれます。拡張された文字変数の長さが原因のデータの切り捨ては発生せず、新規データセットが正常に作成されず。

ログ15.2 MYFILES.UTF8 の SAS ログ出力

```

12 options msglevel=i;
13 libname myfiles 'C:\Documents and Settings\xxxxxx\My Documents\myfiles';
NOTE: エンジンタイプの異なるファイルがライブラリ MYFILES のディレクトリに存在します。
NOTE: ライブラリ参照名 MYFILES を次のように割り当てました。
エンジン: V9
外部ファイル名: C:\Documents and Settings\xxxxxx\My Documents\myfiles
14 libname expand cvp 'C:\Documents and Settings\xxxxxx\My Documents\myfiles'
cvpbytes=2;
WARNING: ライブラリ参照名 EXPAND は MYFILES と同じ物理ライブラリを参照しています。
NOTE: ライブラリ参照名 EXPAND を次のように割り当てました。
エンジン: CVP
外部ファイル名: C:\Documents and Settings\xxxxxx\My Documents\myfiles
15 data myfiles.utf8 (encoding="utf-8");
16 set expand.wlatin2;
17 run;
INFO: データファイル MYFILES.UTF8.DATA に他のホストにネイティブな形式が使用されているか、またはエンコーディングがセッションエンコーディングと一致していません。
クロス環境データアクセスが使用され、パフォーマンスの低下と追加の CPU リソースが必要となる場合もあります。
NOTE: データセット EXPAND.WLATIN2 から 1 オブザベーションを読み込みました。
NOTE: データセット MYFILES.UTF8 には 1 オブザベーションと 4 変数があります。

```

最後に示すのは MYFILES.UTF8 の PROC CONTENTS 出力です。これは、UTF-8 エンコーディングで表示され、各文字変数の長さは 3 です。

アウトプット 15.2 MYFILES.UTF8 の PROC CONTENTS 出力

```

The SAS System 1
The CONTENTS Procedure
Data Set Name MYFILES.UTF8 Observations 1
Member Type DATA Variables 4
Engine V9 Indexes 0
Created Thursday, November 07, 2003 02:40:34 Observation Length 12
Last Modified Thursday, November 07, 2003 02:40:34 Deleted Observations 0
Protection Compressed NO
Data Set Type Sorted NO
Label
Data Representation WINDOWS_32
Encoding utf-8 Unicode (UTF-8)
Engine/Host Dependent Information
Data Set Page Size 4096
Number of Data Set Pages 1
First Data Page 1
Max Obs per Page 335
Obs in First Data Page 1
Number of Data Set Repairs 0
File Name C:\Documents and Settings\xxxxxx\My Documents\myfiles\utf8.sas7bdat
Release Created 9.0100A0
Host Created XP_PRO
Alphabetic List of Variables and Attributes
# Variable Type Len
1 Var1 Char 3
2 Var2 Char 3
3 Var3 Char 3
4 Var4 Char 3

```

ENCODING=オプション

外部ファイルの入力または出力処理のエンコーディングを優先指定してトランスコードします。

該当要素: %INCLUDE ステートメント、FILE ステートメント、FILENAME ステートメント、FILENAME ステートメントの EMAIL (SMTP)アクセス方式、INFILE ステートメント、ODS ステートメント、FILE コマンド、INCLUDE コマンド

カテゴリ: データアクセス

構文

ENCODING= '*encoding-value*'

オプション引数

ENCODING= '*encoding-value*'

外部ファイルの読み込み、書き出し、コピー、保存に使用するエンコーディングを指定します。ENCODING=の値は、外部ファイルのエンコーディングが現在のセッションエンコーディングとは異なることを示します。

外部ファイルを使用してデータの読み込み、書き出し、コピー、保存を行うと、SAS はデータをセッションエンコーディングから指定されたエンコーディングにトランスコードします。

詳細については、「データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値」(563 ページ)を参照してください。

デフォルト: SAS は、現在のセッションエンコーディングを使用します。

詳細

次のテーブルでは、ENCODING オプションが対応するステートメントでどのように使われるかについて説明します。

%INCLUDE ステートメント:	SAS ステートメントおよびデータ行の指定したソースファイルからの読み込み (z/OS 動作環境ではサポートしていません)
FILE ステートメント:	外部ファイルへの書き出し
FILENAME ステートメント:	外部ファイルからの読み込み、外部ファイルへの書き出し
FILENAME ステートメントの EMAIL (SMTP) アクセス方式:	プログラムによる電子メール送信
INFILE ステートメント:	外部ファイルからの読み込み
ODS ステートメント:	SAS プロシジャおよび DATA ステップの出力の生成、保存または再作成に使用する Output Delivery System (ODS)機能の制御
FILE コマンド:	ウィンドウのコンテンツの外部ファイルへの保存
INCLUDE コマンド:	外部ファイルの現在のウィンドウへのコピー

エンコーディングのいくつかはバイトオーダーマーク(BOM)を使用します。BOM はエンコーディングの指定時に生成されます。UTF-8 エンコーディングでは、BOM を生成するには、filename とファイルの DATA ステップステートメントに `encoding=utf-8` を指定しなければなりません。

例

例 1: FILE ステートメントを使用した外部ファイルへの書き出しのエンコーディングの指定

この例では、SAS データセットから外部ファイルを作成します。現在のセッションエンコーディングは Wlatin1 ですが、外部ファイルのエンコーディングは UTF-8 にする必要があります。デフォルトで、SAS は外部ファイルを現在のセッションエンコーディングを使用して書き出します。

外部ファイルへのデータの書き出しに使用するエンコーディングを指定するには、ENCODING=オプションを次のように指定します。

```
libname myfiles 'SAS data-library';
filename outfile 'external-file';
data _null_;
set myfiles.cars;
file outfile encoding="utf-8";
put Make Model Year;
run;
```

SAS に対して外部ファイルを UTF-8 エンコーディングにするように指示すると、SAS はデータを Wlatin1 から指定された UTF-8 エンコーディングにトランスコードします。

例 2: FILENAME ステートメントを使用した外部ファイルの読み込みのエンコーディングの指定

この例では、外部ファイルから SAS データセットを作成します。外部ファイルは UTF-8 文字セットエンコーディングで、現在の SAS セッションは Wlatin1 エンコーディングです。デフォルトでは、SAS は外部ファイルがセッションエンコーディングと同じエンコーディングであると想定します。この場合、文字データは新しい SAS データセットに正しく書き出されません。

外部ファイルの読み込み時に使用するエンコーディングを指定するには、ENCODING=オプションを次のように指定します。

```
libname myfiles 'SAS data-library';

filename extfile 'external-file' encoding="utf-8";
data myfiles.unicode;
infile extfile;
input Make $ Model $ Year;
run;
```

外部ファイルが UTF-8 エンコーディングであると指定すると、SAS は、新しい SAS データセットへの書き出し時に外部ファイルを UTF-8 から現在のセッションエンコーディングにトランスコードします。そのため、データは新しいデータセットに Wlatin1 で正しく書き出されます。

例 3: FILENAME ステートメントを使用した外部ファイルへの書き出しのエンコーディングの指定

この例では、SAS データセットから外部ファイルを作成します。デフォルトで、SAS は外部ファイルを現在のセッションエンコーディングを使用して書き出します。現在のセッションエンコーディングは Wlatin1 ですが、外部ファイルのエンコーディングは UTF-8 にする必要があります。

外部ファイルへのデータ書き出し時に使用するエンコーディングを指定するには、ENCODING=オプションを次のように指定します。

```
libname myfiles 'SAS data-library';
filename outfile 'external-file' encoding="utf-8";
data _null_;
set myfiles.cars;
file outfile;
put Make Model Year;
run;
```

外部ファイルを UTF-8 エンコーディングにするように指定すると、SAS は、外部ファイルへの書き出し時にデータを Wlatin1 から指定された UTF-8 エンコーディングにトランスコードします。

例 4: メッセージ本文と添付ファイルのエンコーディングの変更

この例では、メッセージ本文と添付ファイルのテキストエンコーディングを変更する方法を説明します。

```
filename mymail email 'Joe.Developer@sas.com';
data _null_;
file mymail
subject='Text Encoding'
encoding=greek
```

```
attach=('C:\My Files\Test.out'
content_type='text/plain'
encoding='ebcdic1047'
outencoding='latin1');
run;
```

プログラムでは、次の操作が実行されます。

- ENCODING=電子メールオプションで、メッセージ本文を送信前にギリシャ語(ISO)でエンコードするように指定します。
- ATTACH=電子メールオプションの場合、添付ファイルオプション ENCODING=で、SAS に読み込まれる添付ファイルのエンコーディングを西ヨーロッパ諸語(EBCDIC)に指定します。
- SMTP やその他の電子メールインターフェイスは EBCDIC をサポートしないため、添付ファイルオプション OUTENCODING=によって、添付ファイルは送信前に西ヨーロッパ諸語(ISO)に変換されます。

例 5: INFILE=ステートメントを使用した外部ファイル読み込みのエンコーディングの指定

この例では、外部ファイルから SAS データセットを作成します。外部ファイルのエンコーディングは UTF-8 で、現在の SAS セッションエンコーディングは Wlatin1 です。デフォルトでは、SAS は外部ファイルがセッションエンコーディングと同じエンコーディングであると想定します。この場合、文字データは新しい SAS データセットに正しく書き出されません。

外部ファイルの読み込み時に使用するエンコーディングを指定するには、ENCODING=オプションを次のように指定します。

```
libname myfiles 'SAS data-library';
filename extfile 'external-file';
data myfiles.unicode;
infile extfile encoding="utf-8";
input Make $ Model $ Year;
run;
```

外部ファイルが UTF-8 エンコーディングであると指定すると、SAS は、新しい SAS データセットへの書き出し時に外部ファイルを UTF-8 から現在のセッションエンコーディングにトランスコードします。そのため、データは新しいデータセットに Wlatin1 で正しく書き出されます。

関連項目:

ステートメント:

- [xisError - link not found - The element p0cnuwy17gjlfn1r6oq3ockv0or was not found in the link database](#)
- [xisError - link not found - The element n0d4ukw70ct5y1n15m2bsoqd08qv was not found in the link database](#)
- “FILE Statement” in *SAS Statements: Reference*
- “FILENAME Statement” in *SAS Statements: Reference*
- “INFILE Statement” in *SAS Statements: Reference*

コマンド:

- “FILE Command” in *SAS Companion for z/OS*

- `xisError - link not found - The element n1epme0p3j1ueen1s38tjss396wn was not found in the link database`
- `xisError - link not found - The element p1msyyi1rcjhgvn1liw5xkxktvzj was not found in the link database`
- “INCLUDE Command” in *SAS Companion for z/OS*
- `xisError - link not found - The element p0vc5p15ut7f03n1o0553f2523pj was not found in the link database`

INENCODING=/OUTENCODING=オプション

SAS ライブラリの SAS データセットの読み込みまたは書き出し時のエンコーディングを無効にして変更します。

該当要素: LIBNAME ステートメント

カテゴリ: データアクセス

構文

INENCODING= ANY | ASCIIANY | EBCDICANY | *encoding-value*

OUTENCODING= ANY | ASCIIANY | EBCDICANY | *encoding-value*

説明

ANY

ASCII エンコーディングと EBCDIC エンコーディング間でトランスコーディングを行わないように指定します。

注意: ANY は BINARY の指定と同等です。データはバイナリであるため、実際のエンコーディングには意味がありません。

ASCIIANY

混在したエンコーディングが ASCII エンコーディングと想定して、トランスコーディングを行わないように指定します。

EBCDICANY

混在したエンコーディングが EBCDIC エンコーディングと想定して、トランスコーディングを行わないように指定します。

encoding-value

エンコーディング値を指定します。エンコーディング値のリストについては、“[SAS セッションのエンコーディング値](#)” (573 ページ)を参照してください。

詳細

INENCODING=オプションは、SAS ライブラリの SAS データセットの読み込みに使用されます。OUTENCODING=オプションは、SAS ライブラリの SAS データセットの書き出しに使用されます。

INENCODING=値または OUTENCODING=値は、LIST 引数を使用する場合、SAS ログに書き出されます。

INENCODING=および OUTENCODING=は、混在したエンコーディングが含まれる既存のライブラリを使用する場合に、最も適しています。混在したエンコーディングが含まれるライブラリを読み取るには、INENCODING=を ASCIIANY または EBCDICANY に設定できます。別個のデータセットを書き出すには、

OUTENCODING=を使用して特定のエンコーディングを指定できます。このエンコーディングがデータセットの作成時に適用されます。

比較

- セッションエンコーディングは、ENCODING=システムオプションまたは LOCALE=システムオプションを使用して指定します。各動作環境にはデフォルトのエンコーディングがあります。
- SAS ライブラリのデータセットを読み取るためのエンコーディングを指定するには、LIBNAME ステートメントの INENCODING=オプションを入力ファイルに使用します。LIBNAME ステートメントオプションと ENCODING=データセットオプションを両方とも指定すると、データセットオプションが使用されます。
- SAS ライブラリにデータセットを書き出すためのエンコーディングを指定するには、LIBNAME ステートメントの OUTENCODING=オプションを出力ファイルに使用します。LIBNAME ステートメントオプションと ENCODING=データセットオプションを両方とも指定すると、データセットオプションが使用されます。
- COPY プロシジャの場合、デフォルトの CLONE オプションでは、OUTENCODING=オプションで指定したエンコーディング値ではなく、入力データセットのエンコーディング属性を使用します。CLONE と NOCLONE の詳細については、COPY ステートメントを参照してください。

注: この操作は、SAS/CONNECT または SAS/SHARE 使用時には適用されません。

関連項目:

- [“概要: 各国語サポート関連のエンコーディング” \(9 ページ\)](#)

ステートメント:

- [“LIBNAME Statement” in SAS Statements: Reference](#)

システムオプション:

- [“ENCODING システムオプション: UNIX、Windows、および z/OS” \(465 ページ\)](#)
- [“LOCALE システムオプション” \(470 ページ\)](#)

データセットオプション:

- [“ENCODING=データセットオプション” \(49 ページ\)](#)

ODSCHARSET=オプション

出力の META 宣言で生成される文字セットを指定します。

該当要素: XML エンジンの LIBNAME ステートメント

カテゴリ: データアクセス

構文

ODSCHARSET=*character-set*;

必須引数

character-set

XML エンジンの LIBNAME ステートメントでは、ENCODING=属性で使用する文字セットを指定します。

エンコーディングの例は ISO-8859-1 です。インターネットで使用する公式の文字セットは、IANA (Internet Assigned Numbers Authority) によって登録されます。IANA は、ポート、プロトコルとエンタープライズ番号、オプション、コードとタイプなどのさまざまなインターネットプロトコルパラメータの中央レジストリです。character-set 値の全リストについては、www.unicode.org/reports/tr22/index.html および www.iana.org/assignments/character-sets を参照してください。

character set は、このコンテキストの *encoding-value* に似ています。ただし、*character set* という用語は、インターネットでの使用に適したエンコーディングを識別するために使用されます。

詳細

XML ドキュメントによっては、XML 宣言が不要なものもあります。XML 宣言は、ドキュメントの文字エンコーディングがデフォルトの UTF-8 または UTF-16 以外で、上位プロトコルでエンコーディングが決まっていない場合にのみ必要となります。

XML エンジンの LIBNAME ステートメントの ODSCHARSET オプションで出力する XML ドキュメントに使う文字セットを指定します。

関連項目:

概念的な情報:

- “[各国語サポート関連のエンコーディング](#)” (9 ページ)

ステートメント:

- *SAS XML LIBNAME Engine: ユーザーガイド*

ODSTRANTAB=オプション

出力ファイルの XML ドキュメントをトランスコードするときに使用する変換テーブルを指定します。

該当要素: XML エンジンの LIBNAME ステートメント

カテゴリ: データアクセス

構文

TRANTAB ='translation-table'

オプション引数

translation-table

出力ファイルに使用する変換テーブルを指定します。変換テーブルは、文字セットの文字(通常の文字、表語文字、数字、句読点、記号、コントロール文字など)を数値にマッピングするエンコーディング方法です。変換テーブルの例として、文字を EBCDIC から ASCII-ISO に変換するものがあります。table-name には、SAS で提

供される変換テーブルまたはユーザー定義の変換テーブルを指定できます。値は、SASUSER.PROFILE カタログまたは SASHELP.HOST カタログの SAS カタログエントリ名である必要があります。

詳細

SAS 9.2 では、下位互換性を維持するため、XML エンジンで LIBNAME ステートメントの ODSSTRANTAB=オプションを使用します。エンコーディングを指定する場合、LOCALE=システムオプションを使用する方法が推奨されます。

関連項目:

概念的な情報:

- “トランスコーディングと変換テーブル” (28 ページ)
- “各国語サポート関連のロケール” (5 ページ)の概要の説明

システムオプション:

- “TRANTAB=システムオプション” (476 ページ)
- “LOCALE システムオプション” (470 ページ)

プロシジャ:

- 17 章, “TRANTAB プロシジャ,” (521 ページ)

ステートメント:

- *SAS XML LIBNAME Engine: ユーザーガイド*

PROC SQL の TRANSCODE=列修飾子

文字列に値をトランスコードできるかどうかを指定します。

該当要素: SQL プロシジャの列修飾子の構成要素

構文

TRANSCODE=YES|NO

必須引数

TRANSCODE=YES|NO

文字列について、値をトランスコードできるかどうかを指定します。トランスコーディングを実行しないようにするには、TRANSCODE=NO を使用します。CREATE TABLE AS ステートメントを使用してテーブルを作成する場合、作成されるテーブルの特殊文字の列のトランスコーディング属性は、TRANSCODE=列修飾子を使用して変更しない限り、入力元テーブルの属性と同じであることに注意してください。

デフォルト: YES

制限事項: トランスコーディングの抑止は、V6TAPE エンジンではサポートされていません。

関連項目:

- “[各国語サポート関連のトランスコーディング](#)” (27 ページ)
- *Base SAS プロシジャガイド*

RENCODING=オプション

EBCDICANY または ASCIIANY セッションエンコーディングを使用する SAS/SHARE サーバーセッションのデータをトランスコードする場合に使用する、ASCII ベースまたは EBCDIC ベースのエンコーディングを指定します。

該当要素: SAS/SHARE のみの LIBNAME ステートメント

カテゴリ: データアクセス

注: LIBNAME ステートメントの RENCODING=オプションは、エンコーディングが混在するコンピューティング環境を維持するためにセッションエンコーディングを EBCDICANY または ASCIIANY に設定した SAS/SHARE サーバーを使用する場合にのみ関係します。このコンピューティング環境は、SAS 9 より前の製品ではより一般的でした。

参照項目: SAS/SHARE User's Guide の LIBNAME ステートメント

構文

RENCODING=*ASCII-encoding-value* | *EBCDIC-encoding-value*

説明**ASCII-encoding-value**

UNIX および Windows での ASCII エンコーディングの有効な値のリストについては、“[SAS セッションのエンコーディング値](#)” (573 ページ)を参照してください。

EBCDIC-encoding-value

z/OS での EBCDIC エンコーディングの有効な値のリストについては、“[SAS セッションのエンコーディング値](#)” (573 ページ)を参照してください。

詳細

エンコーディングが混在する環境(たとえば、Latin1 や Latin2 などの互換性のないエンコーディングを使用する SAS/SHARE クライアントセッション)で SAS/SHARE を使用する場合は、次のオプションを設定できます。

- SAS/SHARE サーバーセッションでは、SAS システムオプション ENCODING=EBCDICANY または ENCODING=ASCIIANY を設定します。
- SAS/SHARE クライアントセッションでは、次の条件において、LIBNAME ステートメントで RENCODING=オプションを設定します。
 - ASCII ベースのエンコーディングを使用するクライアントセッションが EBCDICANY サーバーにアクセスする場合
 - EBCDIC ベースのエンコーディングを使用するクライアントセッションが ASCIIANY サーバーにアクセスする場合

RENCODING=オプションにより、SAS/SHARE クライアントは、クライアントセッションのエンコーディングとして、またはそのエンコーディングからトランスコードするときにサーバーのデータに設定されたと想定するエンコーディングを指定できます。

SAS 9 および 9.2 を使用するとき、SAS/SHARE クライアント/サーバーセッションで複数の SBCS または DBCS エンコーディングのデータを処理する場合は、UTF8 エンコーディングを使用するようお勧めします。UTF8 セッションエンコーディングを実行する Unicode サーバーの詳細については、<http://rnd.sas.com/sites/i18n/i18ndocs/i18nsupport/Pages/SAS%20Technical%20Papers.aspx> にアクセスし、*SAS 9.1.3 Service Pack 4 in a Unicode Environment* および *Processing Multilingual Data with the SAS® 9.2 Unicode Server* を検索してください。

比較

SAS 9 および 9.2 では、UTF8 エンコーディングを使用して、従来の複数の SBCS または DBCS エンコーディングの文字を含む多言語データを SAS データセットで保持できます。SAS/SHARE を使用してそのデータへの更新アクセスを共有するには、UTF8 のセッションエンコーディングを使用して SAS/SHARE サーバーを実行する必要があります。SAS では、必要に応じてデータをクライアントのエンコーディングにトランスコードします。

SAS 9 より前は、SAS/SHARE クライアントと SAS/SHARE サーバーが共通のアーキテクチャで実行された場合(たとえば、クライアントとサーバーが UNIX マシン上で実行されていました)、文字データの自動エンコーディングは実行されませんでした。そのため、単一の SAS/SHARE サーバー内で異なる EBCDIC または ASCII エンコーディングのデータセットにアクセスするアプリケーションや、1 つのデータセット内にさまざまなエンコーディングが混在するデータセットにアクセスするアプリケーションを作成することが可能でした。この方法はあまり一般的ではなく、異なる動作環境で実行するクライアントからトランスコーディングテーブルを慎重にプログラミングする必要がありました。

次のステップは、必要に応じて SAS 9 で混在するエンコーディングを保持する方法を示しています。

- SAS/SHARE サーバーは、EBCDIC エンコーディングが混在する場合は EBCDICANY のセッションエンコーディングを、ASCII エンコーディングが混在する場合は ASCIIANY のセッションエンコーディングを使用して実行する必要があります。

これにより、バージョン 8 以前のリリースの動作を復元し、同じ EBCDIC または ASCII ファミリーでの異なるクライアントおよびサーバーエンコーディング間の自動的な文字トランスコーディングを回避します。つまり、次の条件下ではトランスコーディングが行われません。

- クライアントセッションエンコーディングが EBCDIC エンコーディングで、サーバーセッションエンコーディングが EBCDICANY の場合
- クライアントセッションエンコーディングが ASCII エンコーディングで、サーバーセッションエンコーディングが ASCIIANY の場合
- ASCIIANY または EBCDICANY サーバーと同じエンコーディングファミリーを共有しない SAS/SHARE クライアントは、そのサーバーにアクセスする最初の LIBNAME ステートメントで RENCODING=オプションを使用して、必要なトランスコーディングを制御できます。

たとえば、ポーランド語ロケールで実行される ASCII クライアントは、z/OS EBCDICANY サーバーにアクセスし、RENCODING=EBCDIC870 と指定して、ポーランド語でエンコードされたデータを含むことをクライアントが認識するデータにアクセスできます。ドイツ語ロケールで実行される別の ASCII クライアントは、同じ z/OS EBCDICANY サーバーにアクセスし、RENCODING=EBCDIC1141 と指定して、ドイツ語のデータを含むことをクライアントが認識しているデータにアクセスできます。同様に、ASCIIANY サーバーにアクセスする EBCDIC クライアントは、LIBNAME ステートメントの RENCODING=オプションを使用して、アクセスしようとするデータの正確な ASCII エンコーディングを指定できます。

関連項目:

概念的な情報:

- “トランスコーディングの概要” (27 ページ)

TRANSCODE=オプション

文字変数をトランスコードするかどうかを指示する、ATTRIB ステートメントの属性(1 つ以上の変数に出力形式、入力形式、ラベル、長さを関連付ける)を指定します。

該当要素: DATA ステップの ATTRIB ステートメント

カテゴリ: 情報

種類: 宣言型

参照項目: 現在の動作環境(Windows `xisError - link not found - The element n18ovvak3lqhlzn1d7tif15rhkc4 was not found in the link database`、UNIX `xisError - link not found - The element n12k1uxre4a7fen1e3wgp8n76cuc was not found in the link database`、z/OS)向けのドキュメント内の ATTRIB ステートメント

構文

ATTRIB *variable-list(s) attribute-list(s)* ;

必須引数

variable-list

属性に関連付ける変数の名前を付けます。

ヒント: SAS で許可される形式の変数リストが表示されます。

attribute-list

variable-list に割り当てる属性を 1 つ以上指定します。ATTRIB ステートメントには複数の属性を指定できます。全属性のリストについては、“ATTRIB Statement” in *SAS Statements: Reference* を参照してください。

TRANSCODE=YES|NO

文字変数をトランスコードするかどうかを指定します。トランスコーディングを実行しないようにするには、TRANSCODE=NO を使用します。詳細については、“トランスコーディングの概要” (27 ページ)を参照してください。

デフォルト: YES

制限事項: SAS Workspace Server クライアントによっては、TRANSCODE=NO 属性がサポートされない場合があります。TRANSCODE=NO が指定された変数は、SAS 9.3 では返されません。SAS 9.3 より前のリリースでは、TRANSCODE=NO が指定された変数はトランスコードされます。以前の SAS リリースでは、TRANSCODE=NO 属性が指定された変数を含む SAS 9.3 データセットにアクセスできません。

操作:

VTRANSCODE 関数および VTRANSCODEX 関数を使用すると、文字変数についてトランスコーディングがオンとオフのどちらであるかを返すことができます。

TRANSCODE=属性が NO に設定された文字変数がデータセット内にあると、PROC CONTENTS は、データセット内の変数ごとに、TRANSCODE=値が含まれるトランスコーディング列を出力します。データセット内のすべ

ての変数がデフォルトの TRANSCODE=値(YES)に設定された場合、トランスコーディング列は出力されません。

例

例 1: TRANSCODE=オプションと SET ステートメントの併用

SET ステートメントを使用して複数のデータセットからデータセットを 1 つ作成すると、出力データセット内の変数の TRANSCODE=属性が、最初のデータセット内の変数の TRANSCODE=値と同じに設定されます。この例では、B が最初のデータセットで、データセット B 内の Z の TRANSCODE=属性が NO であるため、データセット A 内の変数 Z の TRANSCODE=属性は NO です。

```
data b;
length z $4;
z = 'ice';
attrib z transcode = NO;
data c;
length z $4;
z = 'snow';
attrib z transcode = YES;
data a;
set b;
set c;
/* Check transcode setting for variable Z */
rc1 = vtranscode(z);
put rc1=;
run;
```

例 2: TRANSCODE=オプションと MERGE ステートメントの併用

MERGE ステートメントを使用して複数のデータセットからデータセットを 1 つ作成すると、出力データセット内の変数の TRANSCODE=属性が、最初のデータセット内の変数の TRANSCODE=値と同じに設定されます。この例では、C が最初のデータセットで、データセット C 内の Z の TRANSCODE=属性が YES であるため、データセット A 内の変数 Z の TRANSCODE=属性は YES です。

```
data b;
length z $4;
z = 'ice';
attrib z transcode = NO;
data c;
length z $4;
z = 'snow';
attrib z transcode = YES;
data a;
merge c b;
/* Check transcode setting for variable Z */
rc1 = vtranscode(z);
put rc1=;
run;
```

注: TRANSCODE=属性は、変数が入力データセットまたは ATTRIB TRANSCODE=ステートメントで最初に検出されたときに設定されます。SET または MERGE ステートメントが、ATTRIB TRANSCODE=ステートメントよりも前に現れ、TRANSCODE=属性が SET ステートメントと矛盾する場合、エラーメッセージが表示されます。

関連項目:**関数:**

- “VTRANSCODE 関数” (313 ページ)
- “VTRANSCODEX 関数” (314 ページ)

TRANTAB=オプション

出力ファイルが適切になるよう、SAS ファイルの文字データをトランスコードする場合に使用する変換テーブルを指定します。

該当要素: ODS MARKUP ステートメントと ODS RTF ステートメント

カテゴリ: ODS: サードパーティの出力形式

構文

TRANTAB = (*translation-table*)

オプション引数*translation-table*

出力ファイルに使用する変換テーブルを指定します。変換テーブルは、文字セットの文字(通常の文字、表語文字、数字、句読点、記号、コントロール文字など)を数値にマッピングするエンコーディング方法です。変換テーブルの例として、文字を EBCDIC から ASCII-ISO に変換するものがあります。*table-name* には、SAS で提供される変換テーブルまたはユーザー定義の変換テーブルを指定できます。値は、SASUSER.PROFILE カタログまたは SASHELP.HOST カタログの SAS カタログエントリ名である必要があります。

詳細

変換テーブルは、各国語の要件をサポートするために SAS 6 で導入されました。SAS 8.2 では、変換テーブルの直接使用より優れた、LOCALE=システムオプションが導入されました。SAS 9.1 では、下位互換性を維持するために TRANTAB=オプションがサポートされています。ただし、SAS 8.2 以降のリリースでは、LOCALE=システムオプションを使用することをお勧めします。

注: SAS 9.3 では、下位互換性を維持するため、ODS MARKUP の TRANTAB=オプションを使用します。エンコーディングを指定する場合、LOCALE=システムオプションが推奨されます。

関連項目:**概念的な情報:**

- “トランスコーディングと変換テーブル” (28 ページ)
- “各国語サポート関連のロケール” (5 ページ)

システムオプション:

- “TRANTAB=システムオプション” (476 ページ)

- “LOCALE システムオプション” (470 ページ)

プロシジャ:

- 17 章, “TRANTAB プロシジャ,” (521 ページ)

ステートメント:

- `xisError - link not found - The element n0onpp2holuauhn1gyb7lzx6knr7 was not found in the link database`
- `xisError - link not found - The element p1vvsv8ucnjzjnn1wq5wrlp74mdb was not found in the link database`

XMLENCODING=オプション

外部ドキュメントのインポートまたはエクスポート時に、XML ドキュメントのエンコーディングより優先される値を指定します。

該当要素: XML エンジンの LIBNAME ステートメント

カテゴリ: データアクセス

構文

`XMLENCODING='encoding-value'`

詳細

XML エンジンの LIBNAME ステートメントは、外部ドキュメントをインポートまたはエクスポートするために、SAS ライブラリ参照名を XML ドキュメントに関連付けます。

比較

オプション

encoding-value

外部ファイルの読み込み、書き出し、コピー、保存に使用するエンコーディングを指定します。XMLENCODING=の値は、外部ファイルのエンコーディングが現在のセッションエンコーディングとは異なることを示します。

詳細については、“データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値” (563 ページ)を参照してください。

encoding-value のデフォルトは現在のセッションのエンコーディングです。

関連項目:

ステートメント:

- *SAS XML LIBNAME Engine: ユーザーガイド*

TRANTAB ステートメント

SAS ファイルをエクスポートまたは転送するための文字データのトランスコーディング時に使用する、変換テーブルを指定します。

- 該当要素:** CPORT プロシジャ、UPLOAD プロシジャ、DOWNLOAD プロシジャ
- 制限事項:** 1 つの TRANTAB ステートメントに対して指定できる変換テーブルは 1 つのみです。追加の変換テーブルを指定するには、追加の TRANTAB ステートメントを使用します。
- 操作:** TRANTAB ステートメントでは、エクスポートまたは転送される SAS ファイルの文字セットに適用する、カスタマイズした変換テーブル(たとえば、EBCDIC 文字の ASCII 文字へのマッピング)を指定します。TRANTAB=システムオプションでは、ファイル転送を含む、SAS セッションで使用される変換テーブルを指定します。

構文

```
TRANTAB NAME=translation-table-name <TYPE=(etype-list)><OPT=DISP | SRC | (DISP SRC)>>;
```

必須引数

NAME=*translation-table-name*

エクスポート(PROC CPORT)または転送(PROC UPLOAD または PROC DOWNLOAD)する SAS カタログに適用する変換テーブルの名前を指定します。*translation-table-name* は、SASUSER.PROFILE カタログまたは SASHELP.HOST カタログのいずれかのカタログエントリ名 *t* として指定されます。SASUSER.PROFILE カタログが最初に検索され、次に SASHELP.HOST カタログが検索されます。

大半はデフォルトの変換テーブルが使用されますが、アプリケーションに異なる各国語文字が必要な場合など、追加の変換テーブルの適用が必要になることもあります。

デフォルト以外の変換テーブルを指定するには、次の 2 つの方法があります。

- プロシジャの起動時に変換テーブルを指定するには、必要に応じて、プロシジャで TRANTAB ステートメントを使用します。
- SAS セッションまたはジョブ全体(すべてのファイルエクスポートまたは転送を含む)に変換テーブルを指定するには、TRANTAB=システムオプションを使用します。

オプション引数

TYPE=(*etype-list*)

指定したタイプのエントリにのみ変換テーブルを適用します。*etype-list* には、1 つ以上のエントリのタイプを指定できます。たとえば、カタログエントリのタイプには、DATA や FORMAT などがあります。*etype-list* に指定するエントリタイプが 1 つのみの場合、かっこを省略します。

デフォルトで、UPLOAD、DOWNLOAD および CPORT プロシジャは、指定したすべてのカタログエントリに変換テーブルを適用します。

OPT=DISP | SRC | (DISP SRC)

OPT=DISP 指定したカタログエントリのうち、ウィンドウ表示を作成するエントリにのみ変換テーブルを適用します。

OPT=SRC 指定したカタログエントリのうち、タイプが SOURCE のエントリにのみ変換テーブルを適用します。

OPT=(DISP SRC) 指定したカタログエントリのうち、ウィンドウ表示を作成するか、タイプが SOURCE のエントリにのみ変換テーブルを適用します。

OPT=オプションを指定しない場合、UPLOAD または DOWNLOAD プロシジャは、変換テーブルを指定したカタログ内のすべてのエントリに適用します。

デフォルト: PROC CPORT、PROC UPLOAD および PROC DOWNLOAD は、変換テーブルを指定したカタログ内のすべてのエントリとデータセットに適用します。

詳細

変換テーブルは、各国語の要件をサポートするために SAS 6 で導入されました。SAS 8.2 では、変換テーブルの直接使用より優れた、LOCALE=システムオプションが導入されました。SAS 9.3 では、TRANTAB ステートメントが下位互換性を維持するためにサポートされています。ただし、SAS 8.2 以降のリリースでは、LOCALE=システムオプションを使用することをお勧めします。詳細については、TS-639: Data Conversion Issues in V6-V8 を参照してください。このテクニカルサポートメモでは、英語以外の言語を使用する顧客向けの情報を提供しています(<http://support.sas.com/techsup/technote/ts639.pdf>)。

PROC CPORT はネットワークを介して SAS ファイルをエクスポートする場合に使用します。PROC UPLOAD と PROC DOWNLOAD はネットワークを介して SAS ファイルを転送する場合に使用します。

TRANTAB ステートメントの使用時、PROC UPLOAD または PROC DOWNLOAD ステートメントに INCAT=および OUTCAT=オプションを指定する必要があります。

例

例 1

次の情報をプロシジャ機能に適用します。

- PROC CPORT ステートメントオプション: FILE=
- TRANTAB ステートメントオプション: TYPE=

この例では、カスタマイズした変換テーブルを、PROC CPORT がエクスポートする前にトランスポートファイルに適用する方法を示します。この例では、カスタマイズした変換テーブル TTABLE1 はすでに作成されたと想定します。

例 2: プログラム

ライブラリ参照を割り当てます。LIBNAME ステートメントと FILENAME ステートメントには、それぞれソースライブラリのライブラリ参照名とトランスポートファイルのファイル参照名を割り当てます。

```
libname source
'\\sashq\root\pub\pubdoc\doc\901\authoring\proc\miscsrc\sasfiles\cport';
filename tranfile 'trans3';
proc trantab table=ascii;
save table=ttable1;
```

```
libname source 'SAS data-library';
filename tranfile 'transport-file'
host-option(s)-for-file-characteristics;
```

指定した変換を適用します。TRANTAB ステートメントでは、カスタマイズした変換テーブル TTABLE1 で指定した変換を適用します。TYPE=によって、変換の適用対象が FORMAT エントリのみに制限されます。

```
proc cport catalog=source.formats file=tranfile;
trantab name=ttable1 type=(format);
run;
```

例 3: SAS ログ

NOTE: Proc CPORT はカタログ SOURCE.FORMATS の移送を開始します。
NOTE: カタログのエントリ数は 2 です。最大レコード長は 104 です。
NOTE: エントリ REVENUE.FORMAT を移送します。
NOTE: エントリ DEPT.FORMATC を移送します。

関連項目:

概念的な情報:

- [“各国語サポート関連のトランスコーディング” \(27 ページ\)](#)

システムオプション:

- [“TRANTAB=システムオプション” \(476 ページ\)](#)

プロシジャ:

- [17 章, “TRANTAB プロシジャ,” \(521 ページ\)](#)
- Chapter 15, “CPORT Procedure” in *Base SAS Procedures Guide*
- Chapter 23, “UPLOAD Procedure” in *SAS/CONNECT User's Guide*
- Chapter 24, “DOWNLOAD Procedure” in *SAS/CONNECT User's Guide*

10 部

各国語サポート関連のプロシジャ

16 章	
DBCSTAB プロシジャ	515
17 章	
TRANTAB プロシジャ	521

16 章

DBCSTAB プロシジャ

概要: DBCSTAB プロシジャ	515
DBCSTAB プロシジャの用途	515
構文: DBCSTAB プロシジャ	515
PROC DBCSTAB ステートメント	516
例: DBCSTAB プロシジャ	517
例 1: DBCSTAB プロシジャを使用した変換テーブルの作成	517
例 2: DBCSTAB プロシジャを使用した日本語変換テーブルの作成	518

概要: DBCSTAB プロシジャ

DBCSTAB プロシジャの用途

DBCSTAB プロシジャは SAS でサポートされる DBCS に対する変換テーブルを作成します。

次のような場合は、DBCSTAB プロシジャを使用して既存の DBCS 変換テーブルを変更します。

- 使用している DBCS エンコーディングシステムが SAS によって完全にサポートされていない場合
- 使用している DBCS エンコーディングシステムに標準外の変換テーブルがある場合

有効な DBCSTYPE=値が使用できない場合に、DBCSTAB プロシジャを使用します。有効な DBCSTYPE=値は動作環境に依存します。このような場合、DBCSTAB プロシジャを使用して似た変換テーブルを変更してから、TRANTAB オプションで新しいテーブルを使用するように指定できます。

構文: DBCSTAB プロシジャ

```
PROC DBCSTAB TABLE=table-name
<BASETYPE=base-type> <CATALOG=<libref.>catalog-name>
<DATA=<libref.>table-name> <DBCSLANG=language>
<DESC='description'> <FORCE> <VERIFY> <VERBOSE>;
```

PROC DBCSTAB ステートメント

DBCS に対する変換テーブルを作成します。

構文

```
PROC DBCSTAB TABLE=table-name  
<option(s)>;
```

必須引数

TABLE=

作成する 2 バイトコードテーブルの名前を指定します。このテーブル名は、CATALOG=オプションで指定されたカタログ内のタイプ DBCSTAB のエントリになります。カタログのデフォルト名は、SASUSER.DBCS です。

別名: NAME=, N=

オプション引数

BASETYPE=base-type

2 バイトコードテーブル変換の基準となるタイプを指定します。このオプションを使用すると、作成されるテーブルの数を減らすことができます。

BASETYPE=を指定すると、すべての 2 バイトコードはまず基準コードに変換されてから、要求されたコードへ変換されます。n 個のコードがある場合、n(n-1)回の変換を行う必要があります。

別名: BTYPE=

CATALOG=<libref.>catalog-name

テーブルの保存先となるカタログの名前を指定します。カタログが存在しない場合は、作成されます。

デフォルト: SASUSER.DBCS

DATA=<libref.>table-name

2 バイトコードテーブルを作成するためのデータを指定します。テーブルを作成するには、複数の 2 バイト文字変数が必要となります。DBCSTYPE システムオプションの値と等しく、KCVT 関数で認識される変数名を使用します。

DBCSLANG=language

2 バイトコードテーブルで使用される言語を指定します。このオプションの値は、DBCSLANG システムオプションの値と一致する必要があります。

別名: DBLANG

DESC='description'

エントリの DESCRIPTION フィールドに設定するテキスト文字列を指定します。

FORCE

エラーがある場合でも変換テーブルを生成します。

VERIFY

各コードの入力テーブルのデータ範囲を確認します。このオプションは、無効な 2 バイトコードがないかどうかを確認するために使用されます。

VERBOSE

DBCS テーブル作成時に統計の詳細を出力します。

能なモジュールに含まれるシステムを SAS System で使用するようになりました。
systab 名には、JAPANESE、KOREAN、CHINESE、TAIWANESE が使用できます。
10 番目の引数は、DBCS ユーザーテーブルを指定します。

```
options trantab=(,,,,,,,,systab); /* ninth argument */
```

例 2: DBCSTAB プロシジャを使用した日本語変換テーブルの作成

要素: PROC DBCSTAB ステートメントオプション:

```
TABLE=  
DATA=  
DBLANG=  
BASETYPE=  
VERIFY
```

プログラム

```
data ja_jpn;  
length ibm jis euc pcibm $2.;  
ibm='4040'x;  
jis='2121'x;  
euc='a1a1'x;  
pcibm='8140'x;  
run;  
  
proc dbcstab  
table=japanese  
data=ja_jpn  
dblang=japanese  
basetype=jis  
verify;  
run;
```

SAS ログ

```
1 proc dbcstab  
2 table=ja_jpn  
3 data=work.ja_jpn  
4 dlang=japanese  
5 basetype=jis  
6 verify;  
7 run;
```

NOTE: JIS の Base テーブルを作成しました。

NOTE: JIS の IBM テーブルを作成しました。

NOTE: JIS の PCIBM テーブルを作成しました。

NOTE: JIS の EUC テーブルを作成しました。

NOTE: IBM の Base テーブルを作成しました。

NOTE: IBM の JIS テーブルを作成しました。

NOTE: PCIBM の Base テーブルを作成しました。

NOTE: PCIBM の JIS テーブルを作成しました。

NOTE: EUC の Base テーブルを作成しました。

NOTE: EUC の JIS テーブルを作成しました。

NOTE: 10 DBCS テーブルを作成しました。各テーブルは 1 DBCS 文字です。

NOTE: 各テーブルのサイズは 2 バイトです。

NOTE: 要求したテーブルのメモリのサイズは 612 です。

NOTE: データセット WORK.JA_JPN から 1 オブザベーションを読み込みました。

17 章

TRANTAB プロシジャ

概要: TRANTAB プロシジャ	521
概念: TRANTAB プロシジャ	522
PROC TRANTAB で使用する変換テーブルと文字セットについて	522
PROC TRANTAB を使用した変換テーブルの保存	523
PROC TRANTAB を使用した SAS 変換テーブルの変更	523
PROC TRANTAB 外での変換テーブルの使用	523
構文: TRANTAB プロシジャ	525
PROC TRANTAB ステートメント	526
CLEAR ステートメント	527
INVERSE ステートメント	527
LIST ステートメント	527
LOAD ステートメント	528
REPLACE ステートメント	529
SAVE ステートメント	530
SWAP ステートメント	530
例: TRANTAB プロシジャ	531
例 1: 変換テーブルの表示	531
例 2: 変換テーブルの作成	532
例 3: 開始位置に 10 進値を指定した編集	535
例 4: 開始位置に引用符付き文字を使用した編集	537
例 5: 逆変換テーブルの作成	539
例 6: 並べ替えへのさまざまな変換テーブルの使用	540
例 7: テーブル 1 とテーブル 2 の編集	542

概要: TRANTAB プロシジャ

TRANTAB プロシジャは、カスタマイズした変換テーブルを作成、編集、表示します。また、PROC TRANTAB を使用して、SAS が提供する変換テーブルを表示および変更することもできます。これらの SAS 提供のテーブルは、SASHELP.HOST カタログに保存されています。作成またはカスタマイズした変換テーブルは、SASUSER.PROFILE カタログに格納されます。変換テーブルには TRANTAB のエントリタイプがあります。

変換テーブルは、1 つの(コーディングされた)文字セットの値を別の文字セットの値に変換するときに使用する、動作環境に固有の SAS カタログエントリです。変換テーブルは、次の 2 つのテーブルにわかれています。テーブル 1 は、ASCII から EBCDIC などの変換で使用します。テーブル 2 は、EBCDIC から ASCII などの反対(逆)の変換

で使用します。変換テーブルの各半分は 256 の 2 桁位置の配列で、それぞれの位置に、コーディングされた文字に対応する 1 バイトの符号なし数値があります。

SAS System では、次の目的で変換テーブルを使用します。

- SORT プロシジャの照合順序を決定します。
- CPORT および CIMPORT プロシジャでファイルを転送するときに、トランスポート形式変換を実行します。
- SAS/CONNECT または SAS/SHARE でリモートデータにアクセスするときに、動作環境間の変換を実行します。
- IBM 環境で SAS/GRAPH を実行するときに、動作環境とグラフィックデバイス間のデータ通信を促進します。
- 英語(U.S.)以外の各国語文字セットに適応します。

PROC TRANTAB プロシジャを実行しても、出力はありません。SAS ログの変換テーブルとメモを表示できます。

注: 変換テーブルは、各国語の要件をサポートするために SAS 6 で導入されました。SAS 8.2 では、変換テーブルの直接使用より優れた、LOCALE=システムオプションが導入されました。SAS 9.2 では、TRANTAB プロシジャが下位互換性のためにサポートされています。ただし、SAS 8.2 以降のリリースでは、LOCALE=システムオプションを使用することをお勧めします。PROC TRANTAB は、対話型のプロシジャです。PROC TRANTAB ステートメントをサブミットすると、PROC TRANTAB ステートメントを繰り返すことなく、引き続きステートメントを入力して実行できます。プロシジャを終了するには、QUIT ステートメントをサブミットするか、別の DATA または PROC ステートメントをサブミットします。

概念: TRANTAB プロシジャ

PROC TRANTAB で使用する変換テーブルと文字セットについて

変換テーブルの k 番目の要素は、順序どおりに並べた文字セットの k 番目の要素に対応します。たとえば、変換テーブルの位置 00 (1 バイト目)には、コード化された値が含まれ、その値は、順序どおりに並べた文字セットの最初の要素に対応します。動作環境の文字セットに含まれる文字の位置を判別するには、SAS 関数の RANK を使用します。次の例は、RANK の使用方法を示しています。

```
data _null_;
x=rank('a');
put "The position of a is " x ".";
run;
```

SAS ログには、次のメッセージが出力されます。**"a の位置は 97 です。"**

変換テーブルの各位置には、0 ('00'x)から 255 ('FF'x)の範囲内にある 16 進数が含まれます。16 進値の末尾には常に x が付きます。連続する 1 つ以上の 16 進値を表すには、それらの値をまとめて引用符で囲み、その後に x を 1 つ付けます。たとえば、連続する 3 個の 16 進値からなる文字列は、'08090A'x と表すことができます。SAS ログには、変換テーブルの各行が、引用符で囲まれて、後ろに x が付けられた 16 進値で表示されます。SAS ログではまた、テーブル内の位置に対応する縦方向と横方向の余白に参照番号が表示されます。“例 1: 変換テーブルの表示” (531 ページ) には、SAS ログでの変換テーブルの表示方法が説明されています。

PROC TRANTAB を使用した変換テーブルの保存

PROC TRANTAB を使用してカスタマイズした変換テーブルを作成すると、プロシジャが自動的にテーブルを SASUSER.PROFILE カタログに保存します。これにより、他のユーザーに影響を与えることなくカスタマイズした変換テーブルを使用できます。SORT プロシジャまたは GOPTIONS ステートメントで変換テーブルを指定すると、最初に SASUSER.PROFILE カタログでテーブルが検索されます。指定した変換テーブルが SASUSER.PROFILE カタログにない場合、次に SASHELP.HOST カタログが検索されます。

作成した変換テーブルをグローバルにアクセスできるようにするには、SAS インストール担当者にこのテーブルを SASUSER.PROFILE カタログから SASHELP.HOST カタログに(CATALOG プロシジャを使用して)コピーするように依頼してください。SASHELP.HOST カタログでテーブルが見つからない場合、続けて SASHELP.LOCALE でテーブルが検索されます。

PROC TRANTAB を使用した SAS 変換テーブルの変更

SAS 提供の変換テーブルがニーズに合わない場合、PROC TRANTAB を使用してテーブルを編集したり、新しいテーブルを作成したりできます。つまり、SAS テーブルを指定する PROC TRANTAB ステートメントを発行し、テーブルを編集してから、SAVE ステートメントを使用してそのテーブルを保存できます。変更した変換テーブルは、SASUSER.PROFILE カタログに保存されます。SAS インストール担当者の場合、次の例のように、PROC TRANTAB を使用して変換テーブルを変更し、CATALOG プロシジャを使用して変更したテーブルを SASUSER.PROFILE カタログから SASHELP.HOST カタログにコピーできます。

```
proc catalog c=sasuser.profile;
copy out=sashelp.host entrytype=trantab;
run;
```

PROC TRANTAB を使用して SASHELP.HOST カタログに保存された変換テーブルを変更するには、そのデータライブラリとカタログへの更新(書き込み)アクセス権が必要です。

PROC TRANTAB 外での変換テーブルの使用

SORT プロシジャでの変換テーブルの使用

PROC SORT は、変換テーブルを使用して並べ替えで使用する照合順序を決定します。代替の変換テーブルは、PROC SORT の SORTSEQ=オプションで指定できます。たとえば、動作環境のデフォルトでは EBCDIC 順に並べ替えが行われる場合、ASCII 順に並べ替えを行うには、次のステートメントを発行して ASCII 変換テーブルを指定できます。

```
proc sort sortseq=ascii;
run;
```

PROC TRANTAB でカスタマイズした変換テーブルを作成し、PROC SORT で新しいテーブルを指定することもできます。このテーブルは、英語(U.S.)以外の言語に並べ替え順序を指定する場合に便利です。

変換テーブルを使って違う方法でデータを並べ替える例については、“[例 6: 並べ替えへのさまざまな変換テーブルの使用](#)” (540 ページ) を参照してください。並べ替えに使用できるテーブルと SORTSEQ=オプションの詳細については、“[SORTSEQ=システムオプション: UNIX、Windows および z/OS](#)” (474 ページ) を参照してください。

CPORT および CIMPORT プロシジャでの変換テーブルの使用

CPORT および CIMPORT プロシジャは、変換テーブルを使用して、ある動作環境からエクスポートして別の動作環境にインポートするカタログエントリ内の文字を変換します。PROC CPORT の TRANTAB ステートメントでは、SAS 提供の変換テーブルまたはカスタマイズした変換テーブルの名前を指定できます。詳細については、CPORT プロシジャの“TRANTAB ステートメント” (509 ページ) を参照してください。

リモートライブラリサービスでの変換テーブルの使用

リモートライブラリサービス(RLS)は、SAS 8 のリモートデータにアクセスするときに、変換テーブルを使用して文字を変換します。SAS/CONNECT および SAS/SHARE は、異なるエンコーディング標準を使用する 2 つの動作環境間でファイルを転送または共有する場合に、変換テーブルを使用して文字を変換します。

注: 詳細については、TS-706: How to use the %lswbatch macro

<http://support.sas.com/techsup/technote/ts706.pdf> を参照してください。

SAS/GRAPH での変換テーブルの使用

SAS/GRAPH では、ほとんどの場合、変換テーブルは IBM 動作環境で使用されません。グラフィックコマンドは IBM 動作環境から EBCDIC 表記で送信され、ASCII 表記で非同期グラフィックデバイスにアクセスする必要があるため、変換テーブルが必要になります。具体的には、SAS/GRAPH はこれらのデバイス用に ASCII 表記で内部的にコマンドストリームを作成しますが、コマンドを EBCDIC 表記に変換してから、デバイスへの転送を行う通信ソフトウェアに渡す必要があります。SAS/GRAPH は、内部的に変換テーブルを使用して、ASCII から EBCDIC への初期変換を行います。通信ソフトウェアは、コマンドストリームを再変換して ASCII 表記に戻してから、グラフィックデバイスにアクセスします。

変換テーブルは、動作環境に固有です。多くの場合、デフォルトの変換テーブルである SASGTAB0 か、SAS が提供するいずれかのグラフィック変換テーブルをそのまま使用できます。ただし、これらのテーブルでは正しく変換できないものがある場合、PROC TRANTAB を使用して独自の変換テーブルを作成できます。SASGTAB0 テーブルは、英語(U.S.)以外の言語の文字を検出すると正しく変換できない場合があります。

SAS/GRAPH に代替変換テーブルを指定するには、GOPTIONS ステートメントで TRANTAB=オプションを使用するか、デバイスエントリの TRANTAB デバイスパラメータを変更します。たとえば、次の GOPTIONS ステートメントでは GTABTCAM グラフィック変換テーブルを指定しています。

```
goptions trantab=gtabtcam;
```

SAS/GRAPH で使用される変換テーブルは、デバイス-動作環境間 変換および動作環境-デバイス間変換の両方を実行します。したがって、変換テーブルは 512 バイトで構成され、最初の 256 バイトはデバイス-動作環境間変換(IBM メインフレームでの ASCII から EBCDIC)に、次の 256 バイトは動作環境-デバイス間変換(IBM メインフレームでの EBCDIC から ASCII)に使用されます。PROC TRANTAB では、デバイス-動作環境間変換の変換テーブルの領域はテーブル 1 と見なされ、動作環境-デバイス間変換の領域はテーブル 2 と見なされます。ASCII 変換テーブル(SAS 提供の変換テーブル)のリストについては、“例 1: 変換テーブルの表示” (531 ページ) を参照してください。テーブルの両方の領域が表示されています。

IBM メインフレーム以外の動作環境では、変換テーブルを使用してドライバが作成したデータストリーム内の特定の文字を変換できます。たとえば、ドライバによって通常データストリーム内に縦棒が生成される場合、その縦棒のかわりに別の文字を生成するには、縦棒を別の文字に変換する変換テーブルを作成できます。

SAS/GRAPH で TRANTAB=オプションを使用して変換テーブルを指定する方法の詳細については、SAS/GRAPH Software: Reference, Version 6, First Edition, Volume 1 および Volume 2 を参照してください。

SAS/GRAPH は、キーボードで生成されたコードを指定された文字にマッピングする場合、および文字コードをグラフィック出力デバイスで要求されるコードにマッピングする場合に、キーマップおよびデバイスマップも使用します。これらのマップは SAS/GRAPH に固有です。詳細については、SAS Institute のテクニカルサポートにお問い合わせください。

構文: TRANTAB プロシジャ

ヒント: RUN グループ処理がサポートされます。

```
PROC TRANTAB TABLE=table-name <NLS>;
  CLEAR <ONE|TWO|BOTH>;
  INVERSE;
  LIST <ONE|TWO|BOTH>;
  LOAD TABLE=table-name <NLS>;
  REPLACE position value-1<...value-n>;
  SAVE <TABLE=table-name> <ONE|TWO|BOTH>;
  SWAP;
```

変換テーブルは各国語の要件をサポートするために SAS 6 で導入されました。SAS 8.2 では、変換テーブルの直接使用より優れた、LOCALE=システムオプションが導入されました。SAS 9.3 にておいては、TRANTAB プロシジャは下位互換性のためにサポートされています。ただし、SAS 8.2 以降のリリースでは、LOCALE=システムオプションを使用することをお勧めします。PROC TRANTAB は、対話型のプロシジャです。PROC TRANTAB ステートメントをサブミットすると、PROC TRANTAB ステートメントを繰り返すことなく、引き続きステートメントを入力して実行できます。プロシジャを終了するには、QUIT ステートメントをサブミットするか、別の DATA または PROC ステートメントをサブミットします。

ステートメント	タスク
“CLEAR ステートメント”	変換テーブルのすべての位置をゼロに設定する
“INVERSE ステートメント”	テーブル 1 の逆変換テーブルを作成する
“LIST ステートメント”	変換テーブルを 16 進表現で表示する
“LOAD ステートメント”	変換テーブルを編集できるようにメモリに読み込む
“REPLACE ステートメント”	変換テーブル内の文字を指定値で置換する
“SAVE ステートメント”	変換テーブルを SASUSER.PROFILE カタログに保存する

ステートメント	タスク
“SWAP ステートメント”	テーブル 1 をテーブル 2 と交換する

PROC TRANTAB ステートメント

変換テーブルを作成、編集、表示します。

ヒント: PROC TRANTAB ステートメントに誤ったテーブル名がある場合は、LOAD ステートメントを使用して正しいテーブルを読み込みます。PROC TRANTAB を再起動する必要はありません。新しいテーブルは、SAVE ステートメントを発行するまでカタログに保存されません。そのため、不要なテーブルがカタログに保存されることはありません。

構文

```
PROC TRANTAB TABLE=table-name <NLS>;
```

必須引数

TABLE=table-name

作成、編集または表示する変換テーブルを指定します。指定されたテーブル名は、8 文字以内の有効な 1 レベルの SAS 名である必要があります。

オプション引数

NLS

TABLE=引数で指定したテーブルが、SAS で必ず提供される 5 つの特殊な内部変換テーブルの 1 つであることを指定します。TABLE=引数に 5 つの特殊なテーブルのいずれかを指定する場合は、NLS オプションを使用する必要があります。NLS は各国語サポート(National Language Support)を表します。このオプションおよび関連する変換テーブルによって、英語以外の言語の文字を変換する手段が提供されます。変更した NLS テーブルを使用できるようにするには、SAS システムオプションの TRANTAB=にその名前を指定します。これらの特殊な変換テーブルのいずれかを読み込む場合、SAS ログには、テーブル 2 が初期化されていないという注意が表示されます。つまり、テーブル 2 はゼロのみが含まれた空のテーブルです。PROC TRANTAB が、これらの特殊なケースでの変換にテーブル 2 を使用することはありません。そのため、この注意は無視してかまいません。

SASXPT

ローカルからトランスポートへの出力形式変換テーブル(CPORT プロシジャで使用)。

SASLCL

トランスポートからローカルへの出力形式変換テーブル(CIMPORT プロシジャで使用)。

SASUCS

小文字から大文字への変換テーブル(UPCASE 関数で使用)。

SASLCS

大文字から小文字への変換テーブル(LOWCASE マクロで使用)。

SASCCL

文字分類テーブル(内部使用)。このテーブルには、各文字位置に対応するフラグバイトが含まれます。これらの位置は、各文字が属するクラスを示します。

CLEAR ステートメント

変換テーブルのすべての位置をゼロに設定します。新しいテーブルを作成するときに使用されます。

構文

```
CLEAR <ONE|TWO|BOTH>;
```

オプション引数

ONE | TWO | BOTH

ONE

テーブル 1 をクリアします。

TWO

テーブル 2 をクリアします。

BOTH

テーブル 1 とテーブル 2 を両方ともクリアします。

デフォルト: ONE

INVERSE ステートメント

変換テーブル内にテーブル 1 の逆変換テーブルを作成します。つまり INVERSE はテーブル 2 を作成します。

構文

```
INVERSE;
```

詳細

INVERSE では、複数の変換は保持されません。テーブル 1 で、2 つ(以上)の異なる文字が同一の値に変換されたとします。たとえば、"A"と"B"はどちらも"1"に変換されています。テーブル 2 に対して、INVERSE は最後に変換された文字を値として使用します。つまり、最初のテーブルでは"A"が"B"より前に現れると想定して、"1"を常に"B"に変換し、"A"には変換しません。並べ替えプログラムが正しく動作するには、逆変換テーブルが必要です。

LIST ステートメント

SAS ログに変換テーブルを 16 進表現で表示します。

構文

```
LIST <ONE|TWO|BOTH>;
```

オプション引数

ONE | TWO | BOTH

ONE

テーブル 1 を表示します。

TWO

テーブル 2 を表示します。

BOTH

テーブル 1 とテーブル 2 を両方とも表示します。

デフォルト: ONE

LOAD ステートメント

変換テーブルを編集できるようにメモリに読み込みます。

ヒント: LOAD は、PROC TRANTAB ステートメントに誤ったテーブル名を指定した場合に使用します。プロシジャを再起動せずに、正しい名前を指定できます。
1 つの PROC TRANTAB ステップで複数の変換テーブルを編集するには、LOAD を使用します (必ず最初のテーブルを保存してから、別のテーブルを読み込んでください)。

構文

```
LOAD TABLE=table-name <NLS>;
```

必須引数

TABLE=*table-name*

編集する既存の変換テーブルの名前を指定します。指定されたテーブル名は、有効な 1 レベルの SAS 名である必要があります。

オプション引数

NLS

TABLE=引数で指定したテーブルが、SAS で提供される 5 つの特殊な内部変換テーブルの 1 つであることを指定します。TABLE=引数に 5 つの特殊なテーブルのいずれかを指定する場合は、NLS オプションを使用する必要があります。

SASXPT

ローカルからトランスポートへの出力形式変換テーブル。

SASLCL

トランスポートからローカルへの出力形式変換テーブル。

SASUCS

小文字から大文字への変換テーブル。

SASLCS

大文字から小文字への変換テーブル。

SASCCL

文字分類テーブル。このテーブルには、各文字位置に対応するフラグバイトが含まれます。これらの位置は、各文字が属するクラスを示します。

NLS は各国語サポート(National Language Support)を表します。このオプションおよび関連する変換テーブルによって、英語以外の言語の文字をプログラム、表示およびファイルにマップする手段が提供されます。これらの特殊な変換テーブルのいずれかを読み込む場合、SAS ログには、テーブル 2 が初期化されていないという注意が表示されます。つまり、テーブル 2 はゼロのみが含まれた空のテーブルです。PROC TRANTAB が、これらの特殊なケースで変換にテーブル 2 を使用することはありません。

REPLACE ステートメント

指定された開始位置から変換テーブル内の文字を指定値で置き換えます。

別名: REP

ヒント: 編集内容を保存するには、SAVE ステートメントを発行する必要があります。

構文

REPLACE *position value-1<...value-n>*;

必須引数

position

変換テーブル内の置換の開始位置を指定します。変換テーブル内の編集可能な位置は、10 進値の 0 から 255 までです。位置を指定するには、次のいずれかの操作を実行します。

- 実際の位置を指定するには、10 進または 16 進の値を使用します。10 進値 (20 など) を指定すると、PROC TRANTAB は、テーブル内の位置 20 (21 バイト目) を検索します。16 進値 ('14x など) を指定すると、PROC TRANTAB は、指定された 16 進値と等しい 10 進値の位置 (この場合はテーブル内の位置 20、つまり 21 バイト目) を検索します。
- 引用符の付いた文字を使用します。PROC TRANTAB は、テーブル内の引用符付きの文字 (つまり、引用符で囲まれた文字の 16 進値) を検索し、その文字の位置を開始位置として使用します。たとえば、次の REPLACE ステートメントを指定すると、ステートメントでは最初に出現した 16 進値の "a" とその後続く 2 つの 16 進値を、"ABC" に相当する 16 進値で置き換えます。`replace 'a' 'ABC';`

この操作は、英数字を検索する必要があり、その実際の位置がわからない場合に便利です。引用符付き文字が見つからない場合、PROC TRANTAB ではエラーメッセージを表示して、ステートメントを無視します。

位置 256 から 511 (テーブル 2) までを編集するには、次の操作を実行します。

1. SWAP ステートメントを発行します。
2. 適切な REPLACE ステートメントを発行します。
3. 再び SWAP ステートメントを発行して、テーブルを再配置します。

value-1 <...value-n>

テーブル内の位置から配置する実際の値を、1 つ以上の 10 進、16 進または文字の定数で指定します。複数のタイプの値を組み合わせて使用することもできます。つまり、10 進値、16 進値および文字値を 1 つの REPLACE ステートメントに指定できます。“例 3: 開始位置に 10 進値を指定した編集” (535 ページ) は、REPLACE ステートメントでのこれら 3 つのタイプの値を組み合わせてについて説明します。

SAVE ステートメント

変換テーブルを SASUSER.PROFILE カタログに保存します。

構文

```
SAVE <TABLE=table-name> <ONE|TWO|BOTH>;
```

オプション引数

TABLE=table-name

現在のテーブルを保存するときの名前を指定します。テーブル名は、有効な 1 レベルの SAS 名である必要があります。

デフォルト: TABLE=オプションを省略すると、現在のテーブルは PROC TRANTAB ステートメントまたは LOAD ステートメントで指定した名前で作成されます。

ONE | TWO | BOTH**ONE**

テーブル 1 を保存します。

TWO

テーブル 2 を保存します。

BOTH

テーブル 1 とテーブル 2 を両方とも保存します。

デフォルト: BOTH

SWAP ステートメント

テーブル 1 をテーブル 2 と交換して、位置 256 から 511 までを編集できるようにします。

ヒント: テーブルを編集したら、再び SWAP ステートメントを発行してテーブルを再配置する必要があります。

構文

```
SWAP;
```

例: TRANTAB プロシジャ

例 1: 変換テーブルの表示

要素: LIST ステートメント

この例では、PROC TRANTAB を使用して SAS が提供する ASCII 変換テーブルを表示します。例はすべて UNIX 環境で作成されました。

オプションを設定し、変換テーブルを指定します。

```
options nodate pageno=1 linesize=80 pagesize=60;  
proc trantab table=ascii;
```

変換テーブルの半分を両方とも表示します。LIST BOTH ステートメントでは、変換を行うテーブルと逆変換を行うテーブルの両方を表示します。

```
list both;
```

SAS ログ

```
NOTE: Table specified is ASCII.
ASCII table 1:
0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000102030405060708090A0B0C0D0E0F'x
10 '101112131415161718191A1B1C1D1E1F'x
20 '202122232425262728292A2B2C2D2E2F'x
30 '303132333435363738393A3B3C3D3E3F'x
40 '404142434445464748494A4B4C4D4E4F'x
50 '505152535455565758595A5B5C5D5E5F'x
60 '606162636465666768696A6B6C6D6E6F'x
70 '707172737475767778797A7B7C7D7E7F'x
80 '808182838485868788898A8B8C8D8E8F'x
90 '909192939495969798999A9B9C9D9E9F'x
A0 'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'x
B0 'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'x
C0 'C0C1C2C3C4C5C6C7C8C9CACBCCDCECF'x
D0 'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0 'E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEF'x
F0 'F0F1F2F3F4F5F6F7F8F9FABFBCFDFEFF'x
```

```
ASCII table 2:
0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000102030405060708090A0B0C0D0E0F'x
10 '101112131415161718191A1B1C1D1E1F'x
20 '202122232425262728292A2B2C2D2E2F'x
30 '303132333435363738393A3B3C3D3E3F'x
40 '404142434445464748494A4B4C4D4E4F'x
50 '505152535455565758595A5B5C5D5E5F'x
60 '606162636465666768696A6B6C6D6E6F'x
70 '707172737475767778797A7B7C7D7E7F'x
80 '808182838485868788898A8B8C8D8E8F'x
90 '909192939495969798999A9B9C9D9E9F'x
A0 'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'x
B0 'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'x
C0 'C0C1C2C3C4C5C6C7C8C9CACBCCDCECF'x
D0 'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0 'E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEF'x
F0 'F0F1F2F3F4F5F6F7F8F9FABFBCFDFEFF'x
```

例 2: 変換テーブルの作成

要素: プロシジャの機能:
 LIST ステートメント
 REPLACE ステートメント
 SAVE ステートメント

この例では、PROC TRANTAB を使用してカスタマイズした変換テーブルを作成します。例はすべて UNIX 環境で作成されました。

システムオプションを設定し、編集する変換テーブルを指定します。

```
options nodate pageno=1 linesize=80
pagesize=60;
proc trantab table=newtable;
```


変換テーブル内の文字を指定された開始位置から置き換えます。この REPLACE ステートメントでは、テーブル内の開始位置 0 から値を配置します。REPLACE ステートメントには、任意の長さの 16 進文字列を使用できます。この例では、SAS ログに表示されたときの変換テーブルと同じ状態にするために長さ 16 の文字列を使用します。

```

replace 0
'00010203a309e57ff9ecc40b0c0d0e0f'x
'10111213a5e008e71819c6c51c1d1e1f'x
'c7fce9e2e40a171beaeb8efee050607'x
'c9e616f4f6f2fb04ffd6dca2b6a7501a'x
'20e1edf3faf1d1aabbabfa22e3c282b7c'x
'265facbdbca1abbb5f5f21242a293bac'x
'2d2f5fa6a6a6a62b2ba6a62c255f3e3f'x
'a62b2b2b2b2b2b2d2d603a2340273d22'x
'2b6162636465666768692d2ba6a62b2b'x
'2d6a6b6c6d6e6f7071722da62d2b2d2d'x
'2d7e737475767778787a2d2b2b2b2b2b'x
'2b2b2b5f5fa65f5f5fdf5fb65f5fb55f'x
'7b4142434445464748495f5f5f5f5f'x
'7d4a4b4c4d4e4f5051525f5f5fb15f5f'x
'5c83535455565758595a5f5ff75f5fb0'x
'30313233343536373839b75f6eb25f5f'x
;

```

テーブルを保存します。SAVE ステートメントでは、テーブルを PROC TRANTAB ステートメントで指定された名前保存します。デフォルトでは、テーブルは SASUSER.PROFILE カタログに保存されます。

```
save;
```

SAS ログ内に変換テーブルの半分を両方とも表示します。LIST BOTH ステートメントでは、変換を行うテーブルと逆変換を行うテーブルの両方を表示します。

```
list both;
```

SAS ログ

-->テーブル 2 を作成および編集します。テーブル 2 は空です。つまり、内容はすべて 0 です。テーブル 2 を作成するには、INVERSE ステートメントを使用できます

(See

.) テーブル 2 を編集するには、SWAP ステートメントと REPLACE ステートメントを使用できます。(See

.) " commented-out by old2new conversion -->

NOTE: 指定のテーブルは NEWTABLE です。

WARNING: テーブル NEWTABLE はありません。新規に作成します。

NOTE: NEWTABLE テーブル 1 が初期化されていません。

NOTE: NEWTABLE テーブル 2 が初期化されていません。

NOTE: テーブル NEWTABLE を保存します。

NOTE: 初期化されていないので、NEWTABLE テーブル 2 は保存されません。

NEWTABLE table 1:

```

0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '00010203A309E57FF9ECC40B0C0D0E0F'x
10 '10111213A5E008E71819C6C51C1D1E1F'x
20 'C7FCE9E2E40A171BEAEBE8EFEE050607'x
30 'C9E616F4F6F2FB04FFD6DCA2B6A7501A'x
40 '20E1EDF3FAF1D1AABABFA22E3C282B7C'x
50 '265FACBDBCA1ABBB5F5F21242A293BAC'x
60 '2D2F5FA6A6A6A62B2BA6A62C255F3E3F'x
70 'A62B2B2B2B2B2D2D603A2340273D22'x
80 '2B6162636465666768692D2BA6A62B2B'x
90 '2D6A6B6C6D6E6F707172DA62D2B2D2D'x
A0 '2D7E737475767778787A2D2B2B2B2B'x
B0 '2B2B2B5F5FA65F5F5FDF5FB65F5FB55F'x
C0 '7B4142434445464748495F5F5F5F5F'x
D0 '7D4A4B4C4D4E4F5051525F5F5FB15F5F'x
E0 '5C83535455565758595A5F5FF75F5FB0'x
F0 '30313233343536373839B75F6EB25F5F'x

```

NOTE: NEWTABLE テーブル 2 が初期化されていません。

NEWTABLE table 2:

```

0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000000000000000000000000000000'x
10 '000000000000000000000000000000'x
20 '000000000000000000000000000000'x
30 '000000000000000000000000000000'x
40 '000000000000000000000000000000'x
50 '000000000000000000000000000000'x
60 '000000000000000000000000000000'x
70 '000000000000000000000000000000'x
80 '000000000000000000000000000000'x
90 '000000000000000000000000000000'x
A0 '000000000000000000000000000000'x
B0 '000000000000000000000000000000'x
C0 '000000000000000000000000000000'x
D0 '000000000000000000000000000000'x
E0 '000000000000000000000000000000'x
F0 '000000000000000000000000000000'x

```

例 3: 開始位置に 10 進値を指定した編集

要素: LIST ステートメント
REPLACE ステートメント
SAVE ステートメント

この例では、“例 2: 変換テーブルの作成” (532 ページ) で作成した変換テーブルを編集します。REPLACE ステートメントで指定した 10 進値によって、テーブルへの変更の開始位置が示されます。

両方の SAS ログ内の縦矢印によって、変更の開始点が示されます。

例はすべて UNIX 環境で作成されました。

プログラム 1: 元のテーブルの表示

```
options nodate pageno=1 linesize=80 pagesize=60; proc trantab table=newtable;
list one;
```

プログラムの説明

システムオプションを設定し、編集する変換テーブルを指定します。

```
options nodate pageno=1 linesize=80 pagesize=60; proc trantab table=newtable;
```

元のテーブルを表示します。この LIST ステートメントでは、元の NEWTABLE 変換テーブルを表示します。

```
list one;
```

SAS ログ

元の NEWTABLE 変換テーブル

```
Table specified is NEWTABLE.
NOTE: NEWTABLE table 2 is uninitialized.
NEWTABLE table 1:
↓
0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '00010203A309E57FF9ECC40B0C0D0E0F'x
10 '10111213A5E008E71819C6C51C1D1E1F'x
20 'C7FCE9E2E40A171BEAEBE8EFEE050607'x
30 'C9E616F4F6F2FB04FFD6DCA2B6A7501A'x
40 '20E1EDF3FAF1D1AABABFA22E3C282B7C'x
50 '265FACBDBCA1ABBB5F5F21242A293BAC'x
60 '2D2F5FA6A6A6A62B2BA6A62C255F3E3F'x
70 'A62B2B2B2B2B2D2D603A2340273D22'x
80 '2B6162636465666768692D2BA6A62B2B'x
90 '2D6A6B6C6D6E6F7071722DA62D2D2D'x
A0 '2D7E737475767778787A2D2B2B2B2B'x
B0 '2B2B2B5F5FA65F5F5FDF5FB65F5FB55F'x
C0 '7B4142434445464748495F5F5F5F5F'x
D0 '7D4A4B4C4D4E4F5051525F5F5FB15F5F'x
E0 '5C83535455565758595A5F5FF75F5FB0'x
F0 '30313233343536373839B75F6EB25F5F'x
```

プログラム 2: テーブルの編集

```
replace 10
20 10 200 'x' 'ux' '092040'x;

save;

list one;
```

プログラムの説明

変換テーブル内の文字を指定された開始位置から置き換えます。この REPLACE ステートメントでは、10 進数で 10 の開始位置(元のテーブルの 11 バイト目)から、指定した値を基にバイト単位の置換を行います。

```
replace 10
20 10 200 'x' 'ux' '092040'x;
```

変更を保存します。SAVE ステートメントでは、NEWTABLE 変換テーブルに加えた変更を保存します。

```
save;
```

新しいテーブルを表示します。2 番目の LIST ステートメントでは、編集された NEWTABLE 変換テーブルを表示します。

```
list one;
```

SAS ログ

```
Saving table NEWTABLE.
NOTE: NEWTABLE table 2 will not be saved because it is uninitialized.
NEWTABLE table 1:
↓
0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '00010203A309E57FF9EC140AC8787578'x
10 '09204013A5E008E71819C6C51C1D1E1F'x
20 'C7FCE9E2E40A171BEAEBE8EFEE050607'x
30 'C9E616F4F6F2FB04FFD6DCA2B6A7501A'x
40 '20E1EDF3FAF1D1AABABFA22E3C282B7C'x
50 '265FACBDBCA1ABBB5F5F21242A293BAC'x
60 '2D2F5FA6A6A6A62B2BA6A62C255F3E3F'x
70 'A62B2B2B2B2B2D2D603A2340273D22'x
80 '2B6162636465666768692D2BA6A62B2B'x
90 '2D6A6B6C6D6E6F7071722DA62D2B2D2D'x
A0 '2D7E737475767778787A2D2B2B2B2B'x
B0 '2B2B2B5F5FA65F5F5FDF5FB65F5FB55F'x
C0 '7B4142434445464748495F5F5F5F5F'x
D0 '7D4A4B4C4D4E4F5051525F5F5FB15F5F'x
E0 '5C83535455565758595A5F5FF75F5FB0'x
F0 '30313233343536373839B75F6EB25F5F'x
```

出力詳細

位置 10 (11 バイト目)では、縦矢印が変換テーブルへの変更の開始位置を示しています。

11 バイト目では、10 進値の 20 (16 進値の 14)で 16 進値の C4 が置き換えられます。

12 バイト目では、10 進値の 10 (16 進値の 0A)で 16 進値の 0B が置き換えられます。

13 バイト目では、10 進値の 200 (16 進値の C8)で 16 進値の 0C が置き換えられます。

14 バイト目では、文字'x' (16 進値の 78)で 16 進値の 0D が置き換えられます。

15 バイト目および 16 バイト目では、文字'ux' (16 進値の 75 と 78)で 16 進値の 0E と 0F が置き換えられます。

17 バイト目、18 バイト目および 19 バイト目では、16 進値の 092040 で 16 進値の 101112 が置き換えられます。

例 4: 開始位置に引用符付き文字を使用した編集

要素: LIST ステートメント
LOAD ステートメント
REPLACE ステートメント
SAVE ステートメント

この例では、内容が固定された ASCII 変換テーブルを編集して新しい変換テーブルを作成します。REPLACE ステートメントで指定した引用符付き文字と等しい 16 進値が最初に出現する位置が、テーブルへの変更開始位置になります。この方法は、テーブルへの変更を開始する正確な位置を知る必要がないという点が、“例 3: 開始位置に 10 進値を指定した編集” (535 ページ)とは異なります。PROC TRANTAB が、ユーザーに代わって正確な位置を検索します。

編集されたテーブルは、新しい名前で作成されます。両方の SAS ログ内の横矢印は、変換テーブル内の編集された行を示します。

例はすべて UNIX 環境で作成されました。

プログラム 1: 元のテーブルの表示

```
options nodate pageno=1 linesize=80 pagesize=60;
proc trantab table=ascii;

list one;
```

プログラムの説明

システムオプションを設定し、編集する変換テーブルを指定します。

```
options nodate pageno=1 linesize=80 pagesize=60;
proc trantab table=ascii;
```

変換テーブルを表示します。この LIST ステートメントでは、SAS ログに元の変換テーブルを表示します。

```
list one;
```

SAS ログ

```
NOTE: Table specified is ASCII.
ASCII table 1:
0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000102030405060708090A0B0C0D0E0F'x
10 '101112131415161718191A1B1C1D1E1F'x
20 '202122232425262728292A2B2C2D2E2F'x
30 '303132333435363738393A3B3C3D3E3F'x
40 '404142434445464748494A4B4C4D4E4F'x
50 '505152535455565758595A5B5C5D5E5F'x
60 '606162636465666768696A6B6C6D6E6F'x ←
70 '707172737475767778797A7B7C7D7E7F'x ←
80 '808182838485868788898A8B8C8D8E8F'x
90 '909192939495969798999A9B9C9D9E9F'x
A0 'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'x
B0 'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'x
C0 'C0C1C2C3C4C5C6C7C8C9CACCCDCCECF'x
D0 'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0 'E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEF'x
F0 'F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF'x
```

プログラム 2: テーブルの編集

```
replace 'a' 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';

save table=upper;

load table=upper;

list one;
```

プログラムの説明

変換テーブル内の文字を指定された開始位置から置き換えます。この REPLACE ステートメントでは、最初に出現する 16 進の "a" (61) を検索し、その文字と続く 25 個の 16 進値を大文字の "A" から "Z" までの 16 進値で置き換えます。

```
replace 'a' 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';
```

変更を保存します。この SAVE ステートメントでは、ASCII 変換テーブルに加えた変更を UPPER という新しいテーブル名で保存します。ASCII 変換テーブルに保存された内容は、変更されずに残ります。

```
save table=upper;
```

変換テーブルを読み込んで表示します。この LOAD ステートメントでは、編集された変換テーブル UPPER を読み込みます。この LIST ステートメントでは、SAS ログに変換テーブル UPPER を表示します

```
load table=upper;

list one;
```

SAS ログ

```
NOTE: Table UPPER
being loaded.
UPPER table 1:
0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000102030405060708090A0B0C0D0E0F'x
10 '101112131415161718191A1B1C1D1E1F'x
20 '202122232425262728292A2B2C2D2E2F'x
30 '303132333435363738393A3B3C3D3E3F'x
40 '404142434445464748494A4B4C4D4E4F'x
50 '505152535455565758595A5B5C5D5E5F'x
60 '604142434445464748494A4B4C4D4E4F'x ←
70 '505152535455565758595A7B7C7D7E7F'x ←
80 '808182838485868788898A8B8C8D8E8F'x
90 '909192939495969798999A9B9C9D9E9F'x
A0 'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'x
B0 'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'x
C0 'C0C1C2C3C4C5C6C7C8C9CACBCCDCECF'x
D0 'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0 'E0E1E2E3E4E5E6E7E8E9EABEBECEDEEFF'x
F0 'F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF'x
```

例 5: 逆変換テーブルの作成

要素: INVERSE ステートメント
LIST ステートメント
SAVE ステートメント

この例では、“例 4: 開始位置に引用符付き文字を使用した編集” (537 ページ) で作成した変換テーブルの逆変換テーブルを作成します。この例で作成する新しい変換テーブルは、データ通信に使用する動作環境-デバイス間変換のテーブルです。

```
options nodate pageno=1 linesize=80 pagesize=60;
proc trantab table=upper;
```

逆変換テーブルを作成し、各テーブルを保存して表示します。この INVERSE ステートメントでは、元のテーブル 1 (UPPER) を逆にしてテーブル 2 を作成します。この SAVE ステートメントでは、変換テーブルを保存します。この LIST BOTH ステートメントでは、元の変換テーブルと逆変換テーブルの両方を表示します。

```
inverse;
save;
list both;
```

SAS ログ

INVERSE ステートメントでは、変換テーブルを逆転させるときに検出した複数の変換をすべて SAS ログに表示します。“例 4: 開始位置に引用符付き文字を使用した編集” (537 ページ) では、すべての小文字が変換テーブル UPPER で大文字に変換されました。つまり、UPPER には 2 組の大文字があります。INVERSE で変換できない場合、PROC TRANTAB は値を 00 で埋めます。変換テーブル UPPER の逆変換テーブルには、00 の値が数多く含まれていることに注意してください。

SAS ログには、テーブル 1 の逆変換テーブルを作成するときに検出された重複値がすべて表示されます。スペースを節約するために、この例ではこのようなメッセージの多くは削除されています。

```
NOTE: This table cannot be mapped one to one.
duplicate of '41'x found at '61'x in table one.
duplicate of '42'x found at '62'x in table one.
duplicate of '43'x found at '63'x in table one.
.
.
.
duplicate of '58'x found at '78'x in table one.
duplicate of '59'x found at '79'x in table one.
duplicate of '5A'x found at '7A'x in table one.
NOTE: Saving table UPPER.
UPPER table 1:
0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000102030405060708090A0B0C0D0E0F'x
10 '101112131415161718191A1B1C1D1E1F'x
20 '202122232425262728292A2B2C2D2E2F'x
30 '303132333435363738393A3B3C3D3E3F'x
40 '404142434445464748494A4B4C4D4E4F'x
50 '505152535455565758595A5B5C5D5E5F'x
60 '604142434445464748494A4B4C4D4E4F'x
70 '505152535455565758595A7B7C7D7E7F'x
80 '808182838485868788898A8B8C8D8E8F'x
90 '909192939495969798999A9B9C9D9E9F'x
A0 'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'x
B0 'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'x
C0 'C0C1C2C3C4C5C6C7C8C9CACBCCDCECF'x
D0 'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0 'E0E1E2E3E4E5E6E7E8E9EABECEDEEEF'x
F0 'F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF'x

UPPER table 2:
0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000102030405060708090A0B0C0D0E0F'x
10 '101112131415161718191A1B1C1D1E1F'x
20 '202122232425262728292A2B2C2D2E2F'x
30 '303132333435363738393A3B3C3D3E3F'x
40 '404142434445464748494A4B4C4D4E4F'x
50 '505152535455565758595A5B5C5D5E5F'x
60 '600000000000000000000000000000'x
70 '0000000000000000000000007B7C7D7E7F'x
80 '808182838485868788898A8B8C8D8E8F'x
90 '909192939495969798999A9B9C9D9E9F'x
A0 'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'x
B0 'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'x
C0 'C0C1C2C3C4C5C6C7C8C9CACBCCDCECF'x
D0 'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0 'E0E1E2E3E4E5E6E7E8E9EABECEDEEEF'x
F0 'F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF'x
```

例 6: 並べ替えへのさまざまな変換テーブルの使用

要素: PROC SORT ステートメントオプション: SORTSEQ=
PRINT プロシジャ

この例では、デフォルトの並べ替え順序とは異なる順序でデータを並べ替えるために、異なる変換テーブルを指定する方法を示します。英語(U.S.)以外の言語で記述された文字は、デフォルトの順序とは異なる並べ替え順序が必要になる場合があります。

SAS サンプルライブラリにある TRABASE プログラムを使用すると、複数言語用の変換テーブルを作成できます。例はすべて UNIX 環境で作成されました。

SAS システムオプションを設定します。

```
options nodate pageno=1 linesize=80 pagesize=60;
```

TESTSORT データセットを作成します。 DATA ステップで、4 組の単語のペアを使用して SAS データセットを作成します。各ペアの単語は最初の文字が大文字か小文字かという点のみが異なります。

```
data testsort;
input Values $10.;
datalines;
Always
always
Forever
forever
Later
later
Yesterday
yesterday
;
```

デフォルトの並べ替え順序とは異なる順序でデータを並べ替えます。 PROC SORT では、デフォルトの変換テーブルを使用して、すべての小文字、すべての大文字の順序でデータを並べ替えます。

```
proc sort;
by values;
run;
```

データセットを出力します。 PROC PRINT では、並べ替えたデータセットが出力されません。

```
proc print noobs;
title 'Default Sort Sequence';
run;
```

SAS 出力

次の出力はデフォルトの変換テーブルを使用した値の並べ替えの出力です。デフォルトの並べ替え順序を使用すると、最初にアルファベット順で大文字の単語すべて、次に小文字の単語すべてを並べます。

```
Default Sort Sequence
1

Values

Always
Forever
Later
Yesterday
always
forever
later
yesterday
```

変換テーブル UPPER に従ってデータを並べ替え、新しいデータセットを出力します。SORTSEQ=オプションでは、PROC SORT が変換テーブル UPPER に従ってデータを並べ替えるように指定しています。変換テーブル UPPER は、小文字と大文字を区別せずに処理するようにカスタマイズされています。この方法は、大文字か小文字かに関係なく並べ替えるときに便利です。PROC PRINT では、並べ替えたデータセットが出力されます。

```
proc sort sortseq=upper;
by values;
run;
proc print noobs;
title 'Customized Sort Sequence';
run;
```

SAS 出力

次の出力はカスタマイズされた変換テーブルを使用した値の並べ替えの結果です。このカスタマイズされた並べ替え順序では、最初の文字が大文字か小文字かに関係なく、すべての単語をアルファベット順に並べ替えます。

```
Customized Sort Sequence 2

Values

Always
always
Forever
forever
Later
later
Yesterday
yesterday
```

例 7: テーブル 1 とテーブル 2 の編集

要素: LIST ステートメント

REPLACE ステートメント
SAVE ステートメント
SWAP ステートメント

この例では、変換テーブルの両方の領域を編集する方法を示します。位置 256 から 511 (テーブル 2)までを編集するには、次の操作を実行する必要があります。

- SWAP ステートメントを発行して、テーブル 2 とテーブル 1 を交換します。
- 適切な REPLACE ステートメントを発行して、テーブル 2 を変更します。
- 再び SWAP ステートメントを発行して、テーブルを再配置します。

SAS ログ内の矢印によって、変更された列と行が示されます。

SAS システムオプションを設定し、変換テーブルを指定します。

```
options nodate pageno=1 linesize=80 pagesize=60;  
proc trantab table=upper;
```

元の変換テーブルを表示します。この LIST ステートメントでは、元の UPPER 変換テーブルを表示します。

```
list both;
```

SAS ログ

次の出力は、元の UPPER 変換テーブルです。

```
NOTE: Table specified is UPPER.
UPPER table 1:
↓
0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000102030405060708090A0B0C0D0E0F'x ←
10 '101112131415161718191A1B1C1D1E1F'x
20 '202122232425262728292A2B2C2D2E2F'x
30 '303132333435363738393A3B3C3D3E3F'x
40 '404142434445464748494A4B4C4D4E4F'x
50 '505152535455565758595A5B5C5D5E5F'x
60 '604142434445464748494A4B4C4D4E4F'x
70 '505152535455565758595A7B7C7D7E7F'x
80 '808182838485868788898A8B8C8D8E8F'x
90 '909192939495969798999A9B9C9D9E9F'x
A0 'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'x
B0 'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'x
C0 'C0C1C2C3C4C5C6C7C8C9CACBCCDCECF'x
D0 'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0 'E0E1E2E3E4E5E6E7E8E9EABEBECEDEEEF'x
F0 'F0F1F2F3F4F5F6F7F8F9FABFBCFDFEFF'x

UPPER table 2:
↓
0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000102030405060708090A0B0C0D0E0F'x ←
10 '101112131415161718191A1B1C1D1E1F'x
20 '202122232425262728292A2B2C2D2E2F'x
30 '303132333435363738393A3B3C3D3E3F'x
40 '404142434445464748494A4B4C4D4E4F'x
50 '505152535455565758595A5B5C5D5E5F'x
60 '600000000000000000000000000000'x
70 '0000000000000000000000007B7C7D7E7F'x
80 '808182838485868788898A8B8C8D8E8F'x
90 '909192939495969798999A9B9C9D9E9F'x
A0 'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'x
B0 'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'x
C0 'C0C1C2C3C4C5C6C7C8C9CACBCCDCECF'x
D0 'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0 'E0E1E2E3E4E5E6E7E8E9EABEBECEDEEEF'x
F0 'F0F1F2F3F4F5F6F7F8F9FABFBCFDFEFF'x
```

変換テーブル内の文字を指定された開始位置から置き換えます。REPLACE ステートメントは開始位置 1 から開始し、01 の現在の値を'0A'で置き換えます。

```
replace 1 '0A'x;
```

テーブル 2 を編集用に準備します。最初の SWAP ステートメントでは、テーブル 2 を編集できるように配置します。2 番目の REPLACE ステートメントでは、テーブル 1 と同じ変更をテーブル 2 に加えます。

```
swap;
replace 1 '0A'x;
```

元の位置にテーブルを保存して表示します。2 番目の SWAP ステートメントでは、テーブル 1 とテーブル 2 を元の位置に戻します。SAVE ステートメントでは、デフォルトで変換テーブルの両方の領域を保存します。LIST ステートメントでは、テーブルの両方の領域を表示します。

```

swap;
save;
list both;

```

SAS ログ

編集された UPPER 変換テーブルの両方の領域の 2 バイト目で、16 進値 01 が 16 進値 0A に置き換えられています。この変換が行われたテーブル内の行と列が矢印で示されています。

```

NOTE: Table specified is UPPER.
UPPER table 1:
[darr]
  0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000A02030405060708090A0B0C0D0E0F'x <--
10 '101112131415161718191A1B1C1D1E1F'x
20 '202122232425262728292A2B2C2D2E2F'x
30 '303132333435363738393A3B3C3D3E3F'x
40 '404142434445464748494A4B4C4D4E4F'x
50 '505152535455565758595A5B5C5D5E5F'x
60 '604142434445464748494A4B4C4D4E4F'x
70 '505152535455565758595A7B7C7D7E7F'x
80 '808182838485868788898A8B8C8D8E8F'x
90 '909192939495969798999A9B9C9D9E9F'x
A0 'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'x
B0 'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'x
C0 'C0C1C2C3C4C5C6C7C8C9CACBCCDCECF'x
D0 'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0 'E0E1E2E3E4E5E6E7E8E9EABECEDEEEF'x
F0 'F0F1F2F3F4F5F6F7F8F9FABFBCFDFEFF'x

UPPER table 2:
[darr]
  0 1 2 3 4 5 6 7 8 9 A B C D E F
00 '000A02030405060708090A0B0C0D0E0F'x <--
10 '101112131415161718191A1B1C1D1E1F'x
20 '202122232425262728292A2B2C2D2E2F'x
30 '303132333435363738393A3B3C3D3E3F'x
40 '404142434445464748494A4B4C4D4E4F'x
50 '505152535455565758595A5B5C5D5E5F'x
60 '600000000000000000000000000000'x
70 '0000000000000000000000007B7C7D7E7F'x
80 '808182838485868788898A8B8C8D8E8F'x
90 '909192939495969798999A9B9C9D9E9F'x
A0 'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'x
B0 'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'x
C0 'C0C1C2C3C4C5C6C7C8C9CACBCCDCECF'x
D0 'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'x
E0 'E0E1E2E3E4E5E6E7E8E9EABECEDEEEF'x
F0 'F0F1F2F3F4F5F6F7F8F9FABFBCFDFEFF'x

```


11 部

ロケール、エンコーディング、トランスコーディングの値

18 章	
LOCALE=システムオプションの値	549
19 章	
DBCS データ処理に関連する SAS システムオプション	561
20 章	
SAS 言語要素中のエンコーディング値	563
21 章	
SAS セッションのエンコーディング値	573

18 章

LOCALE=システムオプションの値

ENCODING、PAPERSIZE、DFLANG、DATESTYLE オプションの LOCALE=値とデフォルト設定	549
---	-----

ENCODING、PAPERSIZE、DFLANG、DATESTYLE オプションの LOCALE=値とデフォルト設定

次の表に、SAS 名または Posix 名を使用して指定した有効な LOCALE=値を表示します。エイリアス名も示されています。エイリアス名のないロケールもあります。

表 18.1 LOCALE=システムオプションの値

SAS 名	Posix ロケール	エイリアス
Afrikaans_SouthAfrica	af_ZA	Afrikaans af
Albanian_Albania	sq_AL	Albanian sq
Arabic_Algeria	ar_DZ	
Arabic_Bahrain	ar_BH	
Arabic_Egypt	ar_EG	
Arabic_India	ar_IN	
Arabic_Iraq	ar_IQ	
Arabic_Jordan	ar_JO	
Arabic_Kuwait	ar_KW	
Arabic_Lebanon	ar_LB	
Arabic_Libya	ar_LY	
Arabic_Morocco	ar_MA	

SAS 名	Posix ロケール	エイリアス
Arabic_Oman	ar_OM	
Arabic_Qatar	ar_QA	
Arabic_SaudiArabia	ar_SA	
Arabic_Sudan	ar_SD	
Arabic_Syria	ar_SY	
Arabic_Tunisia	ar_TN	
Arabic_UnitedArabEmirates	ar_AE	Arabic ar
Arabic_Yemen	ar_YE	
Bengali_India	bn_IN	Bengali bn
Bosnian_BosniaHerzegovina	bs_BA	Bosnian bs
Bulgarian_Bulgaria	bg_BG	Bulgarian bg
Byelorussian_Belarus	be_BY	Byelorussian Belarusian Byelorussian_Belarus be
Catalan_Spain	ca_ES	Catalan ca
Chinese_China	zh_CN	Chinese zh
Chinese_HongKong	zh_HK	
Chinese_Macau	zh_MO	
Chinese_Singapore	zh_SG	
Chinese_Taiwan	zh_TW	
Cornish_UnitedKingdom	kw_GB	Cornish kw
Croatian_BosniaHerzegovina	hr_BA	
Croatian_Croatia	hr_HR	Croatian hr

SAS 名	Posix ロケール	エイリアス
Czech_CzechRepublic	cs_CZ	Czech cs
Danish_Denmark	da_DK	Danish da
Dutch_Belgium	nl_BE	
Dutch_Netherlands	nl_NL	Dutch nl
English_Australia	en_AU	
English_Belgium	en_BE	
English_Botswana	en_BW	
English_Canada	en_CA	
English_Caribbean	en_CB	
English_HongKong	en_HK	
English_India	en_IN	
English_Ireland	en_IE	
English_Jamaica	en_JM	
English_NewZealand	en_NZ	
English_Philippines	en_PH	
English_Singapore	en_SG	
English_SouthAfrica	en_ZA	
English_UnitedKingdom	en_GB	
English_UnitedStates	en_US	English ja
English_Zimbabwe	en_ZW	
Estonian_Estonia	et_EE	Estonian et
Faroese_FaroeIslands	fo_FO	Faroese fo
Finnish_Finland	fi_FI	Finnish fi
French_Belgium	fr_BE	

SAS 名	Posix ロケール	エイリアス
French_Canada	fr_CA	
French_France	fr_FR	French fr
French_Luxembourg	fr_LU	
French_Switzerland	fr_CH	
German_Austria	de_AT	
German_Germany	de_DE	German de
German_Liechtenstein	de_LI	
German_Luxembourg	de_LU	
German_Switzerland	de_CH	
Greek_Greece	el_GR	Greek el
Greenlandic_Greenland	kl_GL	Greenlandic kl
Hebrew_Israel	he_IL	Hebrew he
Hindi_India	hi_IN	Hindi hi
Hungarian_Hungary	hu_HU	Hungarian hu
Icelandic_Iceland	is_IS	Icelandic is
Indonesian_Indonesia	id_ID	Indonesian id
Italian_Italy	it_IT	Italian it
Italian_Switzerland	it_CH	
Japanese_Japan	ja_JP	Japanese ja
Korean_Korea	ko_KR	Korean ko

SAS 名	Posix ロケール	エイリアス
Latvian_Latvia	lv_LV	Latvian lv
Lithuanian_Lithuania	lt_LT	Lithuanian lt
Macedonian_Macedonia	mk_MK	Macedonian mk
Malay_Malaysia	ms_MY	Malay ms
Maltese_Malta	mt_MT	Maltese mt
ManxGaelic_UnitedKingdom	gv_GB	ManxGaelic gv
Marathi_India	mr_IN	Marathi mr
NorwegianBokmal_Norway	nb_NO	NorwegianBokmal nb
NorwegianNynorsk_Norway	nn_NO	NorwegianNynorsk nn
Norwegian_Norway	no_NO	Norwegian no
Persian_India	fa_IN	
Persian_Iran	fa_IR	Persian fa
Polish_Poland	pl_PL	Polish pl
Portuguese_Brazil	pt_BR	
Portuguese_Portugal	pt_PT	Portuguese pt
Romanian_Romania	ro_RO	Romanian ro
Russian_Russia	ru_RU	Russian ru
Russian_Ukraine	ru_UA	

SAS 名	Posix ロケール	エイリアス
Serbian_BosniaHerzegovina	sr_BA	
Serbian_Montenegro	sr_ME	
Serbian_Serbia	sr_RS	Serbian sr
Serbian_Yugoslavia	sr_YU	
SerbianLatin_BosniaHerzegovina	sh_BA	
SerbianLatin_Montenegro	sh_ME	
SerbianLatin_Serbia	sh_RS	SerbianLatin sh
Slovak_Slovakia	sk_SK	Slovak Slovakian Slovakian_Slovakia sk
Slovenian_Slovenia	sl_SI	Slovenian sl
Spanish_Argentina	es_AR	
Spanish_Bolivia	es_BO	
Spanish_Chile	es_CL	
Spanish_Colombia	es_CO	
Spanish_CostaRica	es_CR	
Spanish_DominicanRepublic	es_DO	
Spanish_Ecuador	es_EC	
Spanish_ElSalvador	es_SV	
Spanish_Guatemala	es_GT	
Spanish_Honduras	es_HN	
Spanish_Mexico	es_MX	
Spanish_Nicaragua	es_NI	
Spanish_Panama	es_PA	
Spanish_Paraguay	es_PY	
Spanish_Peru	es_PE	

SAS 名	Posix ロケール	エイリアス
Spanish_PuertoRico	es_PR	
Spanish_Spain	es_ES	Spanish es
Spanish_UnitedStates	es_US	
Spanish_Uruguay	es_UY	
Spanish_Venezuela	es_VE	
Swedish_Sweden	sv_SE	Swedish sv
Tamil_India	ta_IN	Tamil ta
Telugu_India	te_IN	Telugu te
Thai_Thailand	th_TH	Thai th
Turkish_Turkey	tr_TR	Turkish tr
Ukrainian_Ukraine	uk_UA	Ukrainian uk
Vietnamese_Vietnam	vi_VN	Vietnamese vi

次の表に動作環境による ENCODING=オプションの有効な Posix 値とデフォルト設定を表示します。DFLANG、DATESTYLE、PAPERSIZE システムオプションは、自動的に設定されます。

次に、例を示します。

```
sas9 -locale arabic_algeria
```

Arabic_Algeria の LOCALE=値を指定した場合の、対応するシステムオプションのデフォルト設定を次に示します。

```
DFLANG=English
DATESTYLE=DMY
PAPERSIZE=A4
```

表 18.2 LOCALE=システムオプションに基づいた ENCODING、DFLANG、DATESTYLE、PAPERSIZE システムオプションのデフォルト値

Posix ロケール	Windows エンコーディング	UNIX エンコーディング	z/OS エンコーディング	DFLANG=	DATESTYLE=	PAPERSIZE=
af_ZA	wlatin1	latin1	open_ed-1047	English	YMD	A4
ar_AE	warabic	arabic	open_ed-425	English	DMY	A4
ar_BH	warabic	arabic	open_ed-425	English	DMY	A4
ar_DZ	warabic	arabic	open_ed-425	English	DMY	A4
ar_EG	warabic	arabic	open_ed-425	English	DMY	A4
ar_IN	warabic	arabic	open_ed-425	English	DMY	A4
ar_IQ	warabic	arabic	open_ed-425	English	DMY	A4
ar_JO	warabic	arabic	open_ed-425	English	DMY	A4
ar_KW	warabic	arabic	open_ed-425	English	DMY	A4
ar_LB	warabic	arabic	open_ed-425	English	DMY	A4
ar_LY	warabic	arabic	open_ed-425	English	DMY	A4
ar_MA	warabic	arabic	open_ed-425	English	DMY	A4
ar_OM	warabic	arabic	open_ed-425	English	DMY	A4
ar_QA	warabic	arabic	open_ed-425	English	DMY	A4
ar_SA	warabic	arabic	open_ed-425	English	DMY	A4
ar_SD	warabic	arabic	open_ed-425	English	DMY	A4
ar_SY	warabic	arabic	open_ed-425	English	DMY	A4
ar_TN	warabic	arabic	open_ed-425	English	DMY	A4
ar_YE	warabic	arabic	open_ed-425	English	DMY	A4
be_BY	wcyrillic	cyrillic	open_ed-1025	English	DMY	A4
bg_BG	wcyrillic	cyrillic	open_ed-1025	English	YMD	A4
bn_IN	wlatin1	latin1	open_ed-1047	English	DMY	A4
ca_ES	wlatin1	latin1	open_ed-1148	English	DMY	A4
cs_CZ	wlatin2	latin2	open_ed-870	Czech	DMY	A4
da_DK	wlatin1	latin9	open_ed-1142	Danish	DMY	A4
de_AT	wlatin1	latin9	open_ed-1141	German	DMY	A4

Posix ロケール	Windows エンコーディング	UNIX エンコーディング	z/OS エンコーディング	DFLANG=	DATESTYLE=	PAPERSIZE=
de_CH	wlatin1	latin9	open_ed-1148	Swiss_German	DMY	A4
de_DE	wlatin1	latin9	open_ed-1141	German	DMY	A4
de_LI	wlatin1	latin9	open_ed-1141	German	DMY	A4
de_LU	wlatin1	latin9	open_ed-1141	German	DMY	A4
el_GR	wgreek	greek	open_ed-875	English	DMY	A4
en_AU	wlatin1	latin1	open_ed-1047	English	DMY	A4
en_BE	wlatin1	latin9	open_ed-1148	English	DMY	A4
en_BW	wlatin1	latin1	open_ed-1047	English	DMY	A4
en_CA	wlatin1	latin1	open_ed-1047	English	DMY	letter
en_CB	wlatin	latin1	open_ed-1047	English	MDY	letter
en_GB	wlatin1	latin9	open_ed-1146	English	DMY	A4
en_HK	wlatin1	latin9	open_ed-1146	English	DMY	A4
en_IE	wlatin1	latin9	open_ed-1146	English	DMY	A4
en_IN	wlatin1	latin9	open_ed-1146	English	DMY	A4
en_JM	wlatin1	latin1	open_ed-1047	English	DMY	letter
en_NZ	wlatin1	latin1	open_ed-1047	English	DMY	A4
en_PH	wlatin1	latin1	open_ed-1047	English	MDY	A4
en_SG	wlatin1	latin9	open_ed-1146	English	DMY	A4
en_US	wlatin1	latin1	open_ed-1047	English	MDY	A4
en_ZA	wlatin1	latin1	open_ed-1047	English	DMY	A4
en_ZW	wlatin1	latin1	open_ed-1047	English	DMY	A4
es_AR	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_BO	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_CL	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_CO	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_CR	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_DO	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter

Posix ロケール	Windows エンコーディング	UNIX エンコーディング	z/OS エンコーディング	DFLANG=	DATESTYLE=	PAPER SIZE=
es_EC	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_ES	wlatin1	latin9	open_ed-1145	Spanish	DMY	A4
es_GT	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_HN	wlatin1	latin1	open_ed-1047	Spanish	MDY	letter
es_MX	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_NI	wlatin1	latin1	open_ed-1047	Spanish	MDY	letter
es_PA	wlatin1	latin1	open_ed-1047	Spanish	MDY	letter
es_PE	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_PR	wlatin1	latin1	open_ed-1047	Spanish	MDY	letter
es_PY	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_SV	wlatin1	latin1	open_ed-1047	Spanish	MDY	letter
es_US	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
es_UY	wlatin1	latin1	open_ed-1047	Spanish	DMY	A4
es_VE	wlatin1	latin1	open_ed-1047	Spanish	DMY	letter
et_EE	wbaltic	latin6	open_ed-1122	English	DMY	A4
fa_IN	warabic	arabic	open_ed-1097	English	YMD	A4
fa_IR	warabic	arabic	open_ed-1097	English	YMD	A4
fi_FI	wlatin1	latin9	open_ed-1143	Finnish	DMY	A4
fo_FO	wlatin1	latin1	open_ed-1047	English	DMY	A4
fr_BE	wlatin1	latin9	open_ed-1148	French	DMY	A4
fr_CA	wlatin1	latin1	open_ed-1047	French	DMY	letter
fr_CH	wlatin1	latin9	open_ed-1148	Swiss_French	DMY	A4
fr_FR	wlatin1	latin9	open_ed-1147	French	DMY	A4
fr-LU	wlatin1	latin9	open_ed-1147	French	DMY	A4
gv_GB	wlatin1	latin8	open_ed-1148	English	DMY	A4
he_IL	whebrew	hebrew	open_ed-424	English	DMY	A4
hi_IN	pcscii806	latin1	open_ed-1137	English	DMY	A4
hr_HR	wlatin2	latin2	open_ed-870	Croatian	YMD	A4

Posix ロケール	Windows エンコーディング	UNIX エンコーディング	z/OS エンコーディング	DFLANG=	DATESTYLE=	PAPERSIZE=
hu_HU	wlatin2	latin2	open_ed-870	Hungarian	YMD	A4
is_IS	wlatin1	latin1	open_ed-1047	English	DMY	A4
id_ID	wlatin1	latin1	open_ed-1047	English	DMY	A4
it_CH	wlatin1	latin9	open_ed-1148	Italian	DMY	A4
it_IT	wlatin1	latin9	open_ed-1144	Italian	DMY	A4
ja_JP	shift-jis	euc-jp, shift-jis(*)	ibm-939	Japanese	YMD	A4
kl_GL	wlatin1	latin1	open_ed-1047	English	DMY	A4
ko_KR	euc-kr	euc-kr	ibm-933	Locale	YMD	A4
kw_GB	wlatin1	latin1	open_ed-1148	English	DMY	A4
lt_LT	wbaltic	latin6	open_ed-1112	English	YMD	A4
lv_LV	wbaltic	latin6	open_ed-1112	English	YMD	A4
mk_MK	wcyrillic	cyrillic	open_ed-1154	English	DMY	A4
mr_IN	pcscii806	latin1	open_ed-1137	English	DMY	A4
ms_MY	wlatin1	latin1	open_ed-1047	English	DMY	A4
mt_MT	wlatin1	latin3	open_ed-905	English	DMY	A4
nb_NO	wlatin1	latin9	open_ed-1142	Norwegian	DMY	A4
nl_BE	wlatin1	latin1	open_ed-1148	Dutch	DMY	A4
nl_NL	wlatin1	latin1	open_ed-1140	Dutch	DMY	A4
nn_NO	wlatin1	latin9	open_ed-1142	Norwegian	DMY	A4
no_NO	wlatin1	latin9	open_ed-1142	Norwegian	DMY	A4
pl_PL	wlatin2	latin2	open_ed-870	Polish	YMD	A4
pt_BR	wlatin1	latin1	open_ed-275	Portuguese	DMY	letter
pt_PT	wlatin1	latin1	open_ed-1140	Portuguese	DMY	A4
ro_RO	wlatin2	latin2	open_ed-870	English	DMY	A4
ru_RU	wcyrillic	cyrillic	open_ed-1025	Russian	DMY	A4
ru_UA	wcyrillic	cyrillic	open_ed-1154	Russian	DMY	A4
sh_YU	wlatin2	latin2	open_ed-870	English	DMY	A4
sk_SK	wlatin2	latin2	open_ed-870	English	DMY	A4

Posix ロケール	Windows エンコーディング	UNIX エンコーディング	z/OS エンコーディング	DFLANG=	DATESTYLE=	PAPER SIZE=
sl_SL	wlatin2	latin2	open_ed-870	Slovenian	YMD	A4
sr_YU	wcyrillic	cyrillic	open_ed-1025	English	DMY	A4
sq_AL	wlatin2	latin2	open_ed-1153	English	YMD	A4
sv_SE	wlatin1	latin9	open_ed-1143	Swedish	YMD	A4
ta_IN	wlatin1	latin1	open_ed-1047	English	DMY	A4
te_IN	wlatin1	latin1	open_ed-1047	English	DMY	A4
th_TH	pcoem874	thai	open_ed-1160	English	DMY	A4
tr_TR	wturkish	latin5	open_ed-1026	English	DMY	A4
uk_UA	wcyrillic	cyrillic	open_ed-1025	English	DMY	A4
vi_VN	wvietnamese	latin1	open_ed-1164	English	DMY	A4
zh_CN	euc-cn	euc-cn	ibm-935	Locale	YMD	A4
zh_HK	ms-950	euc-tw, ms-950 (*)	ibm-937	Locale	YMD	A4
zh_MO	ms-950	euc-tw, mis-950 (*)	ibm-937	Locale	YMD	A4
zh_SG	euc-cn	euc-cn	ibm-935	Locale	DMY	A4
zh_TW	ms-950	euc-tw, ms-950 (*)	ibm-937	Locale	YMD	A4

1 (*) プラットフォームに依存

19 章

DBCS データ処理に関連する SAS システムオプション

DBCS の SAS セッションで使用されるシステムオプションの概要	561
SAS セッションの DBCS 値	561

DBCS の SAS セッションで使用されるシステムオプションの概要

SAS セッションの DBCS エンコーディング値を指定するには、DBCSSLANG=と DBCSTYPE=システムオプションを使用します。DBCS を使用している場合、ENCODING=システムオプションは直接使用しません。

SAS セッションの DBCS 値

次の表に、z/OS、UNIX および Windows 動作環境でサポートされる DBCSSLANG=および DBCSTYPE=システムオプションの値を示します。

注: エンコーディング値にハイフン(-)が含まれる場合は、エンコーディング値を引用符で囲んでください。

表 19.1 DBCSSLANG=および DBCSTYPE=システムオプションでの DBCS サポート値

DBCSSLANG=	z/OS DBCSTYPE=	UNIX DBCSTYPE=	Windows DBCSTYPE=
Chinese	ibm	dec	pcms
Chinese	n/a	hp15	n/a
Chinese	n/a	euc	n/a
Chinese	n/a	pcms	n/a
Japanese	ibm	dec	pcms
Japanese	pcibm	euc	SJIS
Japanese	n/a	hp15	n/a

DBCSSLANG=	z/OS DBCSTYPE=	UNIX DBCSTYPE=	Windows DBCSTYPE=
Japanese	n/a	SJIS	n/a
Korean	ibm	pcibm	pcms
Korean	n/a	pcms	n/a
Korean	n/a	dec	n/a
Korean	n/a	euc	n/a
Korean	n/a	hp15	n/a
Taiwanese	ibm	dec	pcms
Taiwanese	pcibm	euc	n/a
Taiwanese	n/a	hp15	n/a
Taiwanese	n/a	pcms	n/a

20 章

SAS 言語要素中のエンコーディング
値

エンコーディング値を使用する SAS 言語要素の概要	563
データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値	563

エンコーディング値を使用する SAS 言語要素の概要

SAS セッションのエンコーディングが SAS ファイルのエンコーディングまたは SAS ファイル内に存在するデータと異なる場合、トランスコーディングが必要になります。SAS ファイルが Western Latin1 エンコーディングで作成され、ドイツ語 EBCDIC エンコーディングを使用する IBM メインフレームに移動されたとします。IBM メインフレームがそのファイルに正常にアクセスできるように、SAS データファイルを Western Latin1 エンコーディングからドイツ語 EBCDIC エンコーディングにトランスコードする必要があります。トランスコーディングオプションを含む SAS 言語要素などの、トランスコーディングの概念の詳細については、“[各国語サポート関連のトランスコーディング](#)” (27 ページ) を参照してください。

データのトランスコーディングに使用する SBCS、DBCS
および Unicode エンコーディング値

次の表は、各動作環境でデータのトランスコーディングに使用される SBCS、DBCS および Unicode エンコーディング値のリストです。エンコーディング値は、トランスコーディングのオプションを含む SAS 言語要素で有効です。

注: エンコーディング値にハイフン(-)が含まれる場合は、エンコーディング値を引用符で囲んでください。

表 20.1 データのトランスコーディングに使用する SBCS、DBCS および Unicode エンコーディング値

エンコーディング名	短縮名	説明	1 バイトの最大文字数
aarabic	aara	アラビア語 (Macintosh)	1

エンコーディング名	短縮名	説明	1 バイトの最大文字数
agreek	agrk	ギリシャ語 (Macintosh)	1
ahebrew	aheb	ヘブライ語 (Macintosh)	1
aiceland	aice	アイスランド語 (Macintosh)	1
any	anye	トランスコーディング 指定無し	1
arabic	arab	アラビア語(ISO)	1
aroman	arom	Roman (Macintosh)	1
athai	atha	MacOS 21-Thai	1
aturkish	atur	トルコ語(Macintosh)	1
aukrainian	aukr	ウクライナ語 (Macintosh)	1
big5	big5	繁体字中国語(Big5)	2
cyrillic	cyrl	キリル語(ISO)	1
dec-cn	zvms	簡体字中国語(DEC)	4
dec-jp	jvms	日本語(DEC)	2
dec-tw	yvms	繁体字中国語(DEC)	4
ebcdic037	e037	北米諸語(EBCDIC)	1
ebcdic275	e275	ブラジル(EBCDIC)	1
ebcdic424	e424	ヘブライ語(EBCDIC)	1
ebcdic425	e425	アラビア語(EBCDIC)	1
ebcdic500	e500	国際化対応 (EBCDIC)	1
ebcdic838	e838	タイ語(EBCDIC)	1
ebcdic870	e870	中央ヨーロッパ諸語 (EBCDIC)	1
ebcdic875	e875	ギリシャ語(EBCDIC)	1
ebcdic905	e905	ラテン諸語 3 EBCDIC	1
ebcdic924	e924	ヨーロッパ諸語 (EBCDIC)	1

エンコーディング名	短縮名	説明	1 バイトの最大文字数
ebcdic1025	ecyr	キリル語(EBCDIC)	1
ebcdic1026	etur	トルコ語(EBCDIC)	1
ebcdic1047	elat	西欧(EBCDIC)	1
ebcdic1097	e097	ペルシア諸語 (EBCDIC)	1
ebcdic1112	ebal	バルト語(EBCDIC)	1
ebcdic1122	eest	エストニア語 (EBCDIC)	1
ebcdic1130	evie	ベトナム語 (EBCDIC)	1
ebcdic1137	e137	デーバナーガリー文 字(EBCDIC)	1
ebcdic1140	e140	北米諸語(EBCDIC)	1
ebcdic1141	e141	オーストリア/ドイツ (EBCDIC)	1
ebcdic1142	e142	デンマーク/ノルウェ ー(EBCDIC)	1
ebcdic1143	e143	フィンランド/スウェー デン(EBCDIC)	1
ebcdic1144	e144	イタリア(EBCDIC)	1
ebcdic1145	e145	スペイン(EBCDIC)	1
ebcdic1146	e146	英国(EBCDIC)	1
ebcdic1147	e147	フランス(EBCDIC)	1
ebcdic1148	e148	国際化対応 (EBCDIC)	1
ebcdic1149	e149	アイスランド語 (EBCDIC)	1
ebcdic1153	e153	Latin 2 Euro EBCDIC	1
ebcdic1154	e154	Cyrillic Euro EBCDIC	1
ebcdic1155	e155	Turkey Euro EBCDIC	1
ebcdic1156	e156	Baltic Euro EBCDIC	1

エンコーディング名	短縮名	説明	1 バイトの最大文字数
ebcdic1157	e157	Estonia Euro EBCDIC	1
ebcdic1158	e158	Cryillic Ukraine Euro EBCDIC	1
ebcdic1160	e160	cp1160 EBCDIC	1
ebcdic1164	e164	cp1164 EBCDIC	1
ebcdicany	eany	すべての EBCDIC エンコーディングと互換性のあるデータセットを作成可能	1
euc-cn	zeuc	簡体字中国語(EUC)	2
euc-jp	jeuc	日本語(EUC)	4
euc-kr	keuc	韓国語(EUC)	4
euc-tw	yeuc	繁体字中国語(EUC)	4
fujitsu-cn	zfuj	簡体字中国語 (FACOM)	4
fujitsu-jp	jfug	日本語(FACOM)	4
fujitsu-ko	kfuj	韓国語(FACOM)	4
fujitsu-tw	yfuj	繁体字中国語 (FACOM)	4
gb18030	gbke	簡体字中国語 (GB18030)	4
greek	grek	ギリシャ語(ISO)	1
hebrew	hebr	ヘブライ語(ISO)	1
hitachi-cn	zhit	簡体字中国語 (HITAC)	6
hitachi-jp	jhit	日本語(HITAC)	6
hitachi-ko	khit	韓国語(HITAC)	6
hitachi-tw	yhit	繁体字中国語 (HITAC)	4
hitsas-jp	jhts	日本語(XHITAC)	4
hitsas-ko	khits	韓国語(XHITAC)	4
hitsas-tw	yhts	繁体字中国語 (XHITAC)	4

エンコーディング名	短縮名	説明	1 バイトの最大文字数
hp15-tw	yhpx	繁体字中国語 (HP15)	2
ibm-1381	zpce	簡体字中国語 (PCIBM)	2
ibm-930	j930	日本語カタカナ	4
ibm-933	kibm	韓国語(IBM)	4
ibm-935	zibm	簡体字中国語(IBM)	4
ibm-937	yibm	繁体字中国語(IBM)	4
ibm-939	jibm	日本語(IBM)	4
ibm-942	j942	日本語(PCIBM)	2
ibm-949	kpce	韓国語(PCIBM)	2
iso2022cncns	zist	繁体字中国語 (ISO-2022)	4
iso2022cn gb	ziso	簡体字中国語 (ISO-2022)	4
iso2022jp	jiso	日本語(ISO-2022)	8
iso2022kr	kiso	韓国語(ISO-2022)	4
latin1	lat1	西ヨーロッパ諸語 (ISO)	1
latin2	lat2	中央ヨーロッパ諸語 (ISO)	1
latin3	lat3	ラテン諸語 3(ISO 8859/3)	1
latin4	lat4	ラテン諸語 4(ISO 8859/4)	1
latin5	lat5	トルコ語(ISO)	1
latin6	lat6	バルト語(ISO)	1
latin8	lat8	ラテン諸語 8(ISO 8859/14)	1
latin9	lat9	ヨーロッパ諸語(ISO)	1
macos-1	jmac	日本語(PCMAC)	2
macos-2	ymac	繁体字中国語 (PCMAC)	2
macos-3	kmac	韓国語(PCMAC)	2

エンコーディング名	短縮名	説明	1 バイトの最大文字数
macos-25	zmac	簡体字中国語 (PCMAC)	2
ms-932	j932	日本語(PCMS)	2
ms-936	zwin	簡体字中国語 (PCMS)	2
ms-949	kwin	韓国語(PCMS)	2
ms-950	ywin	繁体字中国語 (PCMS)	2
msdos720	p720	アラビア語(MS- DOS)	1
msdos737	p737	ギリシャ語(MS- DOS)	1
msdos775	p775	バルト語(MS-DOS)	1
open_ed-037	eous	米国(Open Edition)	1
open_ed-275	eobr	ブラジル (OpenEdition)	1
open_ed-424	eoiv	ヘブライ語 (OpenEdition)	1
open_ed-425	ea2	アラビア語 (OpenEdition)	1
open_ed-838	eoht	タイ語(OpenEdition)	1
open_ed-870	eol2	中央ヨーロッパ諸語 (OpenEdition)	1
open_ed-875	eoel	ギリシャ語 (OpenEdition)	1
open_ed-905	eol3	ラテン諸語 3(Open Edition EBCDIC)	1
open_ed-924	eolt	ヨーロッパ諸語 (OpenEdition)	1
open_ed-930	oe30	カタカナ (OpenEdition)	4
open_ed-933	oe33	韓国語(OpenEdition)	4
open_ed-935	oe35	簡体字中国語 (OpenEdition)	4
open_ed-937	oe37	繁体字中国語 (OpenEdition)	4

エンコーディング名	短縮名	説明	1 バイトの最大文字数
open_ed-939	oe39	日本語(IBM)	4
open_ed-1025	eoey	キリル語 (OpenEdition)	1
open_ed-1026	eoer	トルコ語 (OpenEdition)	1
open_ed-1047	eo11	西ヨーロッパ諸語 (OpenEdition)	1
open_ed-1097	eofo	ペルシア諸語 (OpenEdition EBCDIC)	1
open_ed-1112	eo12	バルト語 (OpenEdition)	1
open_ed-1122	eoet	エストニア語 (OpenEdition)	1
open_ed-1130	eo13	ベトナム語 (OpenEdition)	1
open_ed-1140	eo40	北米諸語 (OpenEdition)	1
open_ed-1141	eo41	オーストリア/ドイツ (OpenEdition)	1
open_ed-1142	eo42	デンマーク/ノルウェー (OpenEdition)	1
open_ed-1143	eo43	フィンランド/スウェーデン (OpenEdition)	1
open_ed-1144	eo44	イタリア (OpenEdition)	1
open_ed-1145	eo45	スペイン (OpenEdition)	1
open_ed-1146	eo46	英国(OpenEdition)	1
open_ed-1147	eo47	フランス (OpenEdition)	1
open_ed-1148	eo48	国際化対応 (OpenEdition)	1
open_ed-1149	eo15	アイスランド (OpenEdition EBCDIC)	1
open_ed-1153	eo53	Latin 2 Euro OpenEdition EBCDIC	1

エンコーディング名	短縮名	説明	1 バイトの最大文字数
open_ed-1154	e054	Cyrillic Euro OpenEdition EBCDIC	1
open_ed-1155	e055	Turkey Euro OpenEdition EBCDIC	1
open_ed-1156	e056	Baltic Euro OpenEdition EBCDIC	1
open_ed-1157	e057	Estonia Euro OpenEdition EBCDIC	1
open_ed-1158	e058	Cyrillic Ukraine Euro 1158 OpenEdition EBCDIC	1
open_ed-1160	e060	1160 OpenEdition EBCDIC	1
open_ed-1164	e064	1164 OpenEdition EBCDIC	1
pciscii806	p806	インド(PC)	1
pcoem437	p437	米国(IBM-PC)	1
pcoem850	p850	西ヨーロッパ諸語 (IBM-PC)	1
pcoem852	p852	中央ヨーロッパ諸語 (IBM-PC)	1
pcoem857	p857	トルコ語(IBM-PC)	1
pcoem858	p858	ヨーロッパ諸語 (IBM-PC)	1
pcoem860	p860	ポルトガル語(MS- DOS)	1
pcoem862	p862	ヘブライ語(IBM-PC)	1
pcoem863	p863	フランス語(カナダ) (IBM-PC)	1
pcoem864	p864	アラビア語(IBM-PC)	1
pcoem865	p865	北欧諸語(IBM-PC)	1
pcoem866	p866	キリル語(IBM-PC)	1
pcoem869	p869	ギリシャ語(IBM-PC)	1

エンコーディング名	短縮名	説明	1 バイトの最大文字数
pcoem874	p874	タイ語(IBM-PC)	1
pcoem921	p921	バルト語(IBM-PC)	1
pcoem922	p922	エストニア語(IBM-PC)	1
pcoem1129	pvie	ベトナム語(IBM-PC)	1
pc1098	po98	ペルシア諸語(PC)	1
roman8	rom8	HP Roman 8	1
shift-jis	SJIS	日本語(SJIS)	2
thai	thai	タイ語(ISO)	1
us-ascii	ansi	すべての ASCII エンコーディングと互換性のあるデータセットを作成可能	1
utf-8	utf8	Unicode (UTF-8)	4
utf-16be	u16b	Unicode (UTF-16BE)	2 *
utf-16le	u16l	Unicode (UTF-16LE)	2 *
utf-32be	u32b	Unicode (UTF-32BE)	4 **
utf-32le	u32l	Unicode (UTF-32LE)	4 **
warabic	wara	アラビア語 (Windows)	1
wbaltic	wbal	バルト語(Windows)	1
wcyrillic	wcyr	キリル語(Windows)	1
wgreek	wgrk	ギリシャ語 (Windows)	1
whebrew	wheb	ヘブライ語 (Windows)	1
wlatin1	wlt1	西ヨーロッパ諸語 (Windows)	1
wlatin2	wlt2	中央ヨーロッパ諸語 (Windows)	1
wturkish	wtur	トルコ語(Windows)	1

エンコーディング名	短縮名	説明	1 バイトの最大文字数
wvietnamese	wvie	ベトナム語 (Windows)	1

- * UTF-16BE と UTF-16LE は各文字 2 バイトの固定長になります。
** UTF-32BE と UTF-32LE は各文字 4 バイトの固定長になります。

21 章

SAS セッションのエンコーディング値

UNIX エンコーディング値	573
Windows エンコーディング値	574
z/OS エンコーディング値	575

UNIX エンコーディング値

次の表のエンコーディングは、UNIX 環境で有効です。

注: エンコーディング値にハイフン(-)が含まれる場合は、エンコーディング値を引用符で囲んでください。

表 21.1 UNIX の 1 バイトエンコーディング

ENCODING=値	説明
arabic	アラビア語(ISO 8859-6)
cyrillic	キリル語(ISO 8859-5)
greek	ギリシャ語(ISO 8859-7)
hebrew	ヘブライ語(ISO 8859-8)
latin1	西ヨーロッパ諸語(ISO 8859-1)
latin2	中央ヨーロッパ諸語(ISO 8859-2)
latin5	トルコ語(ISO 8859-9)
latin6	バルト語(ISO 8859-4)
latin8	ケルト語(ISO 8859-14)
latin9	ヨーロッパ諸語(ISO 8859-15)
thai	タイ語(ISO 8859-11)

表 21.2 UNIX の 2 バイトエンコーディング

ENCODING=値	説明
big5	繁体字中国語(Big5)
euc-cn	簡体字中国語(EUC)
euc-jp	日本語(EUC)
euc-kr	韓国語(EUC)
euc-tw	繁体字中国語(EUC)
shift-jis	日本語(SJIS)

UNIX では、UTF-8 Unicode エンコーディングもサポートしています。

Windows エンコーディング値

次の表のエンコーディングは、Windows 動作環境で有効です。

注: エンコーディング値にハイフン(-)が含まれる場合は、エンコーディング値を引用符で囲んでください。

表 21.3 Windows の 1 バイトエンコーディング

説明	Windows ENCODING=値	MS-DOS ENCODING=値	IBM-PC ENCODING=値
アラビア語	warabic	msdos720	pcoem864
バルト諸語	wbaltic	msdos775	pcoem921
中央ヨーロッパ諸語	wlatin2	n/a	pcoem852
キリル語	wcyrillic	n/a	pcoem866 pcoem855
エストニア語	wbaltic	n/a	pcoem922
ヨーロッパ諸語	n/a	n/a	pcoem858
ペルシア語	n/a	n/a	pc1098
フランス語(カナダ)	wlatin1	n/a	pcoem863
ギリシャ語	wgreek	msdos737	n/a
ヘブライ語	whebrew	n/a	pcoem862
インド文字符号	n/a	n/a	pciscii806

説明	Windows ENCODING=値	MS-DOS ENCODING=値	IBM-PC ENCODING=値
北欧諸語	n/a	n/a	pcoem865
ポルトガル語	wlatin1	pcoem860	n/a
タイ語	n/a	n/a	pcoem874
トルコ語	wturkish	n/a	pcoem857
米国	wlatin1	n/a	pcoem437
ベトナム語	wvietnamese	n/a	n/a
西ヨーロッパ諸語	wlatin1	n/a	pcoem858

表 21.4 Windows の 2 バイトエンコーディング

説明	PCMS ENCODING=値	ベンダなし ENCODING=値
繁体字中国語	ms-950	big5
簡体字中国語	ms-936	n/a
日本語	ms-932	shift-jis
韓国語	ms-949	n/a

注: Windows では、UTF-8 Unicode エンコーディングもサポートしています。

z/OS エンコーディング値

次の表のエンコーディングは、z/OS 動作環境で有効です。

注: エンコーディング値にハイフン(-)が含まれる場合は、エンコーディング値を引用符で囲んでください。

表 21.5 z/OS の 1 バイトエンコーディング

エンコーディング ENCODING=値	説明
EBCDIC037	EBCDIC cp037-古い北米の言語
EBCDIC275	EBCDIC cp275-ブラジル
EBCDIC425	EBCDIC cp425-アラビア語
EBCDIC838	EBCDIC cp838-タイ語

エンコーディング ENCODING=値	説明
EBCDIC870	EBCDIC cp870-中央ヨーロッパ諸語
EBCDIC875	EBCDIC cp875-ギリシャ語
EBCDIC905	EBCDIC cp905-Latin 3
EBCDIC924	EBCDIC cp924-西ヨーロッパ諸語
EBCDIC1025	EBCDIC cp1025-キリル語
EBCDIC1026	EBCDIC cp1026-トルコ語
EBCDIC1047	EBCDIC cp1047-Latin1
EBCDIC1097	EBCDIC cp1097-ペルシア諸語
EBCDIC1112	EBCDIC cp1112-バルト語
EBCDIC1122	EBCDIC cp1122-エストニア語
EBCDIC1130	EBCDIC cp1130-ベトナム語
EBCDIC1137	EBCDIC cp1137-デーバナーガリー文字
EBCDIC1140	EBCDIC cp1140-北米諸語
EBCDIC1141	EBCDIC cp1141-ドイツ語/オーストリア語
EBCDIC1142	EBCDIC cp1142-デンマーク語/ノルウェー語
EBCDIC1143	EBCDIC cp1143-フィンランド語/スウェーデン語
EBCDIC1144	EBCDIC cp1144-イタリア語
EBCDIC1145	EBCDIC cp1145-スペイン語
EBCDIC1146	EBCDIC cp1146-英語(UK)
EBCDIC1147	EBCDIC cp1147-フランス語
EBCDIC1148	EBCDIC cp1148-国際
EBCDIC1149	EBCDIC cp1149-アイスランド語
EBCDIC1153	EBCDIC cp1153-ユーロ符号を含むラテン諸語 2
EBCDIC1154	EBCDIC cp1154-ユーロ符号を含むキリル諸語
EBCDIC1155	EBCDIC cp1155-ユーロ符号を含むトルコ語
EBCDIC1156	EBCDIC cp1156-ユーロ符号を含むバルト諸語
EBCDIC1157	EBCDIC cp1157-ユーロ符号を含むエストニア語
EBCDIC1158	EBCDIC cp1158-ユーロ符号を含むウクライナキリル語

エンコーディング ENCODING=値	説明
OPEN_ED-037	OpenEdition EBCDIC cp037-古い北米の言語
OPEN_ED-275	OpenEdition EBCDIC cp275-ブラジル
OPEN_ED-425	OpenEdition EBCDIC cp425-アラビア語
OPEN_ED-838	OpenEdition EBCDIC cp838-タイ語
OPEN_ED-870	OpenEdition EBCDIC cp870-中央ヨーロッパ諸語
OPEN_ED-875	OpenEdition EBCDIC cp875-ギリシャ語
OPEN_ED-905	OpenEdition EBCDIC cp905-Latin 3
OPEN_ED-924	OpenEdition EBCDIC cp924-西ヨーロッパ諸語
OPEN_ED-1025	OpenEdition EBCDIC cp1025-キリル語
OPEN_ED-1026	OpenEdition EBCDIC cp1026-トルコ語
OPEN_ED-1047	OpenEdition EBCDIC cp1047-Latin1
OPEN_ED_1097	OpenEdition EBCDIC cp1097-ペルシア諸語
OPEN_ED-1112	OpenEdition EBCDIC cp1112-バルト語
OPEN_ED-1122	OpenEdition EBCDIC cp1122-エストニア語
OPEN_ED-1130	OpenEdition EBCDIC cp1130-ベトナム語
OPEN_ED-1137	OpenEdition EBCDIC cp1137-デーバナーガリー文字
OPEN_ED-1140	OpenEdition EBCDIC cp1140-北米の言語
OPEN_ED-1141	OpenEdition EBCDIC cp1141-ドイツ語/オーストリア語
OPEN_ED-1142	OpenEdition EBCDIC cp1142-デンマーク語/ノルウェー語
OPEN_ED-1143	OpenEdition EBCDIC cp1143-フィンランド語/スウェーデン語
OPEN_ED-1144	OpenEdition EBCDIC cp1144-イタリア語
OPEN_ED-1145	OpenEdition EBCDIC cp1145-スペイン語
OPEN_ED-1146	OpenEdition EBCDIC cp1146-英語(UK)
OPEN_ED-1147	OpenEdition EBCDIC cp1147-フランス語
OPEN_ED-1148	OpenEdition EBCDIC cp1148-国際
OPEN_ED-1149	OpenEdition EBCDIC cp1149-アイスランド語
OPEN_ED-1153	OpenEdition EBCDIC cp1153-ユーロ符号を含むラテン諸語 2
OPEN_ED-1154	OpenEdition EBCDIC cp1154-ユーロ符号を含むキリル諸語

エンコーディング ENCODING=値	説明
OPEN_ED-1155	OpenEdition EBCDIC cp1155-ユーロ符号を含むトルコ語
OPEN_ED-1156	OpenEdition EBCDIC cp1156-ユーロ符号を含むバルト諸語
OPEN_ED-1157	OpenEdition EBCDIC cp1157-ユーロ符号を含むエストニア語
OPEN_ED-1158	OpenEdition EBCDIC cp1158-ユーロ符号を含むウクライナキリル語

表 21.6 z/OS の 2 バイトエンコーディング

説明	ENCODING=値
日本語	OPEN_ED-939
韓国語	OPEN_ED-933
簡体字中国語	OPEN_ED-935
繁体字中国語	OPEN_ED-937

12 部

付録

付録1	
追加各国語サポートの言語要素	581

付録 1

追加各国語サポートの言語要素

追加各国語サポートの言語要素	582
ディクショナリ	582
EURDFDDw. 出力形式	582
EURDFDEw. 出力形式	584
EURDFDNw. 出力形式	585
EURDFDTw.d 出力形式	586
EURDFDWNw. 出力形式	588
EURDFMNw. 出力形式	590
EURDFMYw. 出力形式	592
EURDFWDXw. 出力形式	593
EURDFWKXw. 出力形式	596
EURFRATSw.d 出力形式	598
EURFRBEFw.d 出力形式	600
EURFRCHFw.d 出力形式	601
EURFRDEMw.d 出力形式	602
EURFRDKKw.d 出力形式	603
EURFRESPw.d 出力形式	605
EURFRFIMw.d 出力形式	606
EURFRFRFw.d 出力形式	607
EURFRGBPw.d 出力形式	609
EURFRGRDw.d 出力形式	610
EURFRHUFw.d 出力形式	611
EURFRIEPw.d 出力形式	612
EURFRITLw.d 出力形式	614
EURFRLUFw.d 出力形式	615
EURFRNLGw.d 出力形式	616
EURFRNOKw.d 出力形式	618
EURFRPLZw.d 出力形式	619
EURFRPTEw.d 出力形式	620
EURFRROLw.d 出力形式	621
EURFRRURw.d 出力形式	623
EURFRSEKw.d 出力形式	624
EURFRSITw.d 出力形式	625
EURFRTRLw.d 出力形式	626
EURFRYUDw.d 出力形式	628
EURTOATSw.d 出力形式	629
EURTOBEFw.d 出力形式	630
EURTOCHFw.d 出力形式	631
EURTOCZKw.d 出力形式	633
EURTODEMw.d 出力形式	634
EURTODKKw.d 出力形式	635

EURTOESPw.d 出力形式	637
EURTOFIMw.d 出力形式	638
EURTOFRFw.d 出力形式	639
EURTOGBPw.d 出力形式	640
EURTOGRDw.d 出力形式	642
EURTOHUFw.d 出力形式	643
EURTOIEPw.d 出力形式	644
EURTOITLw.d 出力形式	645
EURTOLUFw.d 出力形式	647
EURTONLGw.d 出力形式	648
EURTONOKw.d 出力形式	649
EURTOPLZw.d 出力形式	651
EURTOPTew.d 出力形式	652
EURTOROLw.d 出力形式	653
EURTORURw.d 出力形式	654
EURTOSEKw.d 出力形式	656
EURTOSITw.d 出力形式	657
EURTOTRLw.d 出力形式	658
EURTOYUDw.d 出力形式	660
EURDFDEw. 入力形式	661
EURDFDTw. 入力形式	662
EURDFMYw. 入力形式	664
EUROCURR 関数	665

追加各国語サポートの言語要素

次の EUR 言語要素は、NL 言語要素によって置き換えられました。SAS 9.3 は EUR 要素をサポートしていますが、SAS では、NL 要素を使用することをお勧めします。

ディクショナリ

EURDFDDw. 出力形式

SAS 日付値が、指定したヨーロッパ言語で *dd.mm.yy* または *dd.mm.yyyy* 形式で書き出されます。

カテゴリ: 日付と時間

配置: 右

構文

EURDFDDw.

説明

w

出力フィールドの幅を指定します。

デフォルト: 8 (フィンランド語の場合は 10)

範囲: 2–10

ヒント: w が 2 から 5 の場合、SAS では可能な限り多くの月と日を表示します。 w が 7 の場合、日付はスラッシュのない 2 桁の年で表示され、値は出力フィールドで右詰めになります。

詳細

EURDFDDw.出力形式では、SAS 日付値を $dd.mm.yy$ または $dd.mm.yyyy$ 形式で書き出します。

dd

月の日を表す 2 桁の整数です。

mm

月を表す 2 桁の整数です。

yy または $yyyy$

年を表す 2 桁または 4 桁の整数です。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“[DFLANG=システムオプション: UNIX、Windows および z/OS](#)” (464 ページ) を参照してください。言語接頭辞を出力形式で指定した場合は、DFLANG=システムオプションの指定を無視します。

例

例の表では、15342 を入力値として使用します。この値は 2002 年 1 月 2 日に相当する SAS 日付値です。最初の PUT ステートメントは、DFLANG=システムオプションがスペイン語に設定されることを前提としています。

```
options dflang=spanish;
```

2 番目の PUT ステートメントでは、スペイン語の接頭辞を含む出力形式を使用して、指定したヨーロッパ言語で SAS 日付値を書き出します。3 番目の PUT ステートメントでは、フランス語の接頭辞を含む出力形式を使用して、指定したヨーロッパ言語で SAS 日付値を書き出します。したがって、DFLANG=オプションの値は無視されます。

```
options dflang=spanish;
data _null_;
input day;
put day eurdfdd8.;
datalines;
15342
;
```

ステートメント	結果
	----+-----1
put date eurdfdd8.;	02.01.02
put date espdfdd8.;	02.01.02
put date fradfd8.;	02/01/02

EURDFDE w . 出力形式

SAS 日付値が、指定したヨーロッパ言語で $ddmmmyy$ または $ddmmmyyyy$ 形式で書き出されます。

カテゴリ: 日付と時間

配置: 右

構文

EURDFDE w .

説明

w

出力フィールドの幅を指定します。

デフォルト: 7 (フィンランド語以外)

範囲: 5–9 (フィンランド語以外)

注: フィンランド語(FIN)の接頭辞を使用する場合、 w の値の範囲は 9-10 でデフォルト値は 9 です。

詳細

EURDFDE w . 出力形式では、SAS 日付値を $ddmmmyy$ または $ddmmmyyyy$ 形式で書き出します。

dd

月の日を表す整数です。

mmm

月名の最初の 3 文字になります。

yy または $yyyy$

年を表す 2 桁または 4 桁の整数です。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“[DFLANG=システムオプション: UNIX、Windows および z/OS](#)” (464 ページ) を参照してください。出力形式で言語接頭辞を指定した場合、DFLANG=オプションは無視されます。

注: EUR 日付の出力形式では、ヨーロッパ系の文字セットとエンコーディングが必要です。一部の出力形式では、ヨーロッパ系以外のエンコーディングを使用すると正常に機能しません。DBCS 環境で使用すると、デフォルトの出力幅と最大幅が 1 バイトシステムでの幅より広くなり、2 バイト文字が使用されます。ただし、UTF-8 などのヨーロッパ系の文字セットもサポートするセッションエンコーディングを使用する必要があります。

例

例の表では、15342 を入力値として使用します。この値は 2002 年 1 月 2 日に相当する SAS 日付値です。最初の PUT ステートメントは、DFLANG=システムオプションがスペイン語に設定されることを前提としています。

```
options dflang=spanish;
```

2 番目の PUT ステートメントでは、スペイン語の接頭辞を含む出力形式を使用して、スペイン語で SAS 日付値を書き出します。3 番目の PUT ステートメントでは、フランス語の接頭辞を含む出力形式を使用して、フランス語で SAS 日付値を書き出します。したがって、DFLANG=オプションの値は無視されます。

```
options dflang=spanish;
data _null_;
input day;
put day eurdfde9.;
put day espdfde9.;
put day fradfde9.;
datalines;
15342
;
```

ステートメント	結果
	----+----1
put date eurdfde9.;	02ene2002
put date espdfde9.;	02ene2002
put date fradfde9.;	02jan2002

EURDFDNw. 出力形式

指定したヨーロッパ言語で、SAS 日付値から曜日が書き出されます。

カテゴリ: 日付と時間

配置: 右

構文

EURDFDNw.

説明

w
出力フィールドの幅を指定します。

デフォルト: 1

範囲: 1–32

詳細

EURDFDNw.出力形式では、SAS 日付値を *day-of-the-week* 形式で書き出します。

day-of-the-week

1=月曜日、2=火曜日のように表されます。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭

辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“DFLANG=システムオプション: UNIX、Windows および z/OS” (464 ページ)を参照してください。出力形式で言語接頭辞を指定した場合、DFLANG=オプションは無視されます。

注: EUR 日付の出力形式では、ヨーロッパ系の文字セットとエンコーディングが必要です。一部の出力形式では、ヨーロッパ系以外のエンコーディングを使用しても正常に機能します。DBCS 環境で使用すると、デフォルトの出力幅と最大幅が 1 バイトシステムでの幅より広くなり、2 バイト文字が使用されます。ただし、UTF-8 などのヨーロッパ系の文字セットもサポートするセッションエンコーディングを使用する必要があります。

例

例の表では、15342 を入力値として使用します。この値は 2002 年 1 月 2 日に相当する SAS 日付値です。最初の PUT ステートメントは、DFLANG=システムオプションがスペイン語に設定されることを前提としています。

```
options dflang=spanish;
```

2 番目の PUT ステートメントでは、スペイン語の接頭辞を含む出力形式を使用して、スペイン語で曜日を書き出します。3 番目の PUT ステートメントでは、イタリア語の接頭辞を含む出力形式を使用して、イタリア語で曜日を書き出します。したがって、DFLANG=オプションの値は無視されます。

```
options dflang=spanish;
data _null_;
input day;
put day eurdfdn.;
put day espdfdn.;
put day itadfdn.;
datalines;
15342
;
```

ステートメント	結果
	----+----1
put day eurdfdn.;	3
put day espdfdn.;	3
put day itadfdn.;	3

EURDFDTw.d 出力形式

SAS 日時値が、指定したヨーロッパ言語で *ddmmmyy:hh:mm:ss.ss* または *ddmmmyyyy hh:mm:ss.ss* 形式で書き出されます。

カテゴリ: 日付と時間

配置: 右

構文

EURDFDTw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 16

範囲: 7–40

ヒント: SAS 日時値を日、時、秒で書き出す場合、幅(*w*)は 16 以上にする必要があります。秒の小数部分とともに値を返す場合、幅に 2 桁を追加します。

d

数値の小数点以下の桁数を指定します。

範囲: 1–39

制限事項:

w より小さい数にします。

もし $w - d < 17$ の場合、SAS により小数点以下の値は切り捨てられます。

詳細

EURDFDTw.d 出力形式では、SAS 日時値を *ddmmmyy:hh:mm:ss.ss* 形式で書き出します。

dd

月の日を表す整数です。

mmm

月名の最初の 3 文字になります。

yy または *yyyy*

年を表す 2 桁または 4 桁の整数です。

hh

00 から 23 の範囲の時間数です。

mm

00 から 59 の範囲の分数です。

ss.ss

小数点以下の秒の端数を含む、00 から 59 の範囲の秒数です。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“[DFLANG=システムオプション: UNIX、Windows および z/OS](#)” (464 ページ) を参照してください。出力形式で言語接頭辞を指定した場合、DFLANG=オプションは無視されます。

注: EUR 日付の出力形式では、ヨーロッパ系の文字セットとエンコーディングが必要です。一部の出力形式では、ヨーロッパ系以外のエンコーディングを使用すると正常に機能しません。DBCS 環境で使用すると、デフォルトの出力幅と最大幅が 1 バイトシステムでの幅より広くなり、2 バイト文字が使用されます。ただし、UTF-8 などのヨーロッパ系の文字セットもサポートするセッションエンコーディングを使用する必要があります。

例

例の表では 1347453583 を入力値として使用します。この値は 2002 年 9 月 12 日、午後 12:39:43 に相当する SAS 日時値です。最初の PUT ステートメントは、DFLANG=システムオプションがドイツ語に設定されることを前提としています。

```
options dflang=german;
```

2 番目の PUT ステートメントでは、ドイツ語の接頭辞を含む出力形式を使用して、ドイツ語で SAS 日時値を書き出します。3 番目の PUT ステートメントでは、イタリア語の接頭辞を含む出力形式を使用して、イタリア語で SAS 日時値を書き出します。したがって、DFLANG=オプションの値は無視されます。

```
options dflang=german;
data _null_;
input date;
put date= ;
put date eurdfdt20.;
put date deufdt20.;
put date itadfdt20.;
datalines;
1347453583;
;
run;
```

ステートメント	結果
	----+----1----+----2
put date eurdfdt20.;	12Sep2002:12:39:43
put date deufdt20.;	12Sep2002:12:39:43
put date itadfdt20.;	12Set2002:12:39:43

EURDFDWNw. 出力形式

指定したヨーロッパ言語で、SAS 日付値から日の名前が書き出されます。

カテゴリ: 日付と時間

配置: 右

構文

EURDFDWNw.

説明

w
出力フィールドの幅を指定します。

デフォルトは使用する言語接頭辞に依存します。次の表に、各言語のデフォルト値を示します。

言語	デフォルト
アフリカーンス語(AFR)	9
カタロニア語(CAT)	9
クロアチア語(CRO)	10
チェコ語(CSY)	7
デンマーク語(DAN)	7
オランダ語(NLD)	9
フィンランド語(FIN)	11
フランス語(FRA)	8
ドイツ語(DEU)	10
ハンガリー語(HUN)	9
イタリア語(ITA)	9
マケドニア語(MAC)	10
ノルウェー語(NOR)	7
ポーランド語(POL)	12
ポルトガル語(PTG)	13
ロシア語(RUS)	11
スロベニア語(SLO)	10
スペイン語(ESP)	9
スウェーデン語(SVE)	7
フランス語(スイス) (FRS)	8
ドイツ語(スイス) (DES)	10

デフォルト: 使用する言語接頭辞に依存します。

範囲: 1-32

ヒント: *w* を省略すると、日名全体が出力されます。

詳細

必要に応じて、SAS により出力形式の幅に合うように日名が切り捨てられます。EURDFDWNw.出力形式では、SAS 日付値を *day-name* 形式で書き出します。

day-name

日の名前です。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“DFLANG=システムオプション: UNIX、Windows および z/OS” (464 ページ) を参照してください。出力形式で言語接頭辞を指定した場合、DFLANG=オプションは無視されます。

注: EUR 日付の出力形式では、ヨーロッパ系の文字セットとエンコーディングが必要です。一部の出力形式では、ヨーロッパ系以外のエンコーディングを使用すると正常に機能しません。DBCS 環境で使用すると、デフォルトの出力幅と最大幅が 1 バイトシステムでの幅より広くなり、2 バイト文字が使用されます。ただし、UTF-8 などのヨーロッパ系の文字セットもサポートするセッションエンコーディングを使用する必要があります。

例

例の表では、15344 を入力値として使用します。この値は 2002 年 1 月 4 日に相当する SAS 日付値です。最初の PUT ステートメントは、DFLANG=システムオプションがフランス語に設定されることを前提としています。

```
options dflang=french;
put day eurdfdwn8.;
```

2 番目の PUT ステートメントでは、フランス語の接頭辞を含む出力形式を使用して、フランス語で曜日を書き出します。3 番目の PUT ステートメントでは、スペイン語の接頭辞を含む出力形式を使用して、スペイン語で曜日を書き出します。したがって、DFLANG=オプションの値は無視されます。

```
options dflang=french;
data _null_;
input day;
put day eurdfdwn8. ;
put day fradfdwn8. ;
put day espdfdwn8. ;
datalines;
15344
;
run;
```

ステートメント	結果
	----+----1
put day eurdfdwn8.;	Vendredi
put day fradfdwn8.;	Vendredi
put day espdfdwn8.;	viernes

EURDFMNw. 出力形式

指定したヨーロッパ言語で、SAS 日付値から月の名前が書き出されます。

カテゴリ: 日付と時間

配置: 右

構文

EURDFMN w .

説明

w

出力フィールドの幅を指定します。

デフォルト: 9 (フィンランド語とスペイン語以外)

範囲: 1–32

注: フィンランド語(FIN)の接頭辞を使用する場合、 w のデフォルト値は 11 です。スペイン語(ESP)の接頭辞を使用する場合、 w のデフォルト値は 10 です。

詳細

必要に応じて、出力形式の幅に合うように月名が切り捨てられます。EURDFMN w .出力形式では、SAS 日付値を *month-name* 形式で書き出します。

month-name

月の名前です。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“[DFLANG= システムオプション: UNIX、Windows および z/OS](#)” (464 ページ)を参照してください。出力形式で言語接頭辞を指定した場合、DFLANG=オプションは無視されます。

注: EUR 日付の出力形式では、ヨーロッパ系の文字セットとエンコーディングが必要です。一部の出力形式では、ヨーロッパ系以外のエンコーディングを使用すると正常に機能しません。DBCS 環境で使用すると、デフォルトの出力幅と最大幅が 1 バイトシステムでの幅より広くなり、2 バイト文字が使用されます。ただし、UTF-8 などのヨーロッパ系の文字セットもサポートするセッションエンコーディングを使用する必要があります。

例

例の表では、15344 を入力値として使用します。この値は 2002 年 1 月 4 日に相当する SAS 日付値です。最初の PUT ステートメントは、DFLANG=システムオプションがイタリア語に設定されることを前提としています。

```
options dflang=ita;
```

2 番目の PUT ステートメントでは、イタリア語の接頭辞を含む出力形式を使用して、イタリア語で月名を書き出します。3 番目の PUT ステートメントでは、ドイツ語の接頭辞を含む出力形式を使用して、ドイツ語で月名を書き出します。したがって、DFLANG=オプションの値は無視されます。

```
options dflang=ita;
data _null_;
input day;
put day eurdfmn10.;
put day itadfmn10.;
put day deudfmn10.;
datalines;
```

```
15344
;
run;
```

ステートメント	結果
	----+----1
put date eurdfmn10.;	janvier
put date itadfmn10.;	Gennaio
put date deudfmn10.;	Januar

EURDFMYw. 出力形式

SAS 日付値が、指定したヨーロッパ言語で *mmmyy* または *mmmyyyy* 形式で書き出されます。

カテゴリ: 日付と時間

配置: 右

構文

EURDFMYw.

説明

w

出力フィールドの幅を指定します。

デフォルト: 5 (フィンランド語以外)

範囲: 5–7

注: フィンランド語(FIN)の接頭辞を使用する場合、w の値はデフォルト値の 8 にする必要があります。

詳細

EURDFMYw.出力形式では、SAS 日付値を *mmmyy* 形式で書き出します。

mmm

月名の最初の 3 文字になります。

yy または *yyyy*

年を表す 2 桁または 4 桁の整数です。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“[DFLANG=システムオプション: UNIX、Windows および z/OS](#)” (464 ページ) を参照してください。出力形式で言語接頭辞を指定した場合、DFLANG=オプションは無視されます。

注: EUR 日付の出力形式では、ヨーロッパ系の文字セットとエンコーディングが必要です。一部の出力形式では、ヨーロッパ系以外のエンコーディングを使用すると正

常に機能しません。DBCS 環境で使用すると、デフォルトの出力幅と最大幅が 1 バイトシステムでの幅より広くなり、2 バイト文字が使用されます。ただし、UTF-8 などのヨーロッパ系の文字セットもサポートするセッションエンコーディングを使用する必要があります。

例

例の表では、15342 を入力値として使用します。この値は 2002 年 1 月 2 日に相当する SAS 日付値です。最初の PUT ステートメントは、DFLANG=システムオプションがスペイン語に設定されることを前提としています。

```
options dflang=spanish;
```

2 番目の PUT ステートメントでは、スペイン語の接頭辞を含む出力形式を使用して、スペイン語で月名を書き出します。3 番目の PUT ステートメントでは、フランス語の接頭辞を含む出力形式を使用して、フランス語で月名を書き出します。したがって、DFLANG=オプションの値は無視されます。

```
options dflang=spanish;
data _null_;
input date;
put date eurdfmy7.;
put date espdfmy7.;
put date fradfmy7.;
datalines;
15342
;
```

ステートメント	結果
	----+----1
put date eurdfmy7.;	ene2002
put date espdfmy7.;	ene2002
put date fradfmy7.;	jan2002

EURDFWDXw. 出力形式

指定したヨーロッパ言語で、SAS 日付値から月名、日名および年名が *dd month-name yy* (または *yyyy*)形式で書き出されます。

カテゴリ: 日付と時間

配置: 右

構文

EURDFWDXw.

説明**w**

出力フィールドの幅を指定します。

デフォルトは使用する言語接頭辞に依存します。次の表に、各言語のデフォルト値を示します。

言語	最大値	デフォルト
アフリカーンス語(AFR)	37	29
カタロニア語(CAT)	40	16
クロアチア語(CRO)	40	16
チェコ語(CSY)	40	16
デンマーク語(DAN)	18	18
オランダ語(NLD)	37	29
フィンランド語(FIN)	20	20
フランス語(FRA)	18	18
ドイツ語(DEU)	18	18
ハンガリー語(HUN)	40	18
イタリア語(ITA)	17	17
マケドニア語(MAC)	40	17
ノルウェー語(NOR)	17	17
ポーランド語(POL)	40	20
ポルトガル語(PTG)	37	23
ロシア語(RUS)	40	16
スロベニア語(SLO)	40	17
スペイン語(ESP)	24	24
スウェーデン語(SVE)	17	17
フランス語(スイス) (FRS)	17	17
ドイツ語(スイス) (DES)	18	18

デフォルト: 使用する言語接頭辞に依存します。

範囲: 3-(最大幅)

ヒント: 完全な曜日と月を含めるのに *w* の値が小さすぎる場合は、必要に応じて省略されます。

詳細

EURDFWDXw.出力形式では、SAS 日付値を *dd month-name yy* または *dd month-name yyyy* 形式で書き出します。

dd

月の日を表す整数です。

month-name

月の名前です。

yy または *yyyy*

年を表す 2 桁または 4 桁の整数です。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“[DFLANG=システムオプション: UNIX、Windows および z/OS](#)” (464 ページ)を参照してください。出力形式で言語接頭辞を指定した場合、DFLANG=オプションは無視されます。

注: EUR 日付の出力形式では、ヨーロッパ系の文字セットとエンコーディングが必要です。一部の出力形式では、ヨーロッパ系以外のエンコーディングを使用すると正常に機能しません。DBCS 環境で使用すると、デフォルトの出力幅と最大幅が 1 バイトシステムでの幅より広くなり、2 バイト文字が使用されます。ただし、UTF-8 などのヨーロッパ系の文字セットもサポートするセッションエンコーディングを使用する必要があります。

比較

EURDFWKXw.出力形式は EURDFWDXw.出力形式と同じですが、EURDFWKX w.出力形式では *dd* の前に *day-of-week* が追加される点が異なります。

例

例の表では、15342 を入力値として使用します。この値は 2002 年 1 月 2 日に相当する SAS 日付値です。最初の PUT ステートメントは、DFLANG=システムオプションがオランダ語に設定されることを前提としています。

```
options dflang=dutch;
```

2 番目の PUT ステートメントでは、オランダ語の接頭辞を含む出力形式を使用して、オランダ語で月名を書き出します。3 番目の PUT ステートメントでは、イタリア語の接頭辞を含む出力形式を使用して、イタリア語で月名を書き出します。したがって、DFLANG=オプションの値は無視されます。

```
options dflang=dutch;
data _null_;
input date;
put date eurdfwdx29.;
put date nlddfwdx29.;
put date itadfdwx17.;
datalines;
15342
;
```

ステートメント	結果
	-----1-----2-----3

ステートメント	結果
put day eurdfwdx29.;	2 januari 2002
put day nlddfwdx29.;	2 januari 2002
put day itadfwdx17.;	02 Gennaio 1998

EURDFWKX^w. 出力形式

指定したヨーロッパ言語で、SAS 日付値から曜日および日付が *day-of-week, dd month-name yy* (または *yyyy*) 形式で書き出されます。

カテゴリ: 日付と時間

配置: 右

構文

EURDFWKX^w.

説明

^w

出力フィールドの幅を指定します。

デフォルトは使用する言語接頭辞に依存します。次の表に、各言語のデフォルト値を示します。

言語	最小値	最大値	デフォルト
アフリカーンス語(AFR)	2	38	28
カタロニア語(CAT)	2	40	27
クロアチア語(CRO)	3	40	27
チェコ語(CSY)	2	40	25
デンマーク語(DAN)	2	31	31
オランダ語(NLD)	2	38	28
フィンランド語(FIN)	2	37	37
フランス語(FRA)	3	27	27
ドイツ語(DEU)	3	30	30
ハンガリー語(HUN)	3	40	28
イタリア語(ITA)	3	28	28

言語	最小値	最大値	デフォルト
マケドニア語(MAC)	3	40	29
ノルウェー語(NOR)	3	26	26
ポーランド語(POL)	2	40	34
ポルトガル語(PTG)	3	38	38
ロシア語(RUS)	2	40	29
スロベニア語(SLO)	3	40	29
スペイン語(ESP)	1	35	35
スウェーデン語(SVE)	3	26	26
フランス語(スイス) (FRS)	3	26	26
ドイツ語(スイス) (DES)	3	30	30

デフォルト: 使用する言語接頭辞に依存します。

ヒント: 完全な曜日と月を含めるのに *w* の値が小さすぎる場合は、必要に応じて省略されます。

詳細

EURDFWKX*w* 出力形式では、SAS 日付値を *day-of-week, dd month-name yy* (または *yyyy*) 形式で書き出します。

day-of-week

曜日の名前です。

dd

月の日を表す整数です。

month-name

月の名前です。

yy または *yyyy*

年を表す 2 桁または 4 桁の整数です。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“[DFLANG= システムオプション: UNIX、Windows および z/OS](#)” (464 ページ) を参照してください。出力形式で言語接頭辞を指定した場合、DFLANG=オプションは無視されます。

注: EUR 日付の出力形式では、ヨーロッパ系の文字セットとエンコーディングが必要です。一部の出力形式では、ヨーロッパ系以外のエンコーディングを使用すると正常に機能しません。DBCS 環境で使用すると、デフォルトの出力幅と最大幅が 1 バイトシステムでの幅より広くなり、2 バイト文字が使用されます。ただし、UTF-8 などのヨーロッパ系の文字セットもサポートするセッションエンコーディングを使用する必要があります。

比較

EURDFWKX_w出力形式は EURDFWDX_w出力形式と同じですが、EURDFWKX_w出力形式では *dd* の前に *day-of-week* が追加される点が異なります。

例

例の表では、15344 を入力値として使用します。この値は 2002 年 1 月 4 日に相当する SAS 日付値です。最初の PUT ステートメントは、DFLANG=システムオプションがドイツ語に設定されることを前提としています。

```
options dflang=German;
```

2 番目の PUT ステートメントでは、ドイツ語の接頭辞を含む出力形式を使用して、ドイツ語で月名を書き出します。3 番目の PUT ステートメントでは、イタリア語の接頭辞を含む出力形式を使用して、イタリア語で月名を書き出します。したがって、DFLANG=オプションの値は無視されます。

```
options dflang=german;
data _null_;
input date;
put date eurdfwxx30.;
put date deudfwxx30.;
put date itadfwxx17.;
datalines;
15344
;
run;
```

ステートメント	結果
	-----1-----2-----3
put date eurdfwxx30.;	Freitag, 4. Januar 2002
put date deudfwxx30.;	Freitag, 4. Januar 2002
put date itadfwxx17.;	Ven, 04 Gen 2002

EURFRATSw.d 出力形式

金額をオーストリアシリングからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRATSw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRATS *w.d* 出力形式では、金額をオーストラリアシリングからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRATSw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、オーストラリアシリングの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```

data _null_;
input amount;
put amount eurfrats5.;
put amount eurfrats9.2;
datalines;
50
5234.56
52345
;
run;

E4
E3,63
E380
E380,41
3.804
E3.804,06

```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrats5.;	E4
	put amount eurfrats9.2;	E3,63
5234.56	put amount eurfrats5.;	E380
	put amount eurfrats9.2;	E380,41
52345	put amount eurfrats5.;	3.804
	put amount eurfrats9.2;	E3.804,06

EURFRBEF $w.d$ 出力形式

金額をベルギーフランからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRBEF $w.d$

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURFRBEF $w.d$ 出力形式では、金額をベルギーフランからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRBEF $w.d$ 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ) を参照してください。

例

次の表に、ベルギーフランの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrbef5.;
put amount eurfrbef9.2;
datalines;
50
5234.56
52345
;
run;

8 put amount eurfrbef5.;
9 put amount eurfrbef9.2;
10 datalines;
E1
E1,24
E130
E129,76
1.298
E1.297,60
```

金額	ステートメント	結果
		----+-----1----2
50	put amount eurfrbef5.;	E1
	put amount eurfrbef9.2;	E1,24
5234.56	put amount eurfrbef5.;	E130
	put amount eurfrbef9.2;	E129,76
52345	put amount eurfrbef5.;	1.298
	put amount eurfrbef9.2;	E1.297,60

EURFRCHFw.d 出力形式

金額をスイスフランからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRCHFw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURFRCHFw.d 出力形式では、金額をスイスフランからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRCHFw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、スイスフランの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrchf5.;
put amount eurfrchf9.2;
datalines;
```

```

50
1234.56
12345
;
run;
SAS Log:
3 put amount eurfrchf5.;
4 put amount eurfrchf9.2;
5 datalines;
E31
E31,17
E770
E769,53
7.695
E7.694,94

```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrchf5.;	E31
	put amount eurfrchf9.2;	E31,17
1234.56	put amount eurfrchf5.;	E770
	put amount eurfrchf9.2;	E769,53
12345	put amount eurfrchf5.;	7.695
	put amount eurfrchf9.2;	E7.694,94

EURFRDEMw.d 出力形式

金額をドイツマルクからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRDEMw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRDEMw.d 出力形式では、金額をドイツマルクからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRDEMw.d 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ) を参照してください。

例

次の表に、ドイツマルクの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrdem5.;
put amount eurfrdem9.2;
datalines;
50
1234.56
12345
;
run;

8 put amount eurfrdem5.;
9 put amount eurfrdem9.2;
10 datalines;
E26
E25,56
E631
E631,22
6.312
E6.311,90
```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrdem5.;	E26
	put amount eurfrdem9.2.;	E25,56
1234.56	put amount eurfrdem5.;	E631
	put amount eurfrdem9.2.;	E631,22
12345	put amount eurfrdem5.;	6.312
	put amount eurfrdem9.2.;	E6.311,90

EURFRDCKw.d 出力形式

金額をデンマーククローネからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRDKK $w.d$

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURFRDKK $w.d$ 出力形式では、金額をデンマーククローネからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRDKK $w.d$ 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ) を参照してください。

例

次の表に、デンマーククローネの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```

data _null_;
input amount;
put amount eurfrdkk5.;
put amount eurfrdkk9.2;
datalines;
50
1234.56
12345
;
run;
SAS log:
3 put amount eurfrdkk5.;
4 put amount eurfrdkk9.2;
5 datalines;
E7
E6,68
E165
E164,83
1.648
E1.648,18

```

金額	ステートメント	結果
		----+----1----2
50	put amount eurfrdkk5.;	E7
	put amount eurfrdkk9.2;	E6,68

金額	ステートメント	結果
1234.56	put amount eurfrdkk5.;	E165
	put amount eurfrdkk9.2;	E164,83
12345	put amount eurfrdkk5.;	1.648
	put amount eurfrdkk9.2;	E1.648,18

EURFRESPw.d 出力形式

金額をスペインペセタからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRESPw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRESPw.d 出力形式では、金額をスペインペセタからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRESPw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、スペインペセタの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfresp5.;
put amount eurfresp9.2;
datalines;
200
20234.56
202345
;
run;
```

```

8 put amount eurfresp5.;
9 put amount eurfresp9.2;
10 datalines;
E1
E1,20
E122
E121,61
1.216
E1.216,12

```

金額	ステートメント	結果
		----+-----1-----2
200	put amount eurfresp5.;	E1
	put amount eurfresp9.2;	E1,20
20234.56	put amount eurfresp5.;	E122
	put amount eurfresp9.2;	E121,61
202345	put amount eurfresp5.;	1.216
	put amount eurfresp9.2;	E1.216,12

EURFRFIM $w.d$ 出力形式

金額をフィンランドマルッカからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRFIM $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRFIM $w.d$ 出力形式では、金額をフィンランドマルッカからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRFIM $w.d$ 出力形式と EUROCURREN 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、フィンランドマルツカの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrfim5.;
put amount eurfrfim9.2;
datalines;
50
1234.56
12345
;
run;

8 put amount eurfrfim5.;
9 put amount eurfrfim9.2;
10 datalines;
E8
E8,41
E208
E207,64
2.076
E2.076,28
```

金額	ステートメント	結果
		----+-----1----2
50	put amount eurfrfim5.;	E8
	put amount eurfrfim9.2;	E8,41
1234.56	put amount eurfrfim5.;	E208
	put amount eurfrfim9.2;	E207,64
12345	put amount eurfrfim5.;	2.076
	put amount eurfrfim9.2;	E2.076,28

EURFRFRFw.d 出力形式

金額をフランスフランからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRFRFw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRFRF*w.d* 出力形式では、金額をフランスフランからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRFRF*w.d* 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ) を参照してください。

例

次の表に、フランスフランの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```

data _null_;
input amount;
put amount eurfrfrf5.;
put amount eurfrfrf9.2;
datalines;
50
1234.56
12345
;
run;
SAS log:
E8
E7,62
E188
E188,21
1.882
E1.881,98

```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrfrf5.;	E8
	put amount eurfrfrf9.2;	E7,62
1234.56	put amount eurfrfrf5.;	E188
	put amount eurfrfrf9.2;	E188,21
12345	put amount eurfrfrf5.;	1.882
	put amount eurfrfrf9.2;	E1.881,98

EURFRGBPw.d 出力形式

金額をイギリスポンドからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRGBPw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURFRGBPw.d 出力形式では、金額をイギリスポンドからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRGBPw.d 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ) を参照してください。

例

次の表に、イギリスポンドの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrgbp5.;
put amount eurfrgbp9.2;
datalines;
50
1234.56
12345
;
run;
SAS log:
3 put amount eurfrgbp5.;
4 put amount eurfrgbp9.2;
5 datalines;
E71
E71.42
1,763
E1,763.32
17632
17,632.39
```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrgbp5.;	E71
	put amount eurfrgbp9.2;	E71.42
1234.56	put amount eurfrgbp5.;	1,763
	put amount eurfrgbp9.2;	E1,763.32
12345	put amount eurfrgbp5.;	17632
	put amount eurfrgbp9.2;	17,632.39

EURFRGRD*w.d* 出力形式

金額をギリシャドラクマからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRGRD*w.d*

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRGRD*w.d* 出力形式では、金額をギリシャドラクマからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRGRD*w.d* 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ギリシャドラクマの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrgrd5.;
put amount eurfrgrd9.2;
datalines;
400
```

```

40234.56
402345
;
run;
SAS log:
3 put amount eurfrgrd5.;
4 put amount eurfrgrd9.2;
5 datalines;
E1
E1,17
E118
E118,03
1.180
E1.180,30

```

金額	ステートメント	結果
		----+-----1-----2
400	put amount eurfrgrd5.;	E1
	put amount eurfrgrd9.2;	E1,17
40234.56	put amount eurfrgrd5.;	E118
	put amount eurfrgrd9.2;	E118,03
402345	put amount eurfrgrd5.;	1.180
	put amount eurfrgrd9.2;	E1.180,30

EURFRHUFw.d 出力形式

金額をハンガリーフォリントからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRHUFw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRHUF $w.d$ 出力形式では、金額をハンガリーフォリントからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRHUF $w.d$ 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ) を参照してください。

例

次の表に、ハンガリーフォリントの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrhuf5.;
put amount eurfrhuf9.2;
datalines;
300
30234.56
302345
;
run;
SAS log:
3 put amount eurfrhuf5.;
4 put amount eurfrhuf9.2;
5 datalines;
E1
E1,15
E116
E116,14
1.161
E1.161,41
```

金額	ステートメント	結果
		----+-----1-----2
300	put amount eurfrhuf5.;	E1
	put amount eurfrhuf9.2;	E1,15
30234.56	put amount eurfrhuf5.;	E116
	put amount eurfrhuf9.2;	E116,14
302345	put amount eurfrhuf5.;	1.161
	put amount eurfrhuf9.2;	E1.161,41

EURFRIEP $w.d$ 出力形式

金額をアイルランドポンドからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRIEPw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURFRIEPw.d 出力形式では、金額をアイルランドポンドからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRIEPw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、アイルランドポンド入力値、SAS ステートメントおよびのユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfriep5.;
put amount eurfriep9.2;
datalines;
1
1234.56
12345
;
run;

8 put amount eurfriep5.;
9 put amount eurfriep9.2;
10 datalines;
E1
E1.27
1,568
E1,567.57
15675
15,674.92
```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurfriep5.;	E1
	put amount eurfriep9.2;	E1.27

金額	ステートメント	結果
1234.56	put amount eurfriep5.;	1,568
	put amount eurfriep9.2;	E1,567.57
12345	put amount eurfriep5.;	15675
	put amount eurfriep9.2;	15,674.92

EURFRITLw.d 出力形式

金額をイタリアリラからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRITLw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRITLw.d 出力形式では、金額をイタリアリラからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRITLw.d 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、イタリアリラの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfritl5.;
put amount eurfritl9.2;
datalines;
2000
7234.56
72345
;
run;
```

```

8 put amount eurfrit15.;
9 put amount eurfrit19.2;
10 datalines;
E1
E1,03
E4
E3,74
E37
E37,36

```

金額	ステートメント	結果
		----+-----1-----2
2000	put amount eurfrit15.;	E1
	put amount eurfrit19.2;	E1,03
7234.56	put amount eurfrit15.;	E4
	put amount eurfrit19.2;	E3,74
72345	put amount eurfrit15.;	E37
	put amount eurfrit19.2;	E37,36

EURFRLUFw.d 出力形式

金額をルクセンブルクフランからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRLUFw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

The EURFRLUFw.d 出力形式では、金額をルクセンブルクフランからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRLUFw.d 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ルクセンブルクフランの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrluf5.;
put amount eurfrluf9.2;
datalines;
50
1234.56
12345
;
run;

8 put amount eurfrluf5.;
9 put amount eurfrluf9.2;
10 datalines;
E1
E1,24
E31
E30,60
E306
E306,02
```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrluf5.;	E1
	put amount eurfrluf9.2;	E1,24
1234.56	put amount eurfrluf5.;	E31
	put amount eurfrluf9.2;	E30,60
12345	put amount eurfrluf5.;	E306
	put amount eurfrluf9.2;	E306,02

EURFRNLGw.d 出力形式

金額をオランダギルダからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRNLGw.d

説明

w
出力フィールドの幅を指定します。
デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURFRNLGw.d 出力形式では、金額をオランダギルダーからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRNLGw.d 出力形式と EUROCRR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、オランダギルダーの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrnl5.;
put amount eurfrnl9.2;
datalines;
50
1234.56
12345
;
run;

8 put amount eurfrnl5.;
9 put amount eurfrnl9.2;
10 datalines;
E23
E22,69
E560
E560,22
5.602
E5.601,92
```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrnl5.;	E23
	put amount eurfrnl9.2;	E22,69
1234.56	put amount eurfrnl5.;	E560
	put amount eurfrnl9.2;	E560,22
12345	put amount eurfrnl5.;	5.602
	put amount eurfrnl9.2;	E5.601,92

EURFRNOK $w.d$ 出力形式

金額をノルウェークローネからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRNOK $w.d$

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURFRNOK $w.d$ 出力形式では、金額をノルウェークローネからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRNOK $w.d$ 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ノルウェークローネの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrnok5.;
put amount eurfrnok9.2;
datalines;
50
1234.56
12345
;
run;
SAS log:
3 put amount eurfrnok5.;
4 put amount eurfrnok9.2;
5 datalines;
E5
E5,44
E134
E134,22
1.342
E1.342,18
```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrnok5.;	E5
	put amount eurfrnok9.2;	E5,44
1234.56	put amount eurfrnok5.;	E134
	put amount eurfrnok9.2;	E134,22
12345	put amount eurfrnok5.;	1.342
	put amount eurfrnok9.2;	E1.342,18

EURFRPLZw.d 出力形式

金額をポーランドズウォティからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRPLZw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRPLZw.d 出力形式では、金額をポーランドズウォティからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRPLZw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ) を参照してください。

例

次の表に、ポーランドズウォティの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrplz5.;
put amount eurfrplz9.2;
datalines;
```

```

50
1234.56
12345
;
run;
SAS log:
3 put amount eurfrplz5.;
4 put amount eurfrplz9.2;
5 datalines;
E12
E11,90
E294
E293,94
2.939
E2.939,29

```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrplz5.;	E12
	put amount eurfrplz9.2;	E11,90
1234.56	put amount eurfrplz5.;	E294
	put amount eurfrplz9.2;	E293,94
12345	put amount eurfrplz5.;	2.939
	put amount eurfrplz9.2;	E2.939,29

EURFRPTE $w.d$ 出力形式

金額をポルトガルエスクードからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRPTE $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRPTEw.d 出力形式では、金額をポルトガルエスクードからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは固定レートで、EURFRPTEw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ) を参照してください。

例

次の表に、ポルトガルエスクードの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrpte5.;
put amount eurfrpte9.2;
datalines;
300
30234.56
302345
;
run;

8 put amount eurfrpte5.;
9 put amount eurfrpte9.2;
10 datalines;
E1
E1,50
E151
E150,81
1.508
E1.508,09
```

金額	ステートメント	結果
		----+-----1-----2
300	put amount eurfrpte5.;	E1
	put amount eurfrpte9.2;	E1,50
30234.56	put amount eurfrpte5.;	E151
	put amount eurfrpte9.2;	E150,81
302345	put amount eurfrpte5.;	1.508
	put amount eurfrpte9.2;	E1.508,09

EURFRROLw.d 出力形式

金額をルーマニアレイからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRROL $w.d$

説明

 w
出力フィールドの幅を指定します。

デフォルト: 6

 d
数値の小数点以下の桁数を指定します。

詳細

EURFRROL $w.d$ 出力形式では、金額をルーマンレイからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRROL $w.d$ 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ) を参照してください。

例

次の表に、ルーマンレイの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrrol5.;
put amount eurfrrol9.2;
datalines;
50
5234.56
52345
;
run;

E4
E3,65
E382
E381,81
3.818
E3.818,02
```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrrol5.;	E4
	put amount eurfrrol9.2;	E3,65
5234.56	put amount eurfrrol5.;	E382
	put amount eurfrrol9.2;	E381,81

金額	ステートメント	結果
52345	put amount eurfrrol5.;	3.818
	put amount eurfrrol9.2;	E3.818,02

EURFRRURw.d 出力形式

金額をロシアルーブルからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRRURw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURFRRURw.d 出力形式では、金額をロシアルーブルからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRRURw.d 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ) を参照してください。

例

次の表に、ロシアルーブルの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrrrur5.;
put amount eurfrrrur9.2;
datalines;
50
5234.56
52345
;
run;

E3
E2,53
E265
```

E264,80
2.648
E2.647,97

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrur5.;	E3
	put amount eurfrur9.2;	E2,53
5234.56	put amount eurfrur5.;	E265
	put amount eurfrur9.2;	E264,80
52345	put amount eurfrur5.;	2.648
	put amount eurfrur9.2;	E2.647,97

EURFRSEK $w.d$ 出力形式

金額をスウェーデンクローナからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRSEK $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURFRSEK $w.d$ 出力形式では、金額をスウェーデンクローナからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRSEK $w.d$ 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ) を参照してください。

例

次の表に、スウェーデンクローナの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```

data _null_;
input amount;
put amount eurfrsek5.;
put amount eurfrsek9.2;
datalines;
50
1234.56
12345
;
run;

E5
E5,34
E132
E131,81
1.318
E1.318,08

```

金額	ステートメント	結果
		----+-----1-----2
50	put amount eurfrsek5.;	E5
	put amount eurfrsek9.2;	E5,34
1234.56	put amount eurfrsek5.;	E132
	put amount eurfrsek9.2;	E131,81
12345	put amount eurfrsek5.;	1.318
	put amount eurfrsek9.2;	E1.318,08

EURFRSITw.d 出力形式

金額をスロヴェニアトラからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRSITw.d

説明

w 出カフィールドの幅を指定します。

デフォルト: 6

d 数値の小数点以下の桁数を指定します。

詳細

EURFRSIT $w.d$ 出力形式では、金額をスロヴェニアトラーからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRSIT $w.d$ 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ)を参照してください。

注: スロベニアの通貨はユーロです。EURFRSIT の情報は、ユーザーの履歴データ用に提供されています。

例

次の表に、スロヴェニアトラーの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrsit5.;
put amount eurfrsit9.2;
datalines;
200
20234.56
202345
;
run;

E1
E1,05
E106
E105,94
1.059
E1.059,40
```

金額	ステートメント	結果
		----+-----1-----2
200	put amount eurfrsit5.;	E1
	put amount eurfrsit9.2;	E1,05
20234.56	put amount eurfrsit5.;	E106
	put amount eurfrsit9.2;	E105,94
202345	put amount eurfrsit5.;	1.059
	put amount eurfrsit9.2;	E1.059,40

EURFRTRL $w.d$ 出力形式

金額をトルコリラからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRTLW.d

説明

w
出力フィールドの幅を指定します。
デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURFRTLW.d 出力形式では、金額をトルコリラからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRTLW.d 出力形式と EUROCRR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ) を参照してください。

例

次の表に、トルコリラの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfrtrl5.;
put amount eurfrtrl9.2;
datalines;
400
40234.56
402345
;
run;

E1
E1,19
E119
E119,42
1.194
E1.194,21
```

金額	ステートメント	結果
		----+-----1-----2
400	put amount eurfrtrl5.;	E1
	put amount eurfrtrl9.2;	E1,19
40234.56	put amount eurfrtrl5.;	E119
	put amount eurfrtrl9.2;	E119,42

金額	ステートメント	結果
402345	put amount eurfrtrl5.;	1.194
	put amount eurfrtrl9.2;	E1.194,21

EURFRYUD $w.d$ 出力形式

金額をユーゴスラビア ディーナールからユーロに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURFRYUD $w.d$

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURFRYUD $w.d$ 出力形式では、金額をユーゴスラビア ディーナールからユーロに換算し、出力形式を適用したユーロの値を表示します。換算レートは変更可能なレートで、EURFRYUD $w.d$ 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、「通貨表現」(61 ページ)を参照してください。

例

次の表に、ユーゴスラビア ディーナールの入力値、SAS ステートメントおよびユーロの換算結果を示します。

```
data _null_;
input amount;
put amount eurfryud5.;
put amount eurfryud9.2;
datalines;
50
5234.56
52345
;
run;

E4
E3,83
E401
```


E400,67
4.007
E4.006,69

金額	ステートメント	結果
		----+----1----2
50	put amount eur fryud5.;	E4
	put amount eur fryud9.2;	E3,83
5234.56	put amount eur fryud5.;	E401
	put amount eur fryud9.2;	E400,67
52345	put amount eur fryud5.;	4.007
	put amount eur fryud9.2;	E4.006,69

EURTOATSw.d 出力形式

金額をユーロからオーストリアシリングに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOATSw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURTOATSw.d 出力形式では、ユーロの金額をオーストリアシリングに換算します。換算レートは固定レートで、EURTOATSw.d 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびオーストリアシリングの換算結果を示します。

```
data _null_;
input amount;
```

```

put amount eurtoats6.;
put amount eurtoats12.2;
datalines;
1
1234.56
12345
;
run;

80 put amount eurtoats6.;
81 put amount eurtoats12.2;
82 datalines;
14
13.76
16988
16987.92
169871
169870.90

```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtoats6.;	14
	put amount eurtoats12.2;	13.76
1234.56	put amount eurtoats6.;	16988
	put amount eurtoats12.2;	16987.92
12345	put amount eurtoats6.;	169871
	put amount eurtoats12.2;	169870.90

EURTOBEF $w.d$ 出力形式

金額をユーロからベルギーフランに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOBEF $w.d$

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOBEFw.d 出力形式では、ユーロの金額をベルギーフランに換算します。換算レートは固定レートで、EURTOBEFw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびベルギーフランの換算結果を示します。

```
data _null_;
input amount;
put amount eurtobef6.;
put amount eurtobef12.2;
datalines;
1
1234.56
12345
;
run;

8 put amount eurtobef6.;
9 put amount eurtobef12.2;
10 datalines;
40
40.34
49802
49802.03
497996
497996.07
```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtobef6.;	40
	put amount eurtobef12.2;	40.34
1234.56	put amount eurtobef6.;	49802
	put amount eurtobef12.2;	49802.03
12345	put amount eurtobef6.;	497996
	put amount eurtobef12.2;	497996.07

EURTOCHFw.d 出力形式

金額をユーロからスイスフランに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOCHF $w.d$

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOCHF $w.d$ 出力形式では、ユーロの金額をスイスフランに換算します。換算レートは変更可能なレートで、EURTOCHF $w.d$ 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびスイスフランの換算結果を示します。

```

data _null_;
input amount;
put amount eurtochf6.;
put amount eurtochf12.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
8 put amount eurtochf6.;
9 put amount eurtochf12.2;
10 datalines;
2
1.60
1981
1980.60
19805
19805.08

```

金額	ステートメント	結果
		----+----1----2
1	put amount eurtochf6.;	2
	put amount eurtochf12.2.;	1.60

金額	ステートメント	結果
1234.56	put amount eurtochf6.;	1981
	put amount eurtochf12.2;	1980.60
12345	put amount eurtochf6.;	19805
	put amount eurtochf12.2;	19805.08

EURTOCZKw.d 出力形式

金額をユーロからチェココルナに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOCZKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURTOCZKw.d 出力形式では、ユーロの金額をチェココルナに換算します。換算レートは変更可能なレートで、EURTOCZKw.d 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびチェココルナの換算結果を示します。

```
data _null_;
input amount;
put amount eurtochk6.;
put amount eurtochk12.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
```

```

104 put amount eurtoczk6.;
105 put amount eurtoczk12.2;
106 datalines;
35
34.86
43032
43032.19
430301
430301.02

```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtoczk6.;	35
	put amount eurtoczk12.2;	34.86
1234.56	put amount eurtoczk6.;	43032
	put amount eurtoczk12.2;	43032.19
12345	put amount eurtoczk6.;	430301
	put amount eurtoczk12.2;	430301.02

EURTODEM $w.d$ 出力形式

金額をユーロからドイツマルクに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTODEM $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURTODEM $w.d$ 出力形式では、ユーロの金額をドイツマルクに換算します。換算レートは固定レートで、EURTODEM $w.d$ 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびドイツマルクの換算結果を示します。

```
data _null_;
input amount;
put amount eurtodem6.;
put amount eurtodem12.2;
datalines;
1
1234.56
12345
;
run;

8 put amount eurtodem6.;
9 put amount eurtodem12.2;
10 datalines;
2
1.96
2415
2414.59
24145
24144.72
```

金額	ステートメント	結果
		----+----1----2
1	put amount eurtodem6.;	2
	put amount eurtodem12.2;	1.96
1234.56	put amount eurtodem6.;	2415
	put amount eurtodem12.2;	2414.59
12345	put amount eurtodem6.;	24145
	put amount eurtodem12.2;	24144.72

EURTODKKw.d 出力形式

金額をユーロからデンマーククローネに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTODKKw.d

説明

w
出力フィールドの幅を指定します。
デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTODKK*w.d* ユーロの金額をデンマーククローネに換算します。換算レートは変更可能なレートで、EURTODKK*w.d* 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびデンマーククローネの換算結果を示します。

```
data _null_;
input amount;
put amount eurtodkk6.;
put amount eurtodkk12.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
62 put amount eurtodkk6.;
63 put amount eurtodkk12.2;
64 datalines;
7
7.49
9247
9246.97
92465
92465.16
```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtodkk6.;	7
	put amount eurtodkk12.2;	7.49
1234.56	put amount eurtodkk6.;	9247
	put amount eurtodkk12.2;	9246.97
12345	put amount eurtodkk6.;	92465
	put amount eurtodkk12.2;	92465.16

EURTOESPw.d 出力形式

金額をユーロからスペインペセタに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOESPw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOESPw.d 出力形式では、ユーロの金額をスペインペセタに換算します。換算レートは固定レートで、EURTOESPw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびスペインペセタの換算結果を示します。

```
data _null_;
input amount;
put amount eurtoesp8.;
put amount eurtoesp12.2;
datalines;
1
1234.56
12345
;
run;

26 put amount eurtoesp8.;
27 put amount eurtoesp12.2;
28 datalines;
166
166.39
205414
205413.50
2054035
2054035.17
```

金額	ステートメント	結果
		----+----1----2
1	put amount eurtoesp8.;	166
	put amount eurtoesp12.2.;	166.39
1234.56	put amount eurtoesp8.;	205414
	put amount eurtoesp12.2.;	205413.50
12345	put amount eurtoesp8.;	2054035
	put amount eurtoesp12.2.;	2054035.17

EURTOFIM $w.d$ 出力形式

金額をユーロからフィンランドマルツカに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOFIM $w.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURTOFIM $w.d$ 出力形式では、ユーロの金額をフィンランドマルツカに換算します。換算レートは固定レートで、EURTOFIM $w.d$ 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびフィンランドマルツカの換算結果を示します。

```
data _null_;
input amount;
put amount eurtofim6.;
put amount eurtofim12.2.;
datalines;
1
```

```

1234.56
12345
;
run;

8 put amount eurtofim6.;
9 put amount eurtofim12.2;
10 datalines;
6
5.95
7340
7340.36
73400
73400.04

```

金額	ステートメント	結果
		----+----1----2
1	put amount eurtofim6.;	6
	put amount eurtofim12.2;	5.95
1234.56	put amount eurtofim6.;	7340
	put amount eurtofim12.2;	7340.36
12345	put amount eurtofim6.;	73400
	put amount eurtofim12.2;	73400.04

EURTOFRFw.d 出力形式

金額をユーロからフランスフランに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOFRFw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOFRFw.d 出力形式では、ユーロの金額をフランスフランに換算します。換算レートは固定レートで、EURTOFRFw.d 出力形式と EUROCURRE 関数に組み込まれて

います。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびフランスフランの換算結果を示します。

```
data _null_;
input amount;
put amount eurtofrf6.;
put amount eurtofrf12.2;
datalines;
1
1234.56
12345
;
run;

8 put amount eurtofrf6.;
9 put amount eurtofrf12.2;
10 datalines;
7
6.56
8098
8098.18
80978
80977.89
```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtofrf6.;	7
	put amount eurtofrf12.2.;	6.56
1234.56	put amount eurtofrf6.;	8098
	put amount eurtofrf12.2.;	8098.18
12345	put amount eurtofrf6.;	80978
	put amount eurtofrf12.2.;	80977.89

EURTOGBPw.d 出力形式

金額をユーロからイギリスポンドに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOGBPw.d

説明

w
出力フィールドの幅を指定します。
デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOGBPw.d 出力形式では、ユーロの金額をイギリスポンドに換算します。換算レートは変更可能なレートで、EURTOGBPw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびイギリスポンドの換算結果を示します。

```
data _null_;
input amount;
put amount eurtogbp6.;
put amount eurtogbp12.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
8 put amount eurtogbp6.;
9 put amount eurtogbp12.2;
10 datalines;
1
0.70
864
864.35
8643
8643.13
```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtogbp6.;	1
	put amount eurtogbp12.2;	0.70
1234.56	put amount eurtogbp6.;	864
	put amount eurtogbp12.2;	864.35

金額	ステートメント	結果
12345	put amount eurtogbp6.;	8643
	put amount eurtogbp12.2;	8643.13

EURTOGRD $w.d$ 出力形式

金額をユーロからギリシャドラクマに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOGRD $w.d$

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOGRD $w.d$ 出力形式では、ユーロの金額をギリシャドラクマに換算します。換算レートは固定レートで、EURTOGRD $w.d$ 出力形式と EUROCURREN 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびギリシャドラクマの換算結果を示します。

```
data _null_;
input amount;
put amount eurtogrd8.;
put amount eurtogrd16.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
65 put amount eurtogrd8.;
66 put amount eurtogrd16.2;
67 datalines;
341
```

340.89
 420843
 420842.99
 4208225
 4208225.33

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtogrd8.;	341
	put amount eurtogrd16.2;	340.89
1234.56	put amount eurtogrd8.;	420843
	put amount eurtogrd16.2;	420842.99
12345	put amount eurtogrd8.;	4208225
	put amount eurtogrd16.2;	4208225.33

EURTOHUFw.d 出力形式

金額をユーロからハンガリーフォリントに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOHUFw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURTOHUFw.d 出力形式では、ユーロの金額をハンガリーフォリントに換算します。換算レートは変更可能なレートで、EURTOHUFw.d 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ) を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびハンガリーフォリントの換算結果を示します。

```

data _null_;
input amount;
put amount eurtohuf8.;
put amount eurtohuf14.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
140 put amount eurtohuf8.;
141 put amount eurtohuf14.2;
142 datalines;
260
260.33
321387
321386.83
3213712
3213712.13

```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtohuf8.;	260
	put amount eurtohuf14.2;	260.33
1234.56	put amount eurtohuf8.;	321387
	put amount eurtohuf14.2;	321386.83
12345	put amount eurtohuf8.;	3213712
	put amount eurtohuf14.2;	3213712.13

EURTOIEPw.d 出力形式

金額をユーロからアイルランドポンドに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOIEPw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURTOIEP*w.d* 出力形式ではユーロの金額をアイルランドポンドに換算します。換算レートは固定レートで、EURTOIEP*w.d* 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびアイルランドポンドの換算結果を示します。

```
data _null_;
input amount;
put amount eurtoiep6.;
put amount eurtoiep12.2;
datalines;
1
1234.56
12345
;
run;

8 put amount eurtoiep6.;
9 put amount eurtoiep12.2;
10 datalines;
1
0.79
972
972.30
9722
9722.48
```

金額	ステートメント	結果
		----+----1----2
1	put amount eurtoiep6.;	1
	put amount eurtoiep12.2.;	0.79
1234.56	put amount eurtoiep6.;	972
	put amount eurtoiep12.2.;	972.30
12345	put amount eurtoiep6.;	9722
	put amount eurtoiep12.2.;	9722.48

EURTOITLw.d 出力形式

金額をユーロからイタリアリラに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOITL $w.d$

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOITL $w.d$ 出力形式では、ユーロの金額をイタリアリラに換算します。換算レートは固定レートで、EURTOITL $w.d$ 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびイタリアリラの換算結果を示します。

```
data _null_;
input amount;
put amount eurtoitl8.;
put amount eurtoitl12.2;
datalines;
1
1234.56
12345
;
run;

44 put amount eurtoitl8.;
45 put amount eurtoitl12.2;
46 datalines;
1936
1936.27
2390441
2390441.49
23903253
23903253.15
```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtoitl8.;	1936
	put amount eurtoitl12.2.;	1936.27

金額	ステートメント	結果
1234.56	put amount eurtoit18.;	2390441
	put amount eurtoit112.2;	2390441.49
12345	put amount eurtoit18.;	23903253
	put amount eurtoit112.2;	23903253.15

EURTOLUFw.d 出力形式

金額をユーロからルクセンブルクフランに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOLUFw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURTOLUFw.d 出力形式では、ユーロの金額をルクセンブルグフランに換算します。換算レートは固定レートで、EURTOLUFw.d 出力形式と EUROCURRE 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびルクセンブルクフランの換算結果を示します。

```
data _null_;
input amount;
put amount eurtoluf6.;
put amount eurtoluf12.2;
datalines;
1
1234.56
12345
;
run;
```

```

8 put amount eurtoluf6.;
9 put amount eurtoluf12.2;
10 datalines;
40
40.34
49802
49802.03
497996
497996.07

```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtoluf6.;	40
	put amount eurtoluf12.2;	40.34
1234.56	put amount eurtoluf6.;	49802
	put amount eurtoluf12.2;	49802.03
12345	put amount eurtoluf6.;	497996
	put amount eurtoluf12.2;	497996.07

EURTONLGw.d 出力形式

金額をユーロからオランダギルダーに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTONLGw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTONLGw.d 出力形式では、ユーロの金額をオランダギルダーに換算します。換算レートは固定レートで、EURTONLGw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびオランダギルダーの換算結果を示します。

```
data _null_;
input amount;
put amount eurtonlg6.;
put amount eurtonlg12.2;
datalines;
1
1234.56
12345
;
run;

8 put amount eurtonlg6.;
9 put amount eurtonlg12.2;
10 datalines;
2
2.20
2721
2720.61
27205
27204.80
```

金額	ステートメント	結果
		----+----1----2
1	put amount eurtonlg6.;	2
	put amount eurtonlg12.2;	2.20
1234.56	put amount eurtonlg6.;	2721
	put amount eurtonlg12.2;	2720.61
12345	put amount eurtonlg6.;	27205
	put amount eurtonlg12.2;	27204.80

EURTONOKw.d 出力形式

金額をユーロからノルウェークローネに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTONOKw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURTONOK $w.d$ 出力形式では、ユーロの金額をノルウェークローネに換算します。換算レートは変更可能なレートで、EURTONOK $w.d$ 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ) を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびノルウェークローネの換算結果を示します。

```

data _null_;
input amount;
put amount eurtonok6.;
put amount eurtonok12.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
158 put amount eurtonok6.;
159 put amount eurtonok12.2;
160 datalines;
9
9.20
11355
11355.11
113546
113545.61

```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtonok6.;	9
	put amount eurtonok12.2;	9.20
1234.56	put amount eurtonok6.;	11355
	put amount eurtonok12.2;	11355.11
12345	put amount eurtonok6.;	113546
	put amount eurtonok12.2;	113545.61

EURTOPLZw.d 出力形式

金額をユーロからポーランドズウォティに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOPLZw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOPLZw.d 出力形式では、ユーロの金額をポーランドズウォティに換算します。換算レートは変更可能なレートで、EURTOPLZw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびポーランドズウォティの換算結果を示します。

```
data _null_;
input amount;
put amount eurtoplz6.;
put amount eurtoplz12.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
80 put amount eurtoplz6.;
81 put amount eurtoplz12.2;
82 datalines;
4
4.20
5185
5185.15
51849
51849.00
```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtoplz6.;	4
	put amount eurtoplz12.2;	4.20
1234.56	put amount eurtoplz6.;	5185
	put amount eurtoplz12.2;	5185.15
12345	put amount eurtoplz6.;	51849
	put amount eurtoplz12.2;	51849.00

EURTOPT $ew.d$ 出力形式

金額をユーロからポルトガルエスクードに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOPT $ew.d$

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURTOPT $ew.d$ 出力形式では、ユーロの金額をポルトガルエスクードに換算します。換算レートは固定レートで、EURTOPT $ew.d$ 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびポルトガルエスクードの換算結果を示します。

```
data _null_;
input amount;
put amount eurtopte8.;
put amount eurtopte12.2;
datalines;
1
```



```

1234.56
12345
;
run;

26 put amount eurtopte8.;
27 put amount eurtopte12.2;
28 datalines;
200
200.48
247507
247507.06
2474950
2474950.29

```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtopte8.;	200
	put amount eurtopte12.2.;	200.48
1234.56	put amount eurtopte8.;	247507
	put amount eurtopte12.2.;	247507.06
12345	put amount eurtopte8.;	2474950
	put amount eurtopte12.2.;	2474950.29

EURTOROLw.d 出力形式

金額をユーロからルーマニアレイに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOROLw.d

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOROLw.d 出力形式では、ユーロの金額をルーマニアレイに換算します。換算レートは変更可能なレートで、EURTOROLw.d 出力形式と EUROCURR 関数に組み

含まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびルーマニアレイの換算結果を示します。

```
data _null_;
input amount;
put amount eurtorol6.;
put amount eurtorol12.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
98 put amount eurtorol6.;
99 put amount eurtorol12.2;
100 datalines;
14
13.71
16926
16925.82
169250
169249.95
```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtorol6.;	14
	put amount eurtorol12.2.;	13.71
1234.56	put amount eurtorol6.;	16926
	put amount eurtorol12.2.;	16925.82
12345	put amount eurtorol6.;	169250
	put amount eurtorol12.2.;	169249.95

EURTORURw.d 出力形式

金額をユーロからロシアルーブルに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTORURw.d

説明

w
出力フィールドの幅を指定します。
デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTORURw.d 出力形式では、ユーロの金額をロシアルーブルに換算します。換算レートは変更可能なレートで、EURTORURw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびロシアルーブルの換算結果を示します。

```
data _null_;
input amount;
put amount eurtorur6.;
put amount eurtorur12.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
8 put amount eurtorur6.;
9 put amount eurtorur12.2;
10 datalines;
20
19.77
24405
24404.78
244036
244035.96
```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtorur6.;	20
	put amount eurtorur12.2;	19.77
1234.56	put amount eurtorur6.;	24405
	put amount eurtorur12.2;	24404.78

金額	ステートメント	結果
12345	put amount eurtorur6.;	244036
	put amount eurtorur12.2;	244035.96

EURTOSEK $w.d$ 出力形式

金額をユーロからスウェーデンクローナに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOSEK $w.d$

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOSEK $w.d$ 出力形式では、ユーロの金額をスウェーデンクローナに換算します。換算レートは変更可能なレートで、EURTOSEK $w.d$ 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびスウェーデンクローナの換算結果を示します。

```
data _null_;
input amount;
put amount eurtosek6.;
put amount eurtosek12.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
86 put amount eurtosek6.;
87 put amount eurtosek12.2;
88 datalines;
9
```

9.37
 11563
 11562.78
 115622
 115622.16

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtosek6.;	9
	put amount eurtosek12.2;	9.37
1234.56	put amount eurtosek6.;	11563
	put amount eurtosek12.2;	11562.78
12345	put amount eurtosek6.;	115622
	put amount eurtosek12.2;	115622.16

EURTOSITw.d 出力形式

金額をユーロからスロヴェニアトラーに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOSITw.d

説明

w

出力フィールドの幅を指定します。

デフォルト: 6

d

数値の小数点以下の桁数を指定します。

詳細

EURTOSITw.d 出力形式では、ユーロの金額をスロヴェニアトラーに換算します。換算レートは変更可能なレートで、EURTOSITw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

注: スロベニアの通貨はユーロです。EURTOSIT の情報は、ユーザーの履歴データ用に提供されています。

例

次の表に、ユーロの入力値、SAS ステートメントおよびスロヴェニアトラの換算結果を示します。

```
data _null_;
input amount;
put amount eurtosit8.;
put amount eurtosit14.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
152 put amount eurtosit8.;
153 put amount eurtosit14.2;
154 datalines;
191
191.00
235801
235800.96
2357895
2357895.00
```

金額	ステートメント	結果
		----+----1----2
1	put amount eurtosit8.;	191
	put amount eurtosit14.2;	191.00
1234.56	put amount eurtosit8.;	235801
	put amount eurtosit14.2;	235800.96
12345	put amount eurtosit8.;	2357895
	put amount eurtosit14.2;	2357895.00

EURTOTRLw.d 出力形式

金額をユーロからトルコリラに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOTRLw.d

説明

w
出力フィールドの幅を指定します。
デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOTRLw.d 出力形式では、ユーロの金額をトルコリラに換算します。換算レートは変更可能なレートで、EURTOTRLw.d 出力形式と EUROCURR 関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“[通貨表現](#)” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびトルコリラの換算結果を示します。

```
data _null_;
input amount;
put amount eurtotrl8.;
put amount eurtotrl14.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
62 put amount eurtotrl8.;
63 put amount eurtotrl14.2;
64 datalines;
337
336.91
415938
415938.08
4159179
4159178.64
```

金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtotrl8.;	337
	put amount eurtotrl14.2;	336.91
1234.56	put amount eurtotrl8.;	415938
	put amount eurtotrl14.2;	415938.08
12345	put amount eurtotrl8.;	4159179
	put amount eurtotrl14.2;	4159178.64

EURTOYUD $w.d$ 出力形式

金額をユーロからユーゴスラビア ディーナールに換算します。

カテゴリ: 通貨の換算

配置: 右

構文

EURTOYUD $w.d$

説明

w
出力フィールドの幅を指定します。

デフォルト: 6

d
数値の小数点以下の桁数を指定します。

詳細

EURTOYUD $w.d$ 出力形式では、ユーロの金額をユーゴスラビア ディーナールに換算します。換算レートは変更可能なレートで、EURTOYUD $w.d$ 出力形式と EUROCURRE関数に組み込まれています。ヨーロッパ通貨換算と通貨換算レートテーブルの詳細については、“通貨表現” (61 ページ)を参照してください。

例

次の表に、ユーロの入力値、SAS ステートメントおよびユーゴスラビ ディーナールの換算結果を示します。

```
data _null_;
input amount;
put amount eurtoyud6.;
put amount eurtoyud12.2;
datalines;
1
1234.56
12345
;
run;
SAS log:
116 put amount eurtoyud6.;
117 put amount eurtoyud12.2;
118 datalines;
13
13.06
16129
16128.79
161280
161280.02
```


金額	ステートメント	結果
		----+-----1-----2
1	put amount eurtoyud6.;	13
	put amount eurtoyud12.2;	13.06
1234.56	put amount eurtoyud6.;	16129
	put amount eurtoyud12.2;	16128.79
12345	put amount eurtoyud6.;	161280
	put amount eurtoyud12.2;	161280.02

EURDFDEw. 入力形式

指定したヨーロッパ言語で日付値を読み込みます。

カテゴリ: 日付と時間

構文

EURDFDEw.

必須引数

w

入力幅を指定します。

デフォルト: 7 (フィンランド語以外)

範囲: 7–32 (フィンランド語以外)

注: フィンランド語(FIN)の接頭辞を使用する場合、w の値の範囲は 10–32 でデフォルトの w 値は 10 です。

詳細

日付値は *ddmmmyy* または *ddmmmyyyy* 形式にする必要があります。

dd

月の日を表す 01 から 31 の整数です。

mmm

月名の最初の 3 文字になります。

yy または *yyyy*

年を表す 2 桁または 4 桁の整数です。

日、月、年の値の間には、空白やその他の特殊文字を使用できます。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“DFLANG=システムオプション: UNIX、Windows および z/OS” (464 ページ) を参照してください。入力

形式で言語接頭辞を指定した場合は、DFLANG=システムオプションは無視されません。

例

この INPUT ステートメントでは、DFLANG=システムオプションの値を使用して、スペイン語で日付値を読み込みます。

```
options dflang=spanish;
input day eurdfde10.;
```

この INPUT ステートメントでは、スペイン語の接頭辞を含む入力形式を使用して、スペイン語で SAS 日付値を読み込みます。したがって、DFLANG=オプションの値は無視されます。

```
input day espdfde10.;;

options dflang=spanish;
data _null_;
input day eurdfde10.;;
put day;
datalines;
01abr1999
01-abr-99
;
```

値	結果
	----+-----1
01abr1999	14335
01-abr-99	14335

EURDFDTw. 入力形式

指定したヨーロッパ言語で、*ddmmyy hh:mm:ss.ss* または *ddmmyyyy hh:mm:ss.ss* 形式の日時値を読み込みます。

カテゴリ: 日付と時間

構文

EURDFDT*w*.

説明

w
入力幅を指定します。

デフォルト: 18

範囲: 13–40

詳細

日付値は *ddmmmyy* または *ddmmmyyyy* 形式にし、その後空白または特殊文字を挿入し、次に *hh:mm:ss.ss* の時間値を追加する必要があります。日付の構文を次に示します。

dd
月の日を表す 01 から 31 の整数です。

mmm
月名の最初の 3 文字になります。

yy または *yyyy*
年を表す 2 桁または 4 桁の整数です。

時間の構文を次に示します。

hh
00 から 23 の範囲の時間数です。

mm
00 から 59 の範囲の分数です。

ss.ss
小数点以下の秒の端数を含む、00 から 59 の範囲の秒数です。

EURDFDTw 入力形式には、日付と時間の両方の値が必須です。ただし、*ss.ss* の部分はオプションです。

注: SAS では、2 桁の年は YEARCUTOFF=システムオプションで定義された 100 年の期間内であると解釈します。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“[DFLANG=システムオプション: UNIX、Windows および z/OS](#)” (464 ページ) を参照してください。入力形式で言語接頭辞を指定した場合は、DFLANG=システムオプションは無視されません。

例

この INPUT ステートメントでは、DFLANG=システムオプションの値を使用して、ドイツ語で日時値を読み込みます。

```
options dflang=german;
input date eurdfd20.;
```

この INPUT ステートメントでは、ドイツ語の接頭辞を含む入力形式を使用して、ドイツ語で SAS 日時値を読み込みます。したがって、DFLANG=オプションの値は無視されます。

```
input date deufdt20. ;

options dflang=german;
data _null_;
input date eurdfd20. ;
put date;
datalines;
23dez99:10:03:17.2
23dez1999:10:03:17.2
;
```

値	結果
	----+----1----+----2
23dez99:10:03:17.2	1261562597.2
23dez1999:10:03:17.2	1261562597.2

EURDFMY w . 入力形式

$mmmyy$ または $mmmyyyy$ 形式の月と年の日付値を読み込みます。

カテゴリ: 日付と時間

構文

EURDFMY w .

説明

w

入力幅を指定します。

デフォルト: 5 (フィンランド語以外)

範囲: 5–32 (フィンランド語以外)

注: フィンランド語(FIN)の接頭辞を使用する場合、 w の値の範囲は 7–32 でデフォルトの w 値は 7 です。

詳細

日付値は $mmmyy$ または $mmmyyyy$ 形式にする必要があります。

mmm

月名の最初の 3 文字になります。

yy または $yyyy$

年を表す 2 桁または 4 桁の整数です。

日、月、年の値の間には、空白やその他の特殊文字を使用できます。EURDFMY w . によって読み込まれる値は、指定した月の第 1 日に相当する SAS 日付値になります。

注: SAS では、2 桁の年は YEARCUTOFF=システムオプションで定義された 100 年の期間内であると解釈します。

DFLANG=システムオプションを使用して、SAS セッションの言語を設定できます (通常は SAS インストール担当者によってサイトのデフォルト言語が設定されるため、このステップは省略できることがあります)。複数言語で日付を取り扱う場合、EUR 接頭辞を言語接頭辞に置換できます。言語接頭辞のリストについては、“DFLANG=システムオプション: UNIX、Windows および z/OS” (464 ページ) を参照してください。入力形式で言語接頭辞を指定した場合、DFLANG=オプションは無視されます。

例

この INPUT ステートメントでは、DFLANG=システムオプションの値を使用して、フランス語で日付値を読み込みます。

```
options dflang=french;
input month eurdfmy7.;
```

2 番目の INPUT ステートメントでは、フランス語の言語接頭辞を使用し、DFLANG は指定しません。

```
input month fradfmy7.;
```

```
options dflang=french;
data _null_;
input month eurdfmy7.;
```

```
put month;
datalines;
avr1999
avr 99
;
```

```
options dflang=english;
data _null_;
input month fradfmy7.;
```

```
put month;
datalines;
avr1999
avr 99
;
```

値	結果
	----+-----1
avr1999	14335
avr 99	14335

EUROCURRE 関数

1 つのヨーロッパ通貨から別のヨーロッパ通貨に変換します。

カテゴリ: 通貨の換算

構文

EUROCURRE(*from-currency-amount*, *from-currency-code*, *to-currency-code*)

必須引数

from-currency-amount

変換する金額を指定する数値。

from-currency-code

変換元の通貨を識別する3文字の通貨コードを指定します(ヨーロッパ通貨と通貨コード(666ページ)を参照してください。)

ヒント: *from-currency-code* の値が空白の場合、EUROCURR は、ユーロから指定されたヨーロッパの国の通貨に通貨値を変換します。

参照項目: “例 4: 1 つの変数が空白の場合の通貨の変換”(668ページ)

to-currency-code

変換先の通貨を識別する3文字の通貨コードを指定します(ヨーロッパ通貨と通貨コード(666ページ)を参照してください。)

ヒント: *to-currency-code* の値が空白の場合、EUROCURR は、指定されたヨーロッパの国の通貨からユーロに値を変換します。

詳細

次の表に、ヨーロッパ通貨とそれぞれに関連付けられた通貨コードを示します。通貨コードを使用して、変換元と変換先の通貨のタイプを識別します。自国の通貨として、次の表に示す通貨のかわりにユーロを使用する国もあります。この情報は、ユーザーの `historical data.buildnls.OTKEY` の必要を満たすために提供されています。

表 A1.1 ヨーロッパ通貨と通貨コード

通貨	通貨コード
オーストリアシリング	ATS
ベルギーフラン	BEF
イギリスポンドスターリング	GBP
チェココルナ	CZK
デンマーククローネ	DKK
ドイツマルク	DEM
オランダギルダー	NLG
ユーロ	EUR
フィンランドマルッカ	FIM
フランスフラン	FRF
ギリシャドラクマ	GRD
ハンガリーフォリント	HUF
アイルランドポンド	IEP
イタリアリラ	ITL
ルクセンブルクフラン	LUF
ノルウェークローネ	NOK
ポーランドズウォティ	PLZ

通貨	通貨コード
ポルトガルエスクード	PTE
ルーマニアレイ	ROL
ロシアルーブル	RUR
スロヴェニアトラー	SIT
スペインペセタ	ESP
スウェーデンクローナ	SEK
スイスフラン	CHF
トルコリラ	TRL
ユーゴスラビア ディーナール	YUD

EUROCURR 関数は、特定の国の通貨を別の国の通貨での同等額に変換します。また、特定の国の通貨をユーロに変換することもできます。EUROCURR では、固定通貨換算レートテーブルまたは変更可能な通貨換算レートテーブルのいずれかの値を使用して通貨を変換します。

1 つの国の通貨をユーロに変換する場合は、*from-currency-amount* の値を、一方の変換レートテーブルにあるその国のレートで除算します。詳細については、“[例 1: ドイツマルクからユーロへの変換](#)” (667 ページ)を参照してください。ユーロから 1 つの国の通貨に変換する場合は、*from-currency-amount* の値に、一方の変換レートテーブルにあるその国のレートを乗算します。詳細については、“[例 2: ユーロからドイツマルクへの変換](#)” (667 ページ)を参照してください。1 つの国の通貨を別の国の通貨に変換する場合は、まず *from-currency-amount* の値をユーロに変換します。SAS では、動作環境において可能な限り高い精度で、その中間値を丸めずに保存します。次に、ユーロの金額を変換先の通貨での金額に変換します。詳細については、“[例 3: フランスフランからドイツマルクへの変換](#)” (668 ページ)を参照してください。

例

例 1: ドイツマルクからユーロへの変換

次の例では、ドイツマルクをユーロでの同等額に変換します。

```
data _null_;
amount=eurocurr(50,'dem','eur');
put amount= ;
run;
```

SAS ログの値は、`amount=25.56459406` です。

例 2: ユーロからドイツマルクへの変換

次の例では、1 ユーロをドイツマルクでの同等額に変換します。

```
data _null_;
amount=eurocurr(25,'eur','dem');
put amount= ;
run;
```

SAS ログの値は、`amount=48.89575` です。

例 3: フランスフランからドイツマルクへの変換

次の例では、50 フランスフランをドイツマルクでの同等額に変換します。

```
data _null_;  
x=50;  
amount=eurocurr(x,'frf','dem');  
put amount=;  
run;
```

SAS ログの値は、`amount=14.908218069` です。

例 4: 1 つの変数が空白の場合の通貨の変換

次の例では、50 ユーロをドイツマルクに変換します。

```
data _null_;  
x=50;  
amount=eurocurr(x,' ','dem');  
put amount=;  
run;
```

SAS ログの値は、`amount=97.7915` です。

用語集

1 バイト文字セット

1 バイトのコンピュータメモリを使用して 1 文字を表現するエンコーディング。1 バイト文字セットの例として、Latin 1 があります。略称: SBCS

2 バイト文字セット

文字の表現に 1 バイトまたは 2 バイトのメモリが必要なエンコーディング。2 バイト文字セットの例として、日本語、韓国語、中国語があります。略称: DBCS

ANSI

米国の製品やサービス、プロセス、システム、および人材の自発的な合意基準の策定の調整を計る組織。ANSI は国際規格を作成するために、ISO (国際標準化機構) と協力して活動しています。略称: ANSI

ASCII

基本的な 128 文字のセットを提供し、多様なコンピュータシステムを支援する 7 ビットエンコーディング規格。ASCII では、英文字の大文字と小文字、発音記号、数字(0-9)、コントロール文字をエンコードします。この 128 文字のセットは、他のほとんどのエンコーディングにも含まれています。略称: ASCII

BIDI

アラビア語やヘブライ語などの記述体系にみられる文字の向きで、通常、数字や埋め込みテキストを除き、右から左の向きになります。他の言語では、文字の向きは左から右になります。略称: BIDI

bidirectional

BIDI を参照してください。

CEDA

任意のディレクトリベース動作環境(たとえば、Solaris、Windows、HP-UX、OpenVMS および z/OS)で作成された SAS データファイルを、別のディレクトリベースの環境で実行された SAS セッションで読み込めるようにする SAS の機能。中間の変換ステップを使用せずに、SAS データファイルにアクセスできます。略称: CEDA

DBCS

2 バイト文字セットを参照してください。

EBCDIC

IBM メインフレームやミッドレンジコンピュータのデータを表現する 1 バイトやマルチバイトのエンコーディングファミリー。EBCDIC では、英文字の大文字と小文字、発

音記号、数字(0-9)、コントロール文字の拡張セットをエンコードします。略称: EBCDIC

I18N

国際化を参照してください。

ISO

規格の策定を推進し、国家間での製品およびサービスの宣伝を促進するための関連活動を後援する組織。知的、科学的、技術的情報の交換の支援も行います。略称: ISO

ISO 646 ファミリ

ISO 646 規格に定義され、最大で 128 文字を含む 7 ビットエンコーディングのグループ名。ISO 646 エンコーディングは、ISO 646 が各国バリエーションで使用される 12 のコードポイントを持つという点を除いて、ASCII に似ています。各国バリエーションは、特定の言語で必要とされる特殊文字です。

ISO 8859 ファミリ

ISO 8859 規格に定義された、16 部からなる 8 ビットエンコーディングのセット。それぞれのエンコーディングに、そのサポート言語で使用される 128 個の ASCII 文字と 128 個の拡張文字の両方が含まれます。たとえば、Latin-1 と呼ばれる ISO 8859-1 は、ISO 8859 ファミリ内でよく使用されるエンコーディングであり、ASCII 文字と西ヨーロッパ言語で使用される文字を含みます。

MBCS

マルチバイト文字セットを参照してください。

NLS

ソフトウェアを各国の市場向けに正しく動作させるために必要な機能の集合。略称: NLS

SBCS

1 バイト文字セットを参照してください。

Unicode

世界の大部分の記述体系で表現されるテキストを矛盾なくエンコーディング、表現、処理するためのコンピュータ業界の規格。Unicode は、109,000 を超える文字を含み、数十のスク립ト体、大文字小文字などの文字のプロパティの規格、双方向文字のスク립ト体の表示規格、多数の関連項目にも対応しています。

アクセント記号付き文字

発音を変えるアクセント記号を付けられて修飾された文字。たとえば、"ñ" はチルダ(~)と文字"n"の組み合わせです。

エンコーディング

コード化された文字セットとコード値のマッピング。

エンコーディング方法

数値表現を文字セットの文字に割り当てるときに使用するルールセット。たとえば、これらのルールにより、文字の数値表現を保存するときに使用するビット数とともに、文字が表示されるコードページの範囲を指定します。エンコーディング方法は、コンピュータ業界で開発されてきた規格です。多くの場合、エンコーディング方法は、コンピュータのハードウェアベンダに固有です。よく使用されるエンコーディング方法として、ASCII、EBCDIC、ISO 646 ファミリ、ISO 8859 ファミリ、Unicode があります。

エンコード

特定の文字エンコーディング体系におけるデータの表記。たとえば、ASCII では、文字"A"は 16 進表現の 41 として表記されます。

表語文字

発音ではなく、語や形態素を表す視覚的な記号。中国の表語文字の例として、山という語を表す山があります。

拡張 2 進化 10 進コード

EBCDIC を参照してください。

各国語サポート

NLS を参照してください。

各国語文字(ナショナルキャラクタ)

ラテン文字(A-Z と a-z)でない、ある記述体系に属する文字(文字、表意文字または象形文字)。

グラフィック文字

書き出し、印刷、表示が可能な文字。

クロス環境データアクセス

CEDA を参照してください。

言語

ロケールの 1 つの要素。任意の 1 つの国または地域に固有とは限りません。たとえば、ポルトガル語は、ポルトガルだけでなくブラジルでも話されていますが、Portuguese_Portugal と Portuguese_Brazil は別々のロケールです。

国際化

1 つの言語またはロケールに基づくという前提なしで行われる、ソフトウェア製品の設計処理。国際化により、文字列の並べ替えと日付、時間、数、通貨の出力形式のルールを含む、国際的な規則がサポートされます。また、一貫したユーザーエクスペリエンスを製品のどの言語版でも可能にします。(略称: I18N)

国際標準化機構

ISO を参照してください。

コントロール文字

情報交換ではなくコントロールのために使用される文字。通常、コントロール文字の出力はできません。

照合順序

テキストデータの並べ替えや比較を決定するルールのセット。

情報交換用米国標準コード(American Standard Code for Information Interchange)

ASCII を参照してください。

双方向

BIDI を参照してください。

地域化

ターゲットの環境や市場の言語、文化、その他の要件を満たすように製品を適合させることで、顧客が製品を使用するときに自分の言語と規則を使用できるようにす

る処理。ユーザーインターフェイス、システムメッセージ、ドキュメントの翻訳も地域化に含まれます。

データ表記

特定の動作環境でデータを保存するときの形式。動作環境が異なると、使用される規格や規則も異なります。たとえば、浮動小数点数を保存する場合は IEEE や IBM 390、文字エンコーディングの場合は ASCII または EBCDIC、メモリ内のバイトオーダーではビッグエンディアンまたはリトルエンディアン、ワード配置では 4 バイト境界または 8 バイト境界、データ型の長さでは 16 ビット、32 ビット、64 ビットが使用されます。

特殊文字

英数字、アンダースコア(_)、空白以外の文字。たとえば、アスタリスク(*)などがあります。

トランスコーディング

1 つのエンコーディングから別のエンコーディングに SAS ファイルの内容を変換する処理。トランスコーディングが必要になるのは、UNIX の Latin 1 エンコーディングから IBM メインフレームのドイツ語 EBCDIC エンコーディングにデータを転送する場合のように、セッションエンコーディングとファイルエンコーディングが異なる場合です。

米国規格協会(American National Standards Institute)

ANSI を参照してください。

変換テーブル

1 つの文字の値を別の文字の値に変換するときに使用する、動作環境に固有の SAS カタログエントリ。変換テーブルは、多くの場合、アプリケーションにおける複数の各国語の使用をサポートするために必要になります。変換テーブルの例として、文字を EBCDIC から ASCII-ISO に変換するものがあります。

マルチバイト文字セット

文字の表現に 1 バイト以上のコンピュータメモリが必要なエンコーディング。マルチバイト文字セットの例として、日本語、韓国語、中国語があります。略称: MBCS

文字

意味を持つ、記述体系の最小構成要素(英文字、数字、表意文字など)。文字は、特定の具象を示すのではなく、抽象的な意味を表します。

文字セット

言語または言語グループで使用される文字の集まり。文字セットには、各国語文字、特殊文字、0 から 9 の数字、コントロール文字などがあります。

ロケール

地域の言語、地域の規則および文化を反映する設定。地域規則には、国または地域の用紙サイズ、日付、時刻、数字の出力形式、通貨記号が含まれます。ロケールの値の例として、French_Canada、Portuguese_Brazil、Chinese_Singapore などがあります。

キーワード

\$

\$BIDIw. 出力形式 83
 \$CPTDWw. 出力形式 85
 \$CPTWDw. 出力形式 85
 \$KANJIw. 出力形式 93
 \$KANJIw. 入力形式 334
 \$KANJIXw. 出力形式 94
 \$KANJIXw. 入力形式 334
 \$LOGVSRw. 出力形式 96
 \$LOGVSw. 出力形式 95
 \$REVERJw. 入力形式 417
 \$REVERSw. 入力形式との比較 418
 \$REVERSw. 入力形式 418
 REVERJw. 入力形式との比較 418
 \$UCS2BEw. 出力形式 207
 \$UCS2BEw. 入力形式 420
 \$UCS2Bw. 出力形式 206
 \$UCS2Bw. 入力形式 419
 \$UCS2LEw. 出力形式 209
 \$UCS2LEw. 入力形式 422
 \$UCS2Lw. 出力形式 208
 \$UCS2Lw. 入力形式 421
 \$UCS2Xew. 出力形式 212
 \$UCS2Xew. 入力形式 424
 \$UCS2Xw. 出力形式 211
 \$UCS2Xw. 入力形式 423
 \$UCS4BEw. 出力形式 214
 \$UCS4Bw. 出力形式 213
 \$UCS4Bw. 入力形式 425
 \$UCS4LEw. 出力形式 217
 \$UCS4Lw. 出力形式 215
 \$UCS4Lw. 入力形式 426
 \$UCS4Xew. 出力形式 219
 \$UCS4Xew. 入力形式 428
 \$UCS4Xw. 出力形式 218
 \$UCS4Xw. 入力形式 427
 \$UESCEw. 出力形式 221
 \$UESCEw. 入力形式 430
 \$UESCw. 出力形式 220
 \$UESCw. 入力形式 429
 \$UNCREw. 出力形式 223

\$UNCREw. 入力形式 432
 \$UNCRw. 出力形式 222
 \$UNCRw. 入力形式 431
 \$UPARENEw. 出力形式 225
 \$UPARENEw. 入力形式 434
 \$UPARENpw. 入力形式 435
 \$UPARENw. 出力形式 224
 \$UPARENw. 入力形式 433
 \$UTF8Xew. 出力形式 227
 \$UTF8Xw. 出力形式 226
 \$UTF8Xw. 入力形式 436
 \$VSLOGRw. 出力形式 229
 \$VSLOGRw. 入力形式 438
 \$VSLOGw. 出力形式 228
 \$VSLOGw. 入力形式 437

%

%KINDEX マクロ関数 444
 %KLEFT マクロ関数 445
 %KLENGTH マクロ関数 445
 %KLOWCASE 自動呼び出しマクロ 43
 %KSCAN マクロ関数 446
 %KSUBSTR マクロ関数 450
 %KTRIM 自動呼び出しマクロ 44
 %KVERIFY 自動呼び出しマクロ 44
 %QKTRIM 自動呼び出しマクロ 44

1

16 進表現
 変換テーブル 528
 1 バイト文字セット(SBCS) 12
 データのトランスコーディングで使用されるエンコーディング値 563

2

2 バイト文字
 文字式の数 266
 2 バイト文字セット(DBCS) 12, 460

関連項目: DBCS エンコーディング
 エンコーディング方法 462
 言語 461
 認識 460
 変換テーブル 515

8

8859 ISO ファミリ 13

A

ANSI (American National Standards
 Institute、米国規格協会) 15
 ASCII 12
 EBCDIC との間のデータ転送 32
 ATTRIB ステートメント 505
 TRANSCODE=オプション 506

B

BASETYPE=オプション
 PROC DBCSTAB ステートメント 516
 BOMFILE システムオプション 459
 BOTH オプション
 CLEAR ステートメント(TRANTAB)
 527
 LIST ステートメント(TRANTAB) 528
 SAVE ステートメント(TRANTAB) 530

C

CATALOG=オプション
 PROC DBCSTAB ステートメント 516
 CEDA 31
 CHARSET=オプション 482
 CHARSET オプション 500
 CIMPORT プロシジャ
 動作環境間のトランスコーディング 30
 変換テーブル 524
 CLEAR ステートメント
 TRANTAB プロシジャ 527
 CORRECTENCODING オプション 489
 CPORT プロシジャ
 動作環境間のトランスコーディング 30
 変換テーブル 524
 CPTDWw.入力形式 328
 CPTWDw.入力形式 329
 CVPBYTES オプション 490

D

DATA=オプション
 PROC DBCSTAB ステートメント 516
 DATESTYLE=システムオプション
 デフォルト値 555

DATESTYLE システムオプション 460
 DBCS

参照項目: 2 バイト文字セット(DBCS)
 DBCSLANG=オプション
 PROC DBCSTAB ステートメント 516
 DBCSLANG システムオプション 461
 DBCSTAB プロシジャ 515
 PROC DBCSTAB ステートメント 516
 構文 515
 日本語変換テーブル 518
 変換テーブルの作成 517
 例 517

DBCSTYPE システムオプション 462
 DBCS エンコーディング 12, 37
関連項目: 2 バイト文字セット(DBCS)
 DBCS でサポートされる言語 33
 SAS セッションのシステムオプション
 561
 エンコーディング間のデータ変換 40
 システムオプション値 561
 指定 38
 使用可能な場合 39
 先頭の空白の削除 269
 データのトランスコーディングで使用さ
 れるエンコーディング値 563
 半角/全角の空白の削除 281
 汎用機版 SAS 39
 フルスクリーン 467
 フルスクリーン入力方法モジュール
 (IMM) 468, 469
 末尾の空白の削除 276
 文字セットの表示要件 38
 文字データ切り捨て 34
 文字列の分割 40
 DBCS システムオプション 460
 DBCS データ
 シフトコードデータの削除 94, 334
 シフトコードデータの追加 93, 334
 DBCS でサポートされる言語 38
 DBCS エンコーディング 33, 37
 エンコーディング 13
 DESC=オプション
 PROC DBCSTAB ステートメント 516
 DFLANG=システムオプション 464
 デフォルト値 555

E

EBCDIC 13
 ASCII との間のデータ転送 32
 OpenEdition エンコーディング 33
 エンコーディングの制約 15
 ドイツ語コードページ 11
 ENCODCOMPAT 関数 258
 ENCODING=システムオプション
 Posix 値 555

- デフォルト設定 555
 - ENCODING=データセットオプション 49
 - ENCODING オプション 495
 - ENCODING システムオプション 465
 - ENCODISVALID 関数 260
 - EURDFDDw.出力形式 582
 - EURDFDEw.出力形式 584
 - EURDFDEw.入力形式 661
 - EURDFDNw.出力形式 585
 - EURDFDTw.出力形式 586
 - EURDFDTw.入力形式 662
 - EURDFDWNw.出力形式 588
 - EURDFMNw.出力形式 590
 - EURDFMYw.出力形式 592
 - EURDFMYw.入力形式 664
 - EURDFWDXw.出力形式 593
 - EURDFWKXw.出力形式 596
 - EURFRATSw.d 出力形式 598
 - EURFRBEFw.d 出力形式 600
 - EURFRCHFw.d 出力形式 601
 - EURFRDEMw.d 出力形式 602
 - EURFRDKKw.d 出力形式 603
 - EURFRESPw.d 出力形式 605
 - EURFRFIMw.d 出力形式 606
 - EURFRFRFw.d 出力形式 607
 - EURFRGBPw.d 出力形式 609
 - EURFRGRDw.d 出力形式 610
 - EURFRHUFw.d 出力形式 611
 - EURFRIEPw.d 出力形式 612
 - EURFRITLw.d 出力形式 614
 - EURFRLUFw.d 出力形式 615
 - EURFRNLGw.d 出力形式 616
 - EURFRNOKw.d 出力形式 618
 - EURFRPLZw.d 出力形式 619
 - EURFRPTEw.d 出力形式 620
 - EURFRROLw.d 出力形式 621
 - EURFRRURw.d 出力形式 623
 - EURFRSEKw.d 出力形式 624
 - EURFRSITw.d 出力形式 625
 - EURFRTRLw.d 出力形式 626
 - EURFRYUDw.d 出力形式 628
 - EUROCURR 関数 665
 - EUROw.d 出力形式 86
 - EUROw.d 入力形式 330
 - EUROXw.d 出力形式 89
 - EUROXw.d 入力形式 332
 - EURTOATSw.d 出力形式 629
 - EURTOBEFw.d 出力形式 630
 - EURTOCHFw.d 出力形式 631
 - EURTOCZKw.d 出力形式 633
 - EURTODEMw.d 出力形式 634
 - EURTODKKw.d 出力形式 635
 - EURTOESPw.d 出力形式 637
 - EURTOFIMw.d 出力形式 638
 - EURTOFRFw.d 出力形式 639
 - EURTOGBPw.d 出力形式 640
 - EURTOGRDw.d 出力形式 642
 - EURTOHUFw.d 出力形式 643
 - EURTOIEPw.d 出力形式 644
 - EURTOITLw.d 出力形式 645
 - EURTOLUFw.d 出力形式 647
 - EURTONLGw.d 出力形式 648
 - EURTONOKw.d 出力形式 649
 - EURTOPLZw.d 出力形式 651
 - EURTOPTew.d 出力形式 652
 - EURTOROLw.d 出力形式 653
 - EURTORURw.d 出力形式 654
 - EURTOSEKw.d 出力形式 656
 - EURTOSITw.d 出力形式 657
 - EURTOTRLw.d 出力形式 658
 - EURTOYUDw.d 出力形式 660
 - EUR 言語要素 582
- F**
- FORCE オプション
 - PROC DBCSTAB ステートメント 516
 - FSDBTYPE システムオプション 467
 - FSIMMOPT システムオプション 469
 - FSIMM システムオプション 468
- G**
- GETLOCENV 関数 260
 - GETPXLANGUAGE 関数 261
 - GETPXLOCALE 関数 262
 - GETPXREGION 関数 263
- H**
- HDATEw.出力形式 91
 - HEBDATEw. 出力形式 92
- I**
- IBw.d 入力形式 408, 410, 411, 412
 - ISO (国際標準化機構) 15
 - ISO エンコーディング 13
 - 8859 ファミリ 13
 - Windows ファミリ 13
- K**
- KCOMPARE 関数 264
 - KCOMPRESS 関数 265
 - KCOUNT 関数 266
 - KCVT 関数 266
 - KINDEXC 関数 268
 - KINDEX 関数 268
 - KLEFT 関数 269
 - KLENGTH 関数 270
 - KLOWCASE 関数 270

- KPROPCASE 関数 271
 KPROPCHAR 関数 273
 KPROPDATA 関数 274
 KREVERSE 関数 276
 KRIGHT 関数 276
 KSCAN 関数 277
 KSTRCAT 関数 278
 KSUBSTRB 関数 279
 KSUBSTR 関数 278
 KTRANSLATE 関数 280
 KTRIM 関数 281
 KTRUNCATE 関数 281
 KUPCASE 関数 282
 KUPCASE マクロ関数 452
 KUPDATEB 関数 284
 KUPDATE 関数 282
 KVERIFY 関数 285
 K 関数 243
- L**
- Latin1 コードページ 10
 LIST ステートメント
 TRANTAB プロシジャ 528
 LOAD ステートメント
 TRANTAB プロシジャ 528
 LOCALE=システムオプション
 値 549
 デフォルトのエンコーディング値 23
 LOCALELANGCHT システムオプション
 471
 LOCALE システムオプション 470
 LOGVSRw.入力形式 336
 LOGVSw.入力形式 335
- M**
- MBCS エンコーディング 12
 META 宣言 482
 MINGUOw.出力形式 97
 MINGUOw.入力形式 337
- N**
- NENGOw.出力形式 99
 NENGOw.入力形式 339
 NLBESTw.出力形式 100
 NLDATEMDw.出力形式 103
 NLDATEMNw.出力形式 104
 NLDATEw.出力形式 102
 NLDATEw.入力形式 340
 NLDATEWNw.出力形式 106
 NLDATEWw.出力形式 105
 NLDATEYMW.出力形式 107
 NLDATEYQw.出力形式 108
 NLDATEYRw.出力形式 109
 NLDATEYWw.出力形式 110
 NLDATE 関数 285
 NLDATMAPw.出力形式 112
 NLDATMDTw.出力形式 113
 NLDATMMDw.出力形式 114
 NLDATMMNw.出力形式 114
 NLDATMTMw.出力形式 115
 NLDATMTZw.出力形式
 日時値 116
 NLDATMw.出力形式 111
 NLDATMw.入力形式 341
 NLDATMWNw.出力形式 118
 NLDATMWw.出力形式 117
 NLDATMWZw.出力形式
 日時値 118
 NLDATMYMW.出力形式 119
 NLDATMYQw.出力形式 120
 NLDATMYRw.出力形式 121
 NLDATMYWw.出力形式 122
 NLDATMZw.出力形式
 日時値 123
 NLMNIAEDw.d 出力形式 124
 NLMNIAEDw.d 入力形式 342
 NLMNIAUDw.d 出力形式 124
 NLMNIAUDw.d 入力形式 343
 NLMNIBGNw.d 出力形式 125
 NLMNIBGNw.d 入力形式 344
 NLMNIBRLw.d 出力形式 126
 NLMNIBRLw.d 入力形式 345
 NLMNICADw.d 出力形式 127
 NLMNICADw.d 入力形式 345
 NLMNICHFw.d 出力形式 128
 NLMNICHFw.d 入力形式 346
 NLMNICNYw.d 出力形式 129
 NLMNICNYw.d 入力形式 347
 NLMNICZKw.d 出力形式 130
 NLMNICZKw.d 入力形式 348
 NLMNIDKKw.d 出力形式 131
 NLMNIDKKw.d 入力形式 349
 NLMNIEEKw.d 出力形式 132
 NLMNIEEKw.d 入力形式 350
 NLMNIEGPw.d 出力形式 133
 NLMNIEGPw.d 入力形式 351
 NLMNIEURw.d 出力形式 133
 NLMNIEURw.d 入力形式 352
 NLMNIGBPw.d 出力形式 134
 NLMNIGBPw.d 入力形式 353
 NLMNIHKDw.d 出力形式 135
 NLMNIHKDw.d 入力形式 354
 NLMNIHRKw.d 出力形式 136
 NLMNIHRKw.d 入力形式 354
 NLMNIHUFw.d 出力形式 137
 NLMNIHUFw.d 入力形式 355
 NLMNIIDRw.d 出力形式 138
 NLMNIIDRw.d 入力形式 356
 NLMNIILSw.d 出力形式 139

- NLMNIILSw.d 入力形式 357
 NLMNIINRw.d 出力形式 140
 NLMNIINRw.d 入力形式 358
 NLMNIJPYw.d 出力形式 141
 NLMNIJPYw.d 入力形式 359
 NLMNIKRWw.d 出力形式 142
 NLMNIKRWw.d 入力形式 360
 NLMNILTLw.d 出力形式 142
 NLMNILTLw.d 入力形式 361
 NLMNILVLw.d 出力形式 143
 NLMNILVLw.d 入力形式 362
 NLMNIMOPw.d 出力形式 144
 NLMNIMOPw.d 入力形式 363
 NLMNIMXNw.d 出力形式 145
 NLMNIMXNw.d 入力形式 363
 NLMNIMYRw.d 出力形式 146
 NLMNIMYRw.d 入力形式 364
 NLMNINOKw.d 出力形式 147
 NLMNINOKw.d 入力形式 365
 NLMNINZDw.d 出力形式 148
 NLMNINZDw.d 入力形式 366
 NLMNIPLNw.d 出力形式 149
 NLMNIPLNw.d 入力形式 367
 NLMNIRUBw.d 出力形式 150
 NLMNIRUBw.d 入力形式 368
 NLMNISEKw.d 出力形式 151
 NLMNISEKw.d 入力形式 369
 NLMNISGDw.d 出力形式 151
 NLMNISGDw.d 入力形式 370
 NLMNITHBw.d 出力形式 152
 NLMNITHBw.d 入力形式 371
 NLMNITRYw.d 出力形式 153
 NLMNITRYw.d 入力形式 372
 NLMNITWDw.d 出力形式 154
 NLMNITWDw.d 入力形式 372
 NLMNIUSDw.d 出力形式 155
 NLMNIUSDw.d 入力形式 373
 NLMNIZARw.d 出力形式 156
 NLMNIZARw.d 入力形式 374
 NLMNLAEDw.d 出力形式 157
 NLMNLAEDw.d 入力形式 375
 NLMNLAUDw.d 出力形式 158
 NLMNLAUDw.d 入力形式 376
 NLMNLBGNw.d 出力形式 159
 NLMNLBGNw.d 入力形式 377
 NLMNLBRLw.d 出力形式 160
 NLMNLBRLw.d 入力形式 378
 NLMNLCADw.d 出力形式 160
 NLMNLCADw.d 入力形式 379
 NLMNLCHFw.d 出力形式 161
 NLMNLCHFw.d 入力形式 380
 NLMNLCNYw.d 出力形式 162
 NLMNLCNYw.d 入力形式 381
 NLMNLCZKw.d 出力形式 163
 NLMNLCZKw.d 入力形式 381
 NLMNLDDKw.d 出力形式 164
 NLMNLDDKw.d 入力形式 382
 NLMNLEEKw.d 出力形式 165
 NLMNLEEKw.d 入力形式 383
 NLMNLEGPw.d 出力形式 166
 NLMNLEGPw.d 入力形式 384
 NLMNLEURw.d 出力形式 167
 NLMNLEURw.d 入力形式 385
 NLMNLGBPw.d 出力形式 168
 NLMNLGBPw.d 入力形式 386
 NLMNLHKDw.d 出力形式 169
 NLMNLHKDw.d 入力形式 387
 NLMNLHRKw.d 出力形式 169
 NLMNLHRKw.d 入力形式 388
 NLMNLHUFw.d 出力形式 170
 NLMNLHUFw.d 入力形式 389
 NLMNLIDRw.d 出力形式 171
 NLMNLIDRw.d 入力形式 390
 NLMNLILSw.d 出力形式 172
 NLMNLILSw.d 入力形式 390
 NLMNLINRw.d 出力形式 173
 NLMNLINRw.d 入力形式 391
 NLMNLJPYw.d 出力形式 174
 NLMNLJPYw.d 入力形式 392
 NLMNLKRWw.d 出力形式 175
 NLMNLKRWw.d 入力形式 393
 NLMNLLTLw.d 出力形式 176
 NLMNLLTLw.d 入力形式 394
 NLMNLLVLw.d 出力形式 177
 NLMNLLVLw.d 入力形式 395
 NLMNLMOPw.d 出力形式 178
 NLMNLMOPw.d 入力形式 396
 NLMNLMXNw.d 出力形式 178
 NLMNLMXNw.d 入力形式 397
 NLMNLMYRw.d 出力形式 179
 NLMNLMYRw.d 入力形式 398
 NLMNLNOKw.d 出力形式 180
 NLMNLNOKw.d 入力形式 399
 NLMNLNZDw.d 出力形式 181
 NLMNLNZDw.d 入力形式 399
 NLMNLPLNw.d 出力形式 182
 NLMNLPLNw.d 入力形式 400
 NLMNLRUBw.d 出力形式 183
 NLMNLRUBw.d 入力形式 401
 NLMNLSEKw.d 出力形式 184
 NLMNLSEKw.d 入力形式 402
 NLMNLSGDw.d 出力形式 185
 NLMNLSGDw.d 入力形式 403
 NLMNLTHBw.d 出力形式 186
 NLMNLTHBw.d 入力形式 404
 NLMNLTRYw.d 出力形式 187
 NLMNLTRYw.d 入力形式 405
 NLMNLTWDw.d 出力形式 187
 NLMNLTWDw.d 入力形式 406
 NLMNLUSDw.d 出力形式 188
 NLMNLUSDw.d 入力形式 407
 NLMNLZARw.d 出力形式 189

- NLMNLZARw.d 入力形式 408
 NLMNYIw.d 出力形式 191
 NLMNYw.d 出力形式 190
 NLNUMIw.d 出力形式 194
 NLNUMw.d 出力形式 193
 NLPCTIw.d 出力形式 197
 NLPCTIw.d 入力形式 414
 NLPCTNw.d 出力形式 198
 NLPCTPw.d 出力形式 199
 NLPCTw.d 出力形式 195
 NLPCTw.d 入力形式 413
 NLPVALUE.w.d 出力形式 200
 NLS(各国語サポート) 3
 DBCS 37
 DBCSTAB プロシジャ 515
 TRANTAB プロシジャ 521
 エンコーディング 9
 互換性 473
 トランスコーディング 27
 ロケール 5
 NLSCOMPATMODE システムオプション 473
 NLSTRMONw.d 出力形式 201
 NLSTRQTRw.d 出力形式 202
 NLSTRWKw.d 出力形式 203
 NLS オプション
 LOAD ステートメント(TRANTAB) 528
 PROC TRANTAB ステートメント 526
 NLTIMAPw.出力形式 204
 NLTIMAPw.入力形式 415
 NLTIMEw.出力形式 205
 NLTIMEw.入力形式 416
 NLTIME 関数 290
 NLTIME ディスクリプタ 290
 NODATM 関数 288
- O**
 ODS 出力
 テキストの言語の変更 471
 ONE オプション
 CLEAR ステートメント(TRANTAB) 527
 LIST ステートメント(TRANTAB) 528
 SAVE ステートメント(TRANTAB) 530
 OpenEdition エンコーディング 33
 OPT=オプション, TRANTAB ステートメント 509
- P**
 PAPERSIZE=システムオプション
 デフォルト値 555
 PAPERSIZE システムオプション 474
 PROC DBCSTAB ステートメント 516
 PROC TRANTAB ステートメント 526
- R**
 RENCODING オプション 503
 REPLACE ステートメント
 TRANTAB プロシジャ 529
 RLS(リモートライブラリサービス)
 変換テーブル 524
 RSASIOTRANSERROR システムオプション 474
- S**
 SAS/CONNECT
 計算サービス 31
 データ転送サービス 31
 リモートライブラリサービス 31
 SAS/GRAPH
 変換テーブル 524
 SAS/SHARE
 リモートライブラリサービス 31
 SASMSG 関数 294
 SASMSG 関数 292
 SAS 言語要素
 エンコーディング値の使用 563
 SAS セッション
 DBCS のシステムオプション 561
 エンコーディング 20
 エンコーディングの設定 22
 エンコーディングの動作 24
 デフォルトのエンコーディング 21
 デフォルトの文字セットエンコーディング 465
 ロケール 470
 ロケールの指定 7
 SAS 文字列関数
 国際化の互換性 243
 SAVE ステートメント
 TRANTAB プロシジャ 530
 SBCS エンコーディング 12
 データのトランスコーディングで使われるエンコーディング値 563
 SETLOCALE 関数 299
 SO/SI(シフトアウト/シフトイン) 39
 文字式からの削除 281
 SORTKEY 関数 297
 SORTSEQ=システムオプション 474
 SORT オプション 483
 SORT プロシジャ
 言語固有の照合順序 474
 変換テーブル 523
 SQL プロシジャ
 言語固有の照合順序 474
 SWAP ステートメント
 TRANTAB プロシジャ 525

T

TABLE=オプション
 SAVE ステートメント(TRANTAB) 530
 TRANSCODE=オプション
 ATTRIB ステートメント 506
 TRANTAB-ODS オプション 501, 507
 TRANTAB=システムオプション 476
 TRANTAB 関数 306
 TRANTAB ステートメント
 UPLOAD プロシジャ 509
 TRANTAB プロシジャ 521
 CLEAR ステートメント 527
 LIST ステートメント 528
 LOAD ステートメント 528
 PROC TRANTAB ステートメント 526
 REPLACE ステートメント 529
 SAVE ステートメント 530
 SWAP ステートメント 525
 概念 522
 逆変換テーブル 539
 並べ替える変換テーブル 540
 変換テーブルと文字セット 522
 変換テーブルの作成 532
 変換テーブルの変更 523, 535, 537,
 542
 変換テーブルの保存 523
 TWO オプション
 CLEAR ステートメント(TRANTAB)
 527
 LIST ステートメント(TRANTAB) 528
 SAVE ステートメント(TRANTAB) 530
 TYPE=オプション, TRANTAB ステートメ
 ント 509

U

Unicode 13
 外部ファイルの BOM 接頭辞 459
 データのトランスコーディングで使用さ
 れるエンコーディング値 563
 表示単位の長さ 311
 文字単位の長さ 310
 UNICODEC 関数 308
 UNICODLEN 関数 310
 UNICODWIDTH 関数 311
 UNICODE 関数 307
 Unicode コンソーシアム 15
 UNIX
 エンコーディング値 573
 UPLOAD プロシジャ
 TRANTAB ステートメント 509
 URLENCODING=システムオプション
 478
 UTF-16 0
 UTF-32 0
 UTF-8 0

V

VALIDMEMNAME システムオプション
 478
 VALIDVARNAME=システムオプション
 478
 VARTRANSCODE 関数 311
 VERBOSE オプション
 PROC DBCSTAB ステートメント 516
 VERIFY オプション
 PROC DBCSTAB ステートメント 516
 VTRANSCODEX 関数 314
 VTRANSCODE 関数 313

W

WEEKUw.出力形式 231
 WEEKVw.出力形式 232
 WEEKWw.出力形式 234
 Windows
 ISO エンコーディング 13
 Latin1 コードページ 10
 エンコーディング値 574

X

XMLENCODING オプション 508

Y

YENw.d 出力形式 235
 YENw.d 入力形式 439
 YYWEEKUw.出力形式 236
 YYWEEKVw.出力形式 237
 YYWEEKWw.出力形式 239

Z

z/OS
 エンコーディング値 575
 エンコーディングのサポート 24

あ

アイルランド
 通貨データ 133, 167
 アラビア語文字
 逆順 83
 視覚サーバーでの論理的順序のテキ
 ストの保存 95, 96
 アラブ首長国連邦
 通貨データ 124, 157
 イスラエル
 通貨データ 139, 172
 イタリア
 通貨データ 133, 167
 引数

- 大文字への変換 282
- 小文字への変換 270
- 指定した引数のトランスコーディング 314
- 長さ 270
- 部分文字列の抽出 278
- 部分文字列の抽出、バイト位置に基づく 279
- インド
 - 通貨データ 140, 173
- インドネシア
 - 通貨データ 138, 171
- 英国
 - 通貨データ 134, 168
- エジプト
 - 通貨データ 133, 166
- エスケープコード 39
- エストニア
 - 通貨データ 132, 165
- 円記号, 削除 439
- エンコーディング 9
 - 1つのデータタイプから別のデータタイプへの変換 266
 - DBCSでサポートされる言語 13
 - SASセッション 20
 - SASセッションでの動作 24
 - SASセッションの設定 22
 - z/OSサポート 24
 - 外部ファイルの読み込みと書き込み 25
 - 混在 51
 - 出力処理 25
 - 手動設定 49
 - 対トランスコーディング 11
 - データセット 20
 - 定義 10
 - デフォルトのSASセッションエンコーディング 21
 - トランスコーディングの互換性 32
 - 入力処理 25, 52
 - 標準組織 15
 - 文字セット 12
 - リリース別データセットサポート 24
- エンコーディング値 18, 563
 - DBCS 563
 - SBCS 563
 - Unicode 563
 - UNIX 573
 - Windows 574
 - z/OS 575
 - データのトランスコーディング 563
 - デフォルトのSASセッション値 21
 - デフォルト、LOCALE=システムオプションに基づく 23
- エンコーディングの手動設定 49
- エンコーディング方法 10, 12
 - DBCS 462
- オーストラリア
 - 通貨データ 124, 158
- オーストリア
 - 通貨データ 167
- オーストリアシリング
 - ユーロへの変換 598
- 大文字
 - 引数の変換 282
 - 小文字への変更 43
- オランダ
 - 通貨データ 133, 167
- か
- 改行文字 32
- 外部ファイル
 - UnicodeファイルのBOM接頭辞 459
 - エンコーディング 25
- 各国語サポート
 - 参照項目: NLS(各国語サポート)
- 各国語サポート(NLS)
 - 出力形式 55
- カナダ
 - 通貨データ 127, 160
- 韓国
 - 通貨データ 142, 175
- 関数
 - K関数 243
 - SAS文字列関数 243
 - カテゴリ別 256
- カンマ, 削除 439
- ギリシャ
 - 通貨データ 133, 167
- 空白
 - 先頭のDBCSの空白の削除 269
 - 末尾のDBCSの空白の削除 276
 - 末尾の削除 44
- グリーンランド
 - 通貨データ 131, 164
- クロアチア
 - 通貨データ 136, 169
- クロス環境データアクセス(CEDA) 31
- ケース
 - 大文字から小文字への変更 43
- 言語 5
 - 国際化対応の日付の入力形式と出力形式 464
- 言語コード
 - 現在の2文字のコード 261
- 言語照合 19
- 言語並べ替えキー 297
- 言語の切り替え 7
 - ODS出力のテキスト言語の変更 471
- 検索
 - 文字式の特定の文字 268

文字式の文字列 268
 文字列内の単語の位置指定 446
 コードページ 9
 互換性 473
 SAS 文字列関数 243
 互換性のあるエンコーディング 32
 互換性のないエンコーディング 32
 国際化 4
 国際化対応の日時値, 書き込み
 ddmmyy:hh:mm:ss:ss 586
 国際化対応の日付/日時出力形式 55
 国際化対応の日付値, 書き込み
 dd.mm.yy 582
 ddmmyy 584
 mmyy 592
 月名 590, 593
 曜日と日付 596
 曜日名 588
 曜日番号 585
 国際化対応の日付の出力形式と入力形式
 言語の指定 464
 国際標準化機構(ISO) 15
 小文字
 引数の変換 270
 大文字からの変更 43

さ

視覚サーバー
 論理的順序のテキストの保存 95, 96
 時間 5
 時間値
 指定したロケールへの変換 290
 システムオプション
 DBCS 値 561
 DBCS の SAS セッション 561
 トランスコーディング 30
 シフトアウト/シフトイン(SO/SI) 39
 文字式からの削除 281
 シフトコードデータ
 DBCS データからの削除 94, 334
 DBCS データへの追加 93, 334
 出力位置
 文字式 269, 276
 出力形式
 NLS 55
 SO/SI システムでの DBCS のサポート 39
 国際化対応の日付/日時出力形式 55
 国際化対応の日付に使用する言語 464
 変数との関連付け 505
 出力処理 25
 照合順序 16
 エンコーディング値 18

概要 16
 言語固有 474
 言語照合 19
 異なる順序の結果 16
 代替順序 17
 バイナリ照合 16
 変換テーブル 18
 小数点, 削除 439
 シンガポール
 通貨データ 151, 185
 スイス
 通貨データ 128, 161
 スウェーデン
 通貨データ 151, 184
 数値 6
 切り捨て 281
 数値データ
 円 235
 台湾の日付値 97
 日本の日付 99
 数値の切り捨て 281
 数値表記
 最善、ロケールに基づく 100
 スペイン
 通貨データ 133, 167
 スロベニア
 通貨データ 133, 167

た

タイ
 通貨データ 152, 186
 台湾
 通貨データ 154, 187
 台湾の日付 97, 337
 単語
 文字列内の位置を指定して検索 446
 地域化 4
 地域コード
 現在の 2 文字のコード 263
 チェコ共和国
 通貨データ 130, 163
 中国
 通貨データ 129, 162
 通貨 6
 1 つのヨーロッパ通貨から別のヨーロッパ通貨への変換 665
 円 235
 通貨データ
 アイルランド 133, 167
 アラブ首長国連邦 124, 157
 イスラエル 139, 172
 イタリア 133, 167
 インド 140, 173
 インドネシア 138, 171
 英国 134, 168

- エジプト 133, 166
 - エストニア 132, 165
 - オーストラリア 124, 158
 - オーストリア 167
 - オランダ 133, 167
 - カナダ 127, 160
 - 韓国 142, 175
 - ギリシャ 133, 167
 - グリーンランド 131, 164
 - クロアチア 136, 169
 - シンガポール 151, 185
 - スイス 128, 161
 - スウェーデン 151, 184
 - スペイン 133, 167
 - スロベニア 133, 167
 - タイ 152, 186
 - 台湾 154, 187
 - チェコ共和国 130, 163
 - 中国 129, 162
 - デンマーク 131, 164
 - ドイツ 133, 167
 - トルコ 153, 187
 - 日本 141, 174
 - ニュージーランド 148, 181
 - ノルウェー 147, 180
 - ハンガリー 137, 170
 - フィンランド 133, 167
 - プエルトリコ 155, 188
 - フェロー諸島 131, 164
 - ブラジル 126, 160
 - フランス 133, 167
 - ブルガリア 125, 159
 - 米国 155, 188
 - ベルギー 133, 167
 - ポーランド 149, 182
 - ポルトガル 133, 167
 - 香港 135, 169
 - マカオ 144, 178
 - マルタ 133, 167
 - マレーシア 146, 179
 - 南アフリカ 156, 189
 - メキシコ 145, 178
 - ラトビア 143, 177
 - リトアニア 142, 176
 - リヒテンシュタイン 128, 161
 - ルクセンブルク 133, 167
 - ロシア 150, 183
 - データセット
 - エンコーディング 20
 - エンコーディングサポート、リリース別 24
 - 混在したエンコーディング 51
 - 特定のエンコーディングの使用 51
 - トランスコーディングの抑制 51
 - 変数のトランスコーディング属性 311
 - データセットオプション
 - トランスコーディング 30
 - データ変換
 - DBCS エンコーディング 40
 - デンマーク
 - 通貨データ 131, 164
 - ドイツ
 - 通貨データ 133, 167
 - ドイツ語 EBCDIC コードページ 11
 - 動作環境
 - トランスコーディング 30
 - トランスコーディング 12, 27
 - EBCDIC エンコーディングと OpenEdition エンコーディング 33
 - EBCDIC と ASCII 間のデータ転送 32
 - SAS オプション 30
 - エンコーディング値 563
 - 改行文字 32
 - 考慮事項 31
 - 互換性があるエンコーディングと互換性がないエンコーディング 32
 - 指定した引数に有効 314
 - 指定した変換テーブル 306
 - 指定した文字変数で有効 313
 - 対エンコーディング 11
 - 動作環境間 30
 - 変換テーブル 28
 - 防止 34
 - 抑制 51
 - 理由 28
 - トランスコーディングエラー 474
 - トランスコーディング属性
 - データセット変数 311
 - トランスポートファイル
 - 変換テーブルの適用 510
 - トルコ
 - 通貨データ 153, 187
- な**
- 長いマクロ変数
 - 値の分割保存 451
 - 長さ
 - 引数 270
 - 変数との関連付け 505
 - 並べ替え
 - 変換テーブル 540
 - 並べ替えキー
 - 言語 297
 - 日時値
 - 国際化対応 662
 - 午前または午後の使用 112
 - 時刻 115
 - 指定したロケールへの変換 288, 290
 - 月の名前 114
 - 月の名前と月の日 114
 - 月の名前、月の日、年 113

- 年 121
 - 年と四半期 120
 - 年と週の名前 122
 - 年と月の名前 119
 - 日時 111
 - 曜日 118
 - 曜日と日時 117
 - 日時形式ディスクリプタ 288
 - 日本
 - 通貨データ 141, 174
 - 日本語変換テーブル 518
 - 日本の日付 99, 339
 - ニュージーランド
 - 通貨データ 148, 181
 - 入力形式
 - SO/SI システムでの DBCS のサポート 39
 - 国際化対応の日付に使用する言語 464
 - 変数との関連付け 505
 - 入力処理 25
 - エンコーディングの手動設定 52
 - 入力方法モジュール(IMM) 468
 - オプション 469
 - ノルウェー
 - 通貨データ 147, 180
- は**
- バイトオーダーマーク(BOM)接頭辞
 - Unicode 外部ファイル 459
 - バイナリ照合 16
 - バイナリ整数値、読み込み 408, 410, 411, 412
 - バリエーション文字 16
 - ハンガリー
 - 通貨データ 137, 170
 - 汎用機
 - DBCS 39
 - 日付 5
 - 日付値 102
 - 日付値
 - 国際化対応 661
 - 国際化対応の月と年 664
 - 指定したロケールへの変換 285
 - 台湾語 97, 337
 - 月の名前 104
 - 月の名前と月の日 103
 - 年 109
 - 年と四半期 108
 - 年と週 110
 - 年と月の名前 107
 - 日本語 99, 339
 - 日付 102
 - 日付と曜日 105
 - ヘブライ語 91
 - ユダヤ暦 92
 - 曜日 106
 - 日付形式ディスクリプタ 285
 - 標準組織 15
 - ファイル参照名
 - 8 文字の制約 450
 - ファイルのアップロード
 - 変換テーブル 509
 - フィンランド
 - 通貨データ 133, 167
 - プエルトリコ
 - 通貨データ 155, 188
 - フェロー諸島
 - 通貨データ 131, 164
 - 不正なデータ 474
 - 部分文字列
 - 引数からの抽出 278
 - 引数からの抽出、バイト位置に基づく 279
 - 文字列 450
 - ブラジル
 - 通貨データ 126, 160
 - フランス
 - 通貨データ 133, 167
 - ブルガリア
 - 通貨データ 125, 159
 - フルスクリーン DBCS エンコーディング 467
 - 入力方法モジュール(IMM) 468, 469
 - 分割
 - 長いマクロ変数値の保存 451
 - 米国
 - 通貨データ 155, 188
 - ヘブライ語の日付値 91
 - ヘブライ語文字 85
 - 逆順 83
 - 視覚サーバーでの論理的順序のテキストの保存 95, 96
 - ベルギー
 - 通貨データ 133, 167
 - 変換テーブル 18, 521
 - 16 進表現 528
 - CIMPORT プロシジャ 524
 - CPORT プロシジャ 524
 - DBCS 515
 - SAS/GRAPH 524
 - SORT プロシジャ 523
 - TRANTAB プロシジャの外部 523
 - 位置 521
 - 位置のクリア 527
 - 逆変換テーブル 539
 - 交換 525
 - 作成 517, 532
 - 指定 476
 - 指定したテーブルを使用したトランスコーディング 306

- データの並べ替え 540
- 定義 521
- トランスコーディング 28
- トランスポートファイルへの適用 510
- 日本語 518
- 変更 523
- 編集 535, 537, 542
- 編集のためのメモリへのロード 528
- 保存 523, 530
- 文字セット 522
- 文字の置換 529
- リモートライブラリサービス(RLS) 524
- 変数
 - 指定した文字変数で有効なトランスコーディング 313
 - 出力形式の関連付け 505
 - トランスコーディング属性 311
 - 長さ、関連付け 505
 - 入力形式の関連付け 505
 - ラベル 505
- ポーランド
 - 通貨データ 149, 182
- ポルトガル
 - 通貨データ 133, 167
- 香港
 - 通貨データ 135, 169
- ま
- マカオ
 - 通貨データ 144, 178
- マクロ変数
 - 長い値の分割保存 451
- 末尾の空白
 - 削除 44
- 末尾の空白の削除 44
- マルタ
 - 通貨データ 133, 167
- マレーシア
 - 通貨データ 146, 179
- 右詰め
 - 文字式 276
- 南アフリカ
 - 通貨データ 156, 189
- メキシコ
 - 通貨データ 145, 178
- 文字
 - 検索 444
- 文字式
 - 2 バイト文字の数 266
 - 圧縮 265
 - 逆順 276
 - 検証 285
 - 更新 282
 - 更新、バイト単位に基づく 284
 - 最初の固有文字の位置 285
- 削除 281
- 指定した単語の選択 277
- 特定の文字の検索 268
- 特定の文字の置換 280
- 比較 264
- 左詰め 269
- 変換 280
- 末尾の半角/全角の空白の削除 281
- 右詰め 276
- 文字値の内容の削除 282
- 文字値の内容の削除、バイト単位に基づく 284
- 文字値の内容の挿入 282
- 文字値の内容の挿入、バイト単位に基づく 284
- 文字値の内容の置換 282
- 文字値の内容の置換、バイト単位に基づく 284
- 文字列の検索 268
- 連結 278
- 文字式の左詰め 269
- 文字式の変換 280
- 文字セット 12
 - DBCS の表示 38
 - 指定、出力の META 宣言 482
 - 定義 10
 - 変換テーブル 522
- 文字セットエンコーディング
 - SAS セッション 465
- 文字データ
 - 右から左への読み込み 417, 418
- 文字変数
 - 指定した変数で有効なトランスコーディング 313
- 文字列 450
 - 位置指定して単語を検索 446
 - 先頭文字の検索 444
 - 長さ 445
 - 長さを返す 445
 - 分割 40
 - 文字列の部分文字列 450
- 文字列関数
 - 国際化の互換性 243
- 文字列の分割 40
- や
- ユーロ
 - 出力形式 86, 89
- ユーロ変換
 - オーストリアシリングからユーロへ 598
- ユダヤ暦の日付 92
- ヨーロッパ通貨換算
 - 1 つの通貨から別の通貨への変換 665
- 用紙サイズと測定単位 6

ら

ラトビア

通貨データ 143, 177

ラベル, 変数との関連付け 505

リトアニア

通貨データ 142, 176

リヒテンシュタイン

通貨データ 128, 161

リモートアプリケーション

不正なデータ 474

リモートライブラリサービス(RLS)

変換テーブル 524

リリースの互換性 473

ルクセンブルク

通貨データ 133, 167

連結

文字式 278

ロケール 5

POSIX 値 262

SAS 起動時の指定 6

SAS セッション 470

SAS セッション中の指定 7

言語の切り替え 7

最適な数値表記 100

指定 6

指定したロケールへの時間値または日
時値の変換 290指定したロケールへの日時値の変換
288指定したロケールへの日付値の変換
285

ロシア

通貨データ 150, 183

論理的順序のテキスト

視覚サーバーでの保存 95, 96

