

## SAS® 認定スペシャリストのためのBase Programming for SAS®9 完全ガイド

\*イタリック体のフォントは、特に指定がない限り、テキストで変更されたものです。

### 内容の修正

ページ番号	項目	コメント
3	Chapter 1、手順	SAS®認定スペシャリスト完全ガイドの演習データがSAS On Demand for Academicsに追加されました。データにアクセスする手順については、 <a href="#">SAS OnDemand for Academicsで演習データを作成するための手順</a> を参照してください。
47	Chapter 4、ExcelワークシートをSASデータセットとして印字する	データファイルstock.xlsxが提供しているデータに含まれておらず、プログラムを実行していただけません。
103	Chapter 7、DATAステップの反復	「DATAステップの反復」のセクションが更新され、画像が鮮明になり、修正されました。このドキュメントの後半にある <a href="#">修正されたセクション: DATAステップの反復</a> を参照してください。
129	Chapter 9、日付定数	更新された構文: <i>' ddmmyy' d</i> または <i>' ddmmyyyy' d</i>
141-144	Chapter 9	本書で指定されているデータセットが正しくありません。Cert.TrialsではなくCert.Ctialsを参照する必要があります。
158	Chapter 10、例：1対1読み込みを使用したデータセットの組み合わせ	最後の段落への更新：  結果のデータセットWork.One2oneには、9オブザベーションが含まれています（最小のデータセットであるCert.Measureから読み込まれたオブザベーション数）。
172	Chapter 10、マッチするオブザベーションの選択	最後の段落への更新：  前の例では、Work.Mergedには12のオブザベーションが含まれていました。以下の出力では、11のオブザベーションのみがIF式の条件を満たしていることに注意してください。
237	Chapter 14、例：新しいName変数の作成	完全なDATAステップになるよう更新しました：  <pre>data work.newnames(drop=name; set cert.staff;   LastName=scan(name,1);   MiddleName=scan(name,3);   FirstName=scan(name,2); run;</pre>
240	Chapter 14、例：SUBSTR関数	更新された完全なコード：  <pre>data work.agencyemp(drop=middlename);   set cert.agencyemp;   MiddleInitial=substr(middlename,1,1); run; proc print data=work.agencyemp; run;</pre>
256	CEIL関数とFLOOR関数	箇条書きの導入文は次のようになります：  これらの関数を使用して、数値引数を最も近い整数に丸めます。
316	Chapter 17、シナリオ2	説明 - 更新された2番目の項目： <ul style="list-style-type: none"> <li>RestHR値が70より大きいオブザベーションを削除します。</li> </ul>
327	Appendix 1、Chapter 4、3	正解：b  説明を更新： IMPORTプロシジャの前にOPTIONSステートメントでOBS=オプションを使用して、SASが外部ファイルから読み込むオブザベーションの数を制限します。PROC PRINTステートメントでOBS=オプションを使用すると、ファイ

## SAS® 認定スペシャリストのためのBase Programming for SAS® 9 完全ガイド

\*イタリック体のフォントは、特に指定がない限り、テキストで変更されたものです。

ページ番号	項目	コメント
		ル全体がインポートされますが、印字は指定されたオブザベーションの数に制限されます。DELIMITER='.'を使用して区切り文字がピリオド(.)であることを示し、GETNAMES=YESを使用して、変数名を含む最初の行を読み込みます。
329	Appendix 1、Chapter 5、6	説明を更新：  DATAステートメントとデータセットwork.mysalesの間に=記号があります。等号は削除する必要があります。
334	Appendix 1、Chapter 10、6	説明を更新：  したがって、一致しないオブザベーションを含む解答 <b>b</b> と <b>c</b> は正しくありません。
343	Appendix 2、シナリオ2	PROC PRINTステートメントの#6コメントには、終了のスラッシュがありません。コードは次のようになります：  <pre>proc print data=work.stress1; /* #6 */</pre>
343	Appendix 2、シナリオ2	#3の更新されたコードの説明：  WHEREステートメントは、RestHRの値が70以下のオブザベーションのみを選択します。
360	インデックス	VALIDNARNAMEはVALIDVARNAMEである必要があります。

## フィードバックに基づく拡張

ページ番号	項目	コメント
112	Chapter 7のクイズ、1	質問の言い換え：  コンパイルフェーズで作成されないものは次のうちどれですか？
113	Chapter 7のクイズ、10	質問の言い換え：  どのプロシジャが変数の異なる値を生成し、データのクリーンアップするために有用ですか？
116	Chapter 8、データに前処理が必要かどうかの判定	次の文を追加：  BYグループ処理に使用するBY変数でデータが並べ替えられていることを確認します。
122	Chapter 8、クイズ、4	オプションb、c、およびdにPROC PRINTステップが追加されました。PROC PRINTステップがないと、データはHTML出力に表示されません。ただし、並べ替えられたデータセットはCertライブラリで表示できます。

## SAS OnDemand for Academicsで演習データを作成するための手順

これらの手順は、次の認定試験ガイドを対象としています：

- SAS 認定スペシャリストのためのBase Programming for SAS® 9 完全ガイド
  - SAS 認定プロフェッショナルのためのBase Programming for SAS® 9 完全ガイド 第3版
1. <https://support.sas.com/content/dam/SAS/support/en/books/data/base-guide-practice-data.zip>に移動し、ZIPファイルをコンピュータ上のダウンロードフォルダなどの場所にダウンロードします。
  2. ファイルを解凍し、解凍したファイルをコンピュータ上の場所に保存して、場所をもう一度見つけられるようにします。
  3. SAS OnDemand for Academicsにログオンし、SAS Studioを開きます。新しいフォルダを作成するためにサーバーファイルとフォルダペインをクリックします。

4. SAS Studioセッションで、**ファイル(ホーム)**を右クリックし、**[新規作成]>[フォルダ]**を選択します。
  - a. フォルダに**base-guide-practice-data**という名前を付けます。
  - b. **[保存]**をクリックします。
5. 新しい**base-guide-practice-data**フォルダを右クリックし、**[新規作成]>[フォルダ]**を選択します。
  - a. フォルダに**cert**という名前を付けます。
  - b. **[保存]**をクリックします。
6. 新しい**cert**フォルダを右クリックして、**[ファイルのアップロード...]**を選択します。
  - a. **[ファイルの選択]**をクリックします。
  - b. 解凍したコンテンツを保存したローカルマシンの**base-guide-practice-data/cert**フォルダに移動します。
  - c. すべてのファイルを選択し (Ctrl-A) 、**[開く]**をクリックします。すべてのファイルが**[ファイルのアップロード]**ウィンドウに表示されます。
  - d. **[アップロード]**をクリックします。
7. 新しい**base-guide-practice-data**フォルダを右クリックし、**[新規作成]>[フォルダ]**を選択します。
  - a. フォルダに**ehs**という名前を付けます。
  - b. **[保存]**をクリックします。
8. **ehs**フォルダを右クリックし、**[ファイルのアップロード...]**を選択します。
  - a. **[ファイルの選択]**をクリックします。
  - b. ローカルマシンの**base-guide-practice-data/ehs**フォルダに移動します。
  - c. すべてのファイルを選択し (Ctrl-A) 、**[開く]**をクリックします。すべてのファイルが**[ファイルのアップロード]**ウィンドウに表示されます。
  - d. **[アップロード]**をクリックします。
9. 新しい**base-guide-practice-data**フォルダを右クリックし、**[ファイルのアップロード...]**を選択します。
  - a. **[ファイルの選択]**をクリックします。
  - b. ローカルマシンの**base-guide-practice-data**フォルダに移動します。
  - c. **readme.txt**を選択します。
  - d. **[アップロード]**をクリックします。
10. SAS OnDemand for Academicsで、**cert**フォルダをクリックします。

## 修正されたセクション：DATAステップの反復

### DATA ステップの反復

DATA ステップがループのように機能し、ステートメントを繰り返し実行してデータ値を読み込み、オブザベーションを1つずつ作成していることがわかります。2回目の反復の開始時、**\_N\_**の値は2であり、**\_ERROR\_**はまだ0です。各ループ(または実行サイクル)は、反復と呼ばれます。

図 7.11 DATA ステップの反復

```

data work.update;
  set cert.invent;
  Total=instock+backord;
  SalePrice=(CostPerUnit*0.65)+CostPerUnit;
  format CostPerUnit SalePrice dollar6.2;
run;
    
```

SET ステートメントが2回実行されると、**CERT.INVENT**の2番目のレコードの値が入力テーブルからPDVに読み込まれます。**WORK.UPDATE**出力データセットは、DATA ステッププログラムの最初の反復で計算された新しいオブザベーションのみを保持します。PDVは、Item、IDnum、InStock、BackOrd、およびCostPerUnit 変

数の CERT.INVENT の 2 行目の値を保持していることに注意してください。Total と SalePrice の値は、欠損値に初期化されています。

図 7.12 プログラムデータベクトルと出力データセット

プログラムデータベクトル								
Item	IDnum	InStock	BackOrd	CostPerUnit	Total	SalePrice	N	ERROR
6 Glass Mugs	SB082	6	12	\$1.50	.	.	2	0

  

SASデータセットWork.Updateの出力								
Obs	Item	IDnum	InStock	BackOrd	CostPerUnit	Total	SalePrice	
1	Bird Feeder	LG088	3	20	\$5.00	23	\$8.25	

次に、Total の値が、InStock と BackOrd の現在の値に基づいて計算されます。Total の値は、PDV でのみ変更されます。

```
data work.update;
  set cert.invent;
  Total=instock+backord;
  SalePrice=(CostPerUnit*0.65)+CostPerUnit;
  format CostPerUnit SalePrice dollar6.2;
run;
```

図 7.13 プログラムデータベクトルと出力データセット

プログラムデータベクトル								
Item	IDnum	InStock	BackOrd	CostPerUnit	Total	SalePrice	N	ERROR
6 Glass Mugs	SB082	6	12	\$1.50	18	.	2	0

+      =      ↑

SASデータセットWork.Updateの出力								
Obs	Item	IDnum	InStock	BackOrd	CostPerUnit	Total	SalePrice	
1	Bird Feeder	LG088	3	20	\$5.00	23	\$8.25	

次に、SalePrice の値は、CostPerUnit の値に 0.65 を掛けて計算され、CostPerUnit の値に追加されます。これで、PDV にすべての新しい変数が入力されました。SalePrice の新しい値\$2.48 は PDV にあります。

```
data work.update;
  set cert.invent;
  Total=instock+backord;
  SalePrice=(CostPerUnit*0.65)+CostPerUnit;
  format CostPerUnit SalePrice dollar6.2;
run;
```

図 7.14 プログラムデータベクトルと出力データセット

プログラムデータベクトル								
Item	IDnum	InStock	BackOrd	CostPerUnit	Total	SalePrice	N	ERROR
6 Glass Mugs	SB082	6	12	\$1.50	18	\$2.48	2	0

+ (CostPerUnit X 0.65) =      ↑

SASデータセットWork.Updateの出力								
Obs	Item	IDnum	InStock	BackOrd	CostPerUnit	Total	SalePrice	
1	Bird Feeder	LG088	3	20	\$5.00	23	\$8.25	
2	6 Glass Mugs	SB082	6	12	\$1.50	18	\$2.48	

\*イタリック体のフォントは、特に指定がない限り、テキストで変更されたものです。

i>

RUN ステートメントは、DATA ステップループの終了を示します。DATA ステップの最後で、PDV の値が 2 番目のオブザベーションとしてデータセットに書き込まれます。図 7.14 に示すように、WORK.UPDATE データセットには、RUN ステートメントの後に 2 つのオブザベーションがあります。

```
data work.update;
  set cert.invent;
  Total=instock+backord;
  SalePrice=(CostPerUnit*0.65)+CostPerUnit;
  format CostPerUnit SalePrice dollar6.2;
run;
```

次に、\_N\_ の値が 2 から 3 に増加し、制御は DATA ステップの先頭に戻り、Item、IDnum、InStock、BackOrd、および CostPerUnit の値は保持されます。Total と SalePrice の値は欠損値にリセットされます。

SET ステートメントには、データセットの元の値を保持するための暗黙の RETAIN ステートメントが含まれていません。

```
data work.update;
  set cert.invent;
  Total=instock+backord;
  SalePrice=(CostPerUnit*0.65)+CostPerUnit;
  format CostPerUnit SalePrice dollar6.2;
run;
```

図 7.15 プログラムデータベクトルと出力データ



生データを読み込むとき、SAS は、次の例外を除いて、実行の各サイクルの開始時に DATA ステップの各変数の値を欠損値に設定します。

- RETAIN ステートメントで指定された変数
- sum ステートメントで作成される変数
- 自動変数

対照的に、SAS データセットから変数を読み込む場合、SAS は DATA ステップの実行の最初のサイクルの前にもみ値を欠損値に設定します。したがって、図 7.15 に示すように、変数は、新しい値が使用可能になるまでその値を保持します。図 7.15 では、DATA ステップの先頭で \_N\_ が 2 から 3 に増加すると、Total と SalePrice のみは欠損値に設定されます。CERT.INVENT データセットに由来する他の変数は、割り当てステートメントを通じて、または SET または MERGE ステートメントの次の実行を通じてのみ変更されます。SET または MERGE ステートメントのオプションを使用して作成された変数も、実行の 1 つのサイクルから次のサイクルまで値を保持します。