

SAS Technical News

Autumn 2001

*For Higher
Customer Satisfaction,
We Bridge
the SAS System
Between
Customer's World.*

CONTENTS

- 1** 特集 AppDev Studio バージョン2.0

- 9** 新刊マニュアルのご紹介

- 10** Q&A

- 14** SASトレーニングのお知らせ

- 16** 最新リリース情報

- 16** SASプログラマーズ・ワークショップ開催のお知らせ

特集

AppDev Studio バージョン2.0



1. はじめに

この秋、SASクライアントアプリケーションの統合開発環境である日本語版AppDev Studioバージョン2.0がリリースされます。この特集では、AppDev Studioバージョン2.0に含まれるwebAFソフトウェアを利用したJSPページの作成を中心に、AppDev Studio バージョン2.0(以下ADS)について、次の順番でご紹介します。

ADSの概要

JavaServer Pages (JSP)について

webAFソフトウェアによる、JSPを使用したクライアントアプリケーションの作成

2 ADSの概要

2.1 開発可能なアプリケーションの種類

ADSは、SASシステムを利用するクライアントアプリケーションを作成するための、統合開発環境です。ADSで開発可能なクライアントアプリケーションは多岐にわたります。下記はその一例です。

- ・サーバーサイドJava(サーブレット、JSP)を利用したWebアプリケーション
- ・ASPを利用したWebアプリケーション
- ・CGIを利用したWebアプリケーション
- ・Java アプレット
- ・Javaアプリケーション
- ・VBやVC++等のWindowsアプリケーション
- ・SAS/AFアプリケーション

2.2 ADSに含まれるソフトウェア

AppDev StudioはSASシステムを含む完結した開発環境を提供します。そのため、開発用のSASサーバーを用意する必要はありません。ADSに含まれるソフトウェアは以下のとおりです。

・webAFソフトウェア

webAFソフトウェアは、JavaServer PagesやJavaサーブレット、Javaアプレット、およびJavaアプリケーションを作成するためのビジュアル開発環境で、下記の特徴を持っています。

- ・SASサーバーを利用するためのクラス群が用意されているため、SASサーバーへのアクセスやSASデータの利用が簡単に実現可能
- ・JDBCやJavaBeansをサポートしている
- ・JavaServer PagesやJavaサーブレットといった、サーバーサイドJavaアプリケーションも作成可能
- ・Java2に対応しているため、Swingを利用して美しいJava GUIクライアントを作成可能

・webEISソフトウェア

webEISソフトウェアを使用すれば、GUIによって簡単にOLAP(Online Analytical Processing)アプリケーションを作成できます。webEISソフトウェアの特徴は下記のとおりです。

- ・プログラミングが不要なため、Javaの知識がなくても簡単にOLAPアプリケーションを作成可能
- ・作成されたアプリケーションは、JavaアプレットやJSPとしてWeb配信が可能
- ・ユーザーはOLAPアプリケーションをインタラクティブに操作可能

・その他のソフトウェア

ADSには、webAFソフトウェア、webEISソフトウェア以外に、以下のソフトウェアが含まれます。

- ・Base SASソフトウェア
- ・SAS/AFソフトウェア
- ・SAS/EISソフトウェア
- ・SAS/FSPソフトウェア
- ・SAS/GRAPHソフトウェア
- ・SAS Integration Technologiesソフトウェア
- ・SAS/IntrNetソフトウェア
- ・SAS/CONNECTソフトウェア
- ・SAS/SHAREソフトウェア

2.3 システム必要条件

AppDev Studio 2.0をサポートするOSは、以下のとおりです。ADSはあくまで開発環境であるため、サーバー系のWindows(Windows 2000 Server等)には対応していませんのでご注意ください。

- ・Windows 2000 Professional
- ・Windows NT Workstation (SP5以降)
- ・Windows 98 Second Edition
- ・Windows 95 (Winsock 2.0が必要)

その他の詳細なシステム必要条件については、米国SAS InstituteのWebページ (<http://www.sas.com/rnd/appdev/doc/reqs.htm>)をご参照ください。

2.4 本番用SASサーバーについての注意点

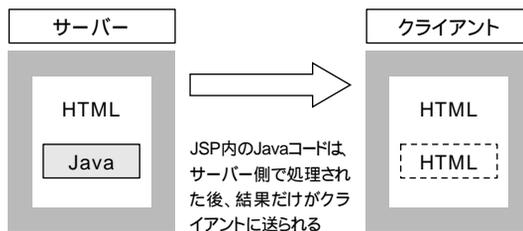
ADSは開発環境にすぎないため、ライセンス上、開発されたアプリケーションを利用するためにはSASサーバーを別途用意する必要があります。SASサーバーには、クライアント/サーバー間の接続方法に応じて、SAS/IntrNetソフトウェアまたはSAS Integration Technologiesソフトウェアが必要ですが、AppDev StudioやSAS/AFソフトウェアは必要ありません。しかし、使用するSASシステムの機能によっては、他のソフトウェア(SAS/GRAPHソフトウェアやSAS/STATソフトウェア等)が必要になります。SASサーバーとして使用できるSASシステムのバージョンは、8.1、8.2、その他バージョン6では、6.12 TS070、および6.12 TS065です。ただし、6.12 TS070と6.12 TS065は、英語版のみがリリースされています。したがって、日本語を扱う場合は、バージョン8のSASサーバーを使用する必要があります。

3 JSP(JavaServer Pages)入門

この特集の後半では、webAFソフトウェアを利用してJSPページを作成します。まず、JSPになじみのない方のために、JSPについて簡単に説明します。

3.1 JSPとは?

JSPとは、Javaプログラムが埋め込まれたHTMLページです。Javaプログラムの部分はサーバー側で処理され、クライアント(Webブラウザ)には処理によって動的に作成されたHTMLだけが送られます。Sun Microsystemsにより提唱されたJavaサーブレットの拡張で、動的なWebページの実現に広く利用されています。



3.2 JSPの利点

動的Webコンテンツ作成のための技術として、JSPには次のような利点があります。

・プログラミング環境

強力なオブジェクト指向言語であるJava言語が使用できます。再利用したいビジネスロジックはJavaBeansとして部品化しておくと、後で簡単に利用できます。

・カスタムタグライブラリのサポート

この新機能を利用するとJava言語を使用して、独自のタグを定義できます。カスタムタグをJSPページに埋め込むことにより、Javaの知識がなくても、タグを記述するだけで動的なページを作成できます。ADSには、SASシステムの機能にアクセスするための「sasads」で始まるカスタムタグが多数用意されています。

・仮想セッションのサポート

通常のHTTP接続はセッションレスな接続ですが、JSP(サーブレット)ではクッキーを利用した仮想セッションがサポートされています。

・移植性

Java Virtual Machine (JVM)がそのマシン上に用意されていれば、Javaプログラムはどのプラットフォームでも同様に動作します。このため、コンパイル済みのJavaプログラムも、異なるプラットフォームへ簡単に移植できます。

・拡張性

CGIプログラムは、リクエストのたびに新しいプロセスとして起動するため、クライアントのリクエストを受けて起動する際にリソースを大量に消費します。しかし、JSP(サーブレット)は、プロセスではなくスレッドを使用するため、リクエストあたりに使用するリソースが少なく済みます。

3.3 アプレット VS サーバーサイドJava

webAFでは、クライアント側で動作するJavaプログラム(Javaアプレット)と、サーバー側で動作するJavaプログラム(Javaサーブレット、JSP)の双方を開発可能です。ここでは、それぞれの利点を説明します。

・サーバーサイドJava(サーブレット、JSP)の利点

- ・クライアント側にプログラムをダウンロードする必要がないため、起動時間が短い
- ・クライアント側の動作環境(Javaプラグインのバージョン等)を選ばない
- ・アプレットにあるセキュリティ制限(サンドボックス環境)がない

・クライアントJava(アプレット)の利点

- ・Javaを利用した多機能なクライアントを作成できる
- ・Webサーバー以外にはサーバー側の設定(サーブレットコンテナ等)が不要

3.4 JSPタグ

JSPでは、通常のHTMLタグの他に、JSP独自のタグも使用します。ここでは、主要ないくつかのJSPタグについて説明します。

・表示(<% ~ %>)

<% ~ %>の間にある文字列(Stringオブジェクト)は、そのままクライアントに送られます。

例

```
<html>
  <body>
    <h1> <%= "こんにちは!" %> </h1>
  </body>
</html>
```

クライアントへ送られるHTML

```
<html>
  <body>
    <h1> こんにちは! </h1>
  </body>
</html>
```

ブラウザでの表示



[図1]

・Java言語の埋め込み(<% ~ %>)

<% ~ %>の間にある部分は、Javaプログラムとして、JSPページがリクエストされた際に実行されます。

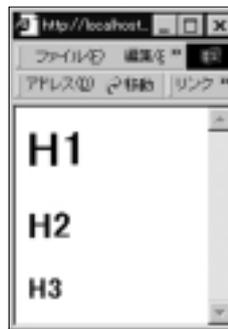
例

```
<html>
  <body>
    <% for (int i=1; i<4; i++) { %>
      <h<%= i %>> H<%= i %> </h<%= i %>><br>
    <% } %>
  </body>
</html>
```

クライアントへ送られるHTML

```
<html>
  <body>
    <h1>H1</h1><br>
    <h2>H2</h2><br>
    <h3>H3</h3><br>
  </body>
</html>
```

ブラウザでの表示



[図2]

・JSPページについての情報(<%@ page ~ %>)

<%@ page ~ %>の間には、JSPページで使用されている文字コードや、インポートする必要のあるクラス等、JSPページについての情報を記述します。

例

```
<%@ page language="java"
import="java.sql.*, java.util.*"
contentType="text/html; charset=SJIS" %>
```

・カスタムタグライブラリの情報

(<%@ taglib uri=" ~ " prefix=" ~ " %>)

JSPページで使用するカスタムタグライブラリとプリフィックスを宣言します。

例

```
<%@ taglib
uri=http://www.sas.com/taglib/sasads
prefix="sasads" %>
```

3.5 あらかじめ用意されているオブジェクト

JSPでは、いくつかのオブジェクトがあらかじめ定義されています。後述の例(ODS出力のストリーミング)ではoutオブジェクトが使用されています。これはクライアント(ブラウザ)への出力を表します。outオブジェクトのprintメソッドを利用して、クライアントに出力を送ることができます。

例

```
<%
    out.print("<HTML><BODY>TEST</BODY></HTML>");
%>
```

out以外にも、request、response、session等のオブジェクトがJSP内で使用できます。

4 webAFを使用した、JSPページの作成

この章では、以下の順で単純なJSPページを作成しながら、webAFソフトウェアの使用方法を説明します。4.1~4.3の基本編では、動的なコンテンツの作成にカスタムタグだけを使用するため、Java言語の知識は必要ありません。

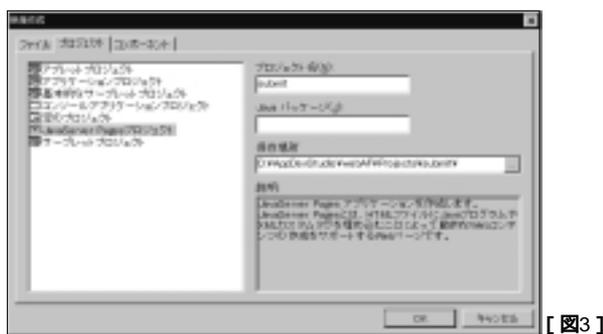
- 4.1 SASプログラムのサブミット
- 4.2 ODS出力のストリーミング
- 4.3 SASデータセットの表示

4.1 SASプログラムのサブミット

最初の例では、SASプログラムをサブミットし、プロシジャ出力を表示するJSPページを作成しながら、webAFソフトウェアの基本的な操作方法を説明します。

・新規プロジェクトの作成

- i) webAFソフトウェアを起動します。
- ii) [ファイル] [新規作成] を選択し、新規作成ダイアログを開きます。(図3)



[図3]

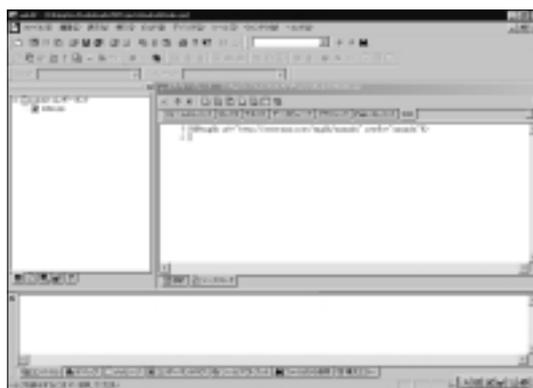
- iii) JavaServer Pagesプロジェクトを選択してプロジェクト名にsubmitと入力し、OKボタンを押します。
- iv) JSPオプションダイアログでは、デフォルトの設定のまま完了ボタンを押します。(図4)



[図4]

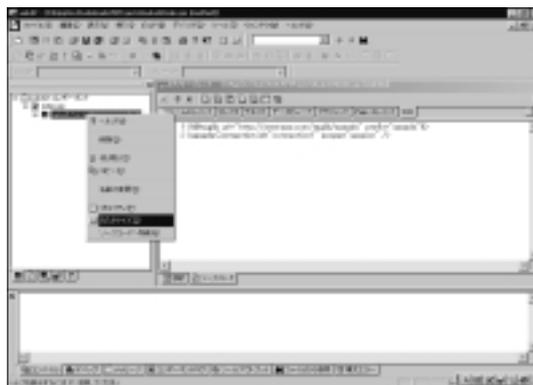
・SASサーバーへの接続

- i) 接続の設定を行う前に、SASサービスを起動しておきます。今回はIOMプロトコル(SAS独自プロトコル)を使用するため、IOMスプawnerを起動します。Windowsのスタートメニューから、[プログラム] [AppDev Studio] [サービス] [SAS V8] [SAS IOM Spawner開始] を選択してください。
- ii) Connectionタグを追加します。コンポーネントパレットのSASタブを選択し、ConnectionアイコンをJSPソースコードにドラッグ&ドロップします。(図5)



[図5]

- iii) Connectionタグの属性を設定します。画面左側のプロジェクトナビゲーターでConnectionタグを右クリックし、ポップアップメニューから[カスタマイズ] を選択してください。(図6)



[図6]

iv)Customizerダイアログを使用して、Connectionタグの属性を設定します。今回はIOMプロトコルを使用するため、接続タブを選択し、標準のサーバーの設定で「IOM(PC、Unix、またはMVS)」を選択します。(図7)他の項目はデフォルトのままかまいません。次にテストタブを選択し、接続テストを行ってからCustomizerダイアログを閉じます。ソースコードのConnectionタグに、属性が追加されていることを確認してください。



[図7]

・Submitタグの設定

- i)Submitタグをソースコードに追加します。コンポーネントパレットから人型のSubmitタグをドラッグし、ソースコードに追加します。
- ii)リモート接続ダイアログが開くので、そこで接続を選択します。先ほど設定した接続を使用するため、「定義済みの接続を使用する」を選択します。「このフレームにドロップされるすべてのリモートオブジェクトでこの接続を使用する」をチェックしておく、次回以降は接続を選択する必要がなくなります。(図8)



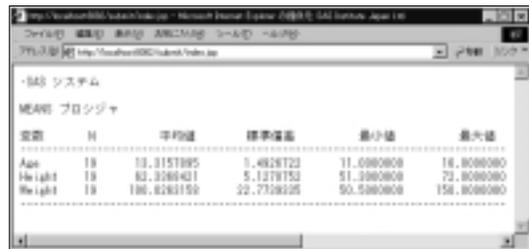
[図8]

- iii)Submitタグの属性を設定します。プロジェクトナビゲータ内でSubmitタグを右クリックしてプロパティを選択し、PropertySheetダイアログを開きます。display属性を変更して、SAS プロシジャの出力がWebページに出力されるようにします。デフォルトでは値がNONEになっているため、SASプログラムが実行されても、Webページには何も表示されません。ここでは、display属性の値をOUTPUTに変更します。(図9)その他の属性はデフォルトのままにして、PropertySheetダイアログを開けてください。



[図9]

- iv)SASコードを追加します。ソースコードエディタを使用して、Submitタグの内側にSASコードを記述します。
- v)必要なHTMLタグとJSPタグを追加します。JSPタグでcontentTypeを設定しないと、Webページが文字化けするので注意してください。
- vi)JSPを表示するために、Java Webサーバー(サーブレットコンテナ)を起動します。メニューから[ツール] [サービス] [Java Webサーバー開始]を選択します。
- vii)JSPのテストをします。メニューから[ビルド] [ブラウザで実行] [Internet Explorer]を選択すると、Internet Explorerが起動し、JSPページが表示されます。(図10)



[図10]

以上で、このJSPページは完成です。Java言語を使用せずに、カスタムタグだけでSASシステムを簡単に利用できることがお分かりいただけたでしょうか。この例では、アイコンのドラッグ&ドロップや、ダイアログでカスタムタグの設定を行いました。ソースコードエディタを使用してカスタムタグを編集することも可能です。

この例で作成したJSPのソースコード

```
<%@taglib uri="http://www.sas.com/taglib/sasads"
prefix="sasads"%>
<%@page language="java"
contentType="text/html;charset=SJIS" %>
<html>
<body>
<sasads:Connection id="connection1" scope="session"
persistedName="Custom Connection">
serverArchitecture="IOM" port="5310"
accessMethod="IOM" />
```

```
<sasads:Submit id="submit1" connection="connection1"
display="OUTPUT">
options nocenter nodate;
proc means data=sashelp.class;
run;
</sasads:Submit>
</body>
</html>
```

4.2 ODS出力のストリーミング

ここでは、SASシステムバージョン8のODS出力をブラウザで表示する方法を取り上げます。SASサーバーはサーブレットコンテナに対してODS出力を送る必要がありますが、ここではソケット通信を使用してODS出力の送信を行います。

・新規プロジェクトの作成

先ほどの例と同様に、新規プロジェクトを作成します。この例のプロジェクト名は「ods」とします。

・Connectionタグの追加と設定

SASシステムを利用するためには、必ずSASサーバーとの接続が必要になります。最初の例と同様に、Connectionタグを追加して属性を設定します。

・サーバーソケットをセットアップするためのJavaコードを追加します。ソースコードに<% ~ %>を利用して、ソケットリスナーをセットアップするためのJavaプログラムを追加します。

```
<%
// SocketListenerオブジェクトを作成します
com.sas.servlet.util.SocketListener socket =
new com.sas.servlet.util.SocketListener();
// 使用するポート番号を変数portに代入します
int port = socket.setup();
// SocketListenerを開始します
socket.start();
%>
```

・Submitタグの追加と属性の設定

Submitタグを追加し、タグ内にSASプログラムを記述します。ODSHTMLの出力先にはソケットを指定します。

```
<sasads:Submit id="submit1"
connection="connection1">
filename sock SOCKET 'localhost:<%=port%>';
ods html body=sock ;
title;
proc means data=sashelp.class;
run;
ods html close;
</sasads:Submit>
```

・ブラウザへの出力とソケットの終了

SASサーバーから受け取ったHTMLを、outオブジェクトを利用してブラウザに出力します。<% ~ %> 内に、以下のJavaプログラムを記述します。

```
<%
// ソケットからの入力をクライアント(ブラウザ)に送信します
socket.write(out);
```

```
// SocketListenerを終了します
socket.close();
%>
```

・必要なJSPタグを追加します

以下のJSPタグで、このページで使用される文字コードなどを宣言します。

```
<%@ page language="java"
contentType="text/html;charset=Shift_JIS" %>
```

・Java Web サーバー(サーブレットコンテナ)を起動します。

メニューバーから[ツール] [サービス] [Java Web サーバー開始] を選択し、サーブレットコンテナを起動します。

・JSP表示のテストをします。

メニューバーから[ビルド] [ブラウザで実行] [Internet Explorer] を選択し、表示をテストします。(図11)

変数	N	平均値	標準偏差	最小値	最大値
Age	19	13.3157865	1.4506722	11.0000000	16.0000000
Height	19	62.3969421	5.1270752	51.3000000	72.0000000
Weight	19	100.0263158	22.7799335	50.5000000	150.0000000

[図11]

以上で、このJSPページは完成です。若干のJavaプログラミングが必要ですが、ODS出力も簡単にWebページに出力できることが、おわかりいただけたでしょうか。

この例で作成したJSPのソースコード

```
<%@taglib uri="http://www.sas.com/taglib/sasads"
prefix="sasads"%>
<%@ page language="java"
contentType="text/html;charset=SJIS" %>
<sasads:Connection id="connection1" scope="session"
persistedName="Custom Connection"
serverArchitecture="IOM" port="5310"
accessMethod="IOM" />
```

```
<%
// SocketListenerオブジェクトを作成します
com.sas.servlet.util.SocketListener socket =
new com.sas.servlet.util.SocketListener();
// 使用するポート番号を変数portに代入します
int port = socket.setup();
// SocketListenerを開始します
socket.start();
%>
```

```
<sasads:Submit id="submit1"
connection="connection1">
filename sock SOCKET 'localhost:<%=port%>';
ods html body=sock;
proc means data=sashelp.class;
run;
ods html close;
```

```

</sasads:Submit>

<%
// ソケットからの入力をクライアント(ブラウザ)に送信します
socket.write(out);
// SocketListenerを終了します
socket.close();
%>

```

4.3 SASデータセットの表示

ADSが提供するカスタムタグを利用すると、簡単にSASデータセットを表示できます。テーブルの外見も、Customizerダイアログを使用して簡単に変更できます。この例では、データセットの表示と、テーブルのカスタマイズを行います。

・新規プロジェクトの作成

新規プロジェクトを作成します。ここではプロジェクト名を「table」とします。

・Connectionタグの追加と属性の設定

SASサーバーを利用するために、Connectionタグを利用します。最初の例と同様に、タグを追加し、属性を設定します。

・DataSetタグの追加

コンポーネントパレットのSASタブにあるDataSetInterfaceをドラッグして、ソースコードに追加します。リモート接続ダイアログでは、先ほど設定したconnection1を選択してください。

・DataSetタグの属性の設定

画面左側のプロジェクトナビゲーターでDataSetタグを右クリックし、ポップアップメニューから「カスタマイズ」を選択します。Customizerダイアログが開くので、データセット名に「SASHELP.CLASS」と入力します。その他の変更は必要ありません。ソースコードのDataSetタグにdataSet属性が追加されたことを確認してください。

・Tableタグの追加

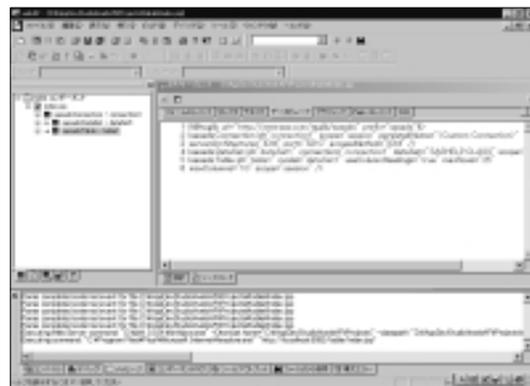
コンポーネントパレットのデータビューアタブにあるTableをドラッグして、ソースコードに追加します。

・Tableタグの属性の設定

プロジェクトナビゲーターのTableタグを右クリックして、ポップアップメニューから「カスタマイズ」を選択します。Customizerダイアログが開くので、Data Sourceに先ほど追加したDataSetタグのIDである「dataSet1」を指定します。(図12)Customizerダイアログを閉じたら、ソースコードのTableタグにmodel属性が追加されていることを確認してください。また、プロジェクトナビゲーターでも、モデル(DataSet)とビューア(Table)が関連付けられている事が矢印によってわかります。(図13)



[図12]



[図13]

・Java Web Serverの起動

表示のテストをするために、Java Web Server(サーブレットコンテナ)を起動します。メニューバーから「ツール」[「サービス」][「Java Web Server開始」]を選択します。

・表示のテスト

メニューバーから「ビルド」[「ブラウザで実行」][「Internet Explorer」]を選択して、表示のテストを行います。(図14)



[図14]

・表のカスタマイズ(罫線の追加、数字の右揃え)

テーブルは表示できましたが、デフォルトでは罫線もなく、シンプルなテーブルです。そこで、表をカスタマイズしてみましょう。まずは罫線を追加します。先ほどと同様にTableタグのCustomizerダイアログを開き、TableタブのBorder widthを2に変更します。(図15)



[図15]

次にAge(年齢)、Height(身長)、Weight(体重)の数字を右揃えにします。今度はRow/Columnタブを使用します。左側のColumn Formatsを右クリックして、ポップアップメニューから「Add column format」を選択します。(図16)



[図16]

Customizerダイアログの右側に属性が現れるので、それぞれ以下の図のように設定します。Formatでは、DTタグのALIGN属性を「RIGHT」に設定します。また、データの値を入りたい場所には「##」を挿入します。(図17)



[図17]

設定が終わったら、先程と同様にブラウザで確認してみましょう。以下の図のように罫線が追加され、数値が右揃えで表示されます。(図18)



[図18]

以上で、このJSPページは完成しました。基本的なHTMLの知識は必要ですが、Customizerダイアログを使用して、簡単に表の外見を変更できる事がわかりただけかと思えます。

この例で作成したJSPのソースコード

```
<%@taglib uri="http://www.sas.com/taglib/sasads"
prefix="sasads"%>
<sasads:Connection id="connection1" scope="session"
persistedName="<Custom Connection>"
serverArchitecture="IOM" port="5310"
accessMethod="IOM" />
<sasads:DataSet id="dataSet1"
connection="connection1"
dataSet="SASHELP.CLASS" scope="session" />
<HTML><BODY>
<sasads:Table id="table1" model="dataSet1"
useColumnHeadings="true"
```

```
maxRows="25"
  maxColumns="10" scope="session" borderWidth="2" >
<sasads:ColumnFormat startColumn="3"
repeatColumn="1" startRow="1"
repeatRow="1" >
<TD ALIGN="RIGHT">##</TD>
</sasads:ColumnFormat>
</sasads:Table>
</BODY></HTML>
```

5 おわりに

この特集では、webAFソフトウェアを使用したJSPページの作成を中心に、AppDev Studio バージョン 2.0についてご紹介しました。webAFソフトウェアには、便利なクラスやJSPカスタムタグが多数用意されているため、SASクライアントアプリケーションをすばやく開発できます。また、紙面の都合で詳しくご紹介できませんでしたが、webEISソフトウェアを使用すると、GUIで簡単にOLAPアプリケーションを作成できます。まだ日本語版はリリースされておりませんが、SASクライアントの統合開発環境として、AppDev Studio バージョン2.0にぜひご期待ください。

New Publications

新刊マニュアルのご紹介

SASインスティテュートジャパンおよび SAS Institute より発行された新刊マニュアルをご紹介します。なお、こちらに記載する価格はすべて税抜きです。ご購入は、専用の申込用紙に必要事項をご記入の上、FAXで下記「マニュアル販売係」までお申し込みください。

Getting Started with AppDev Studio,
Second Edition

注文番号: 58135(英語版)

価格: 3,100円

対象: 初級者

本書では、AppDev Studioソフトウェアを用いてシンクライアントアプリケーションを開発し、運用するための技術について記載しています。情報を伝達するための技術的なアプローチを選び、かつそれらの有効性および影響を理解するのに役立ちます。具体的な例を用いながら、ODS、CGI/HTMLアプリケーション、Javaアプレットおよびアプリケーション、Java サーブレット、そしてJavaServer Pagesといった技術の詳細が記載しています。これらの技術を使用することで、SASシステムによる情報伝達の可能性を広げることができます。本書はSASシステムにより、サーバから広範囲に情報を配信するアプリケーションの開発、運用および保守を行うアプリケーション開発者や情報技術管理者を対象としています。AppDev Studioソフトウェアに関連した下記プロダクト等を利用する開発者には有効な一冊です。

- SAS/AFソフトウェア
- SAS/EISソフトウェア
- SAS/IntrNetソフトウェア
- webAFソフトウェアおよびwebEISソフトウェア

マニュアル販売係

T E L 03-3533-3835

F A X 03-3533-3781

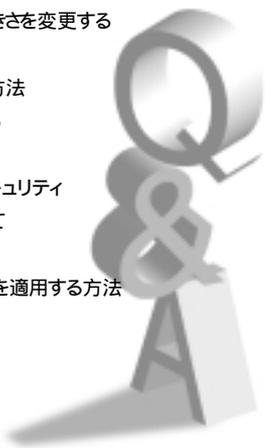
E-mail booksale@jpn.sas.com

なお、マニュアル申込用紙および最新の Publications Catalog (マニュアル案内パンフレット)は弊社ホームページ (<http://www.sas.com/japan/manual/>)にて公開しております。

2001年11月より、個人購入の場合に限り、代金引換での購入が可能となりました。詳細は弊社ホームページをご覧ください。上記マニュアル販売係までお問い合わせください。

Q&A

SAS/ACCESSソフトウェア ODBCインタフェースでMS SQL Serverにアクセスするとき、データベース名をプログラムで指定する
 プリファレンスから独自のODSスタイルテンプレートを使用する方法
 4バイトのテキストデータを2バイトのバイナリコードに圧縮する方法
 SAS/GRAPHソフトウェアで箱ひげ図の大きさを変更する
 横棒グラフの統計量のラベル変更
 タイトルやフットノートをHTMLリンクにする方法
 画面から入力した値をプログラムで利用する
 拡張エディタを呼び出すコマンド
 SAS/IntrNetソフトウェア リリース8.2のセキュリティ
 大規模な正確検定を行うオプションについて
 ワイブル分布の推定方法
 SASシステム リリース8.2でライセンス情報を適用する方法



Q バージョン 6のSAS/ACCESSソフトウェア SYBASE and SQL Serverインタフェースを使ってMS SQL Serverにアクセスしていましたが、バージョン 8eではMS SQL Serverがサポートされなくなったため、SAS/ACCESSソフトウェア ODBCインタフェースに変更しました。SAS/ACCESSソフトウェア ODBCインタフェースでも、データベース名をプログラムから指定できますか？

A 下記の方法で、プログラムからデータベース名を指定できます。

LIBNAMEステートメント

REQUIREDオプションを使ってデータベース名を指定できます。

```
libname sql_s ODBC
required="DSN=sqlserver;UID=user_id;PWD=password;
DATABASE=Pubs";
data mydbl;
set sql_s.authors;
run;
```

SQLパスルー

connect to odbc("dsn=データソース名; database=データベース名")
 のように指定できます。

```
proc sql;
connect to ODBC ("dsn=sqlserver; database=Pubs");
create view work.mydbl as
select * from connection to ODBC
(select * from dbo.authors);
disconnect from ODBC;
quit;
```

注:Microsoft社 SQL Serverは、SASシステムバージョン8eにおいてSAS/ACCESSソフトウェアソフトウェアODBCインタフェースとSAS/ACCESSソフトウェア OLE DBインタフェースによってサポートされるようになりました。

Q プリファレンスで選択できるスタイルに、独自のテンプレートを加えることはできますか？

A SASシステム バージョン 8eでは、ODS (Output Delivery System)を利用して、プロシジャなどの結果をHTMLで出力することが可能となりました。また、メニュー「ツール | オプション | プリファレンス | 結果」での設定により、常にHTML出力を生成させることも可能です。

このメニュー「プリファレンス | 結果」で選択できるスタイルテンプレートは、デフォルトではSASシステムが標準で提供するものになります。独自のスタイルテンプレートを利用可能にするには、次のように、選択リストにスタイルテンプレートの名前を登録する必要があります。

- 1.メニューから「ソリューション | アクセサリ | レジストリエディタ」を選択します。
- 2.レジストリエディタで「Sas_registry\Preferences\Styles」を選択し、右クリックで表示されるサブメニューから「キーの作成」を選択します。
- 3.キーの名前として、登録するテンプレートの名前を入力します。

上記の登録作業後、メニュー「プリファレンス | 結果」にて選択できるようになります。

Q コード変換テーブルを作成したいのですが、ノートパッドで作成したため、漢字コードがテキスト4バイトの状態です。これを16進 2バイトのデータセットに変換する方法はありますか？

例: '6181'(4バイト長) '6181'x(2バイト長)

A KCVT関数などを使ってコード変換を行なう場合、「2バイト(コード体系によって異なる)」の変換テーブルを用意する必要があります(「SAS Technical Report J-121 日本語/DBCS機能使用の手引き」(注文番号:10074)の「DBCS TABプロシジャ」の項などを参照)。SASシステム自体が提供している関数などで、4バイトから2バイトへの圧縮機能を持つものは、残念ながらありません。ただし、プログラムとしていったん外部ファイルに出力してから実行すれば、上記の処理を実現できます。

以下のプログラムを参考にしてください。

```
/* サンプルデータセット(SAMPLE1) */
DATA SAMPLE1 ;
LENGTH ibm pcibm $4 ;
INPUT ibm pcibm ;
DATALINES ;
6181 a4d2
6182 a4d3
6183 a4d4
;
RUN ;

/* 変換用プログラム */
DATA _NULL_ ;
SET SAMPLE1 END=EOF;
FILE 'c:\temp\convert.sas' ;
IF _N_ = 1 THEN DO ;
```

```

      PUT "DATA custtab; " ;
      PUT "      LENGTH ibm pcibm $2;" ;
END ;
PUT "      ibm = '      ' ibm +(-1) 'x' ;" ;
PUT "      pcibm = '      ' pcibm +(-1) 'x' ;" ;
PUT "      output;" ;
IF EOF THEN DO ;
      PUT "RUN ;" ;
END ;
RUN ;

```

上記のプログラムを実行すると、「C:\temp\convert.sas」というファイル名で、4バイトから2バイトへの圧縮を行なうプログラムが生成されます。このプログラムを実行すれば、圧縮が可能です。

Q SAS/GRAPHソフトウェアで箱ひげ図を作成する場合、symbol ステートメントで「i=BOX」と指定しますが、その際に、作成する箱ひげ図(全体)の大きさを指定できないでしょうか。「i=BOX」を指定しているsymbolステートメントで、オプション height や width の値を大きくしても、出力される箱ひげ図は大きくなりません。

A SASシステム リリース 6.12では、Gplot プロシジャで生成する箱ひげ図の大きさを、オプションなどで指定(変更)することはできません。

ただし、SAS/GRAPHソフトウェアのANOTATE機能を利用して、上記質問のご希望に近いグラフを作成することはできます。こちらについては米国本社のWEB上にサンプルプログラムがありますので、参照してください。

該当WEBサイト

http://www.sas.com/service/techsup/sample/sample_graph.html
該当ページ内の項目、「Control Attributes of a Box Plot with Macro」を選択すると、サンプルプログラムが表示されます。

なお、SASシステム バージョン 8eからは、新規オプションとして、BWIDTH オプションが追加されています。このオプションにより、今回の質問にある、箱の部分に関する指定(変更)が可能となります。

また、箱ひげ図作成専用の「BOXPLOT プロシジャ」が、SAS/STATソフトウェアに追加されています。

プログラム例:

```

/* **** テストデータ作成 **** */
data datal;
  do group=1 to 3;
    do i=1 to 30;
      y=rannor(12345);
      output;
    end;
  end;
run;

/* **** 箱ひげ図(標準)作成 **** */
axis1 offset=(20,20) order=(1 2 3);
proc gplot data=datal;
  plot y*group /haxis=axis1;
  symbol1 i=box;

```

```

run;
quit;

/* **** 箱ひげ図(標準)作成 **** */
proc gplot data=datal;
  plot y*group /haxis=axis1;
  symbol1 i=box BWIDTH=10 WIDTH=5 HEIGHT=10;
run;
quit;

/* **** 箱ひげ図(BOXPLOTプロシジャ)作成 **** */
proc boxplot data=datal;
  plot y*group /haxis=axis1
  BOXWIDTH=5 HEIGHT=5;
run;
quit;

```

Q GCHARTプロシジャを利用して横棒グラフを作成する際に、統計量を出力しています。この統計量のラベルを任意に変更することはできますか?

A SASシステム バージョン 8eから、sumlabel= オプション等を利用して、統計量に任意のラベル名をつけることができるようになりました。

プログラム例:

```

proc gchart data=sasuser.carsales;
  hbar person / sumvar=total
               sum freq mean
               sumlabel = '売上合計'
               freqlabel = '販売台数'
               meanlabel = '平均単価';
run;
quit;

```

統計量のラベルを変更するオプションには、以下のものがあります。

- sumlabel
- freqlabel
- meanlabel
- percentlabel
- cfreqlabel
- cpercentlabel

注: sumlabel=オプションを適用できるプロシジャは、GCHARTプロシジャのみです。ステートメントでは、HBAR、およびHBAR3D、VBAR、VBAR3Dの各ステートメントで適用できます。

Q ODS機能を使用してHTMLファイルを出力する際、タイトルやフットノートを任意のホームページへのリンクにしたいのですが、方法を教えてください。

A TITLEステートメントや、FOOTNOTEステートメントでLINK=オプションを使用すれば、リンク先を指定できます。以下の例をご参照ください。

```

title link="http://www.sas.com/japan/"
"SAS Institute Japan";
footnote link="http://www.sas.com/japan/"
"SAS Institute Japan";

ods html file='c:\temp\test.html' ;
proc means data=sashelp.class;
run;
ods html close;

```

Q ユーザに対して値の入力を要求する画面を表示するようなプログラムの作成を考えています。どのようにすればよいでしょうか。

A マクロ機能の %WINDOW ステートメントを使用することによって、簡単な画面を表示させることができます。また、その中で値の入力を求め、その値をマクロ変数に割り当てることもできます。

以下の例では、ユーザに対して名前を入力を求め、その値をマクロ変数 NAME に割り当てています。

```

/* マクロを定義します。 */
%macro getname;
  %global name;
  %let name=;

  /* マクロウィンドウを定義します。 */
  %window macwin color=white columns=64 icolumn=8
  irow=8 rows=12
  #2 @12 "Welcome to the SAS System Release
&sysver.."
  attr=highlight color=blue
  #4 @20 "Your platform: "
  "&sysscp" color=orange
  #5 @12 "Input your name(1-16): "
  name 16 attr=underline required=yes;

  /* マクロウィンドウを表示します。 */
  %display macwin;
%mend;

/* マクロを呼び出します。 */
%getname;

/* 入力で得た値を使うことができます。 */
%put your name: &name;

```

Q プログラムエディタを表示させるときはPGMコマンドを使用しますが、バージョン 8eで拡張エディタを表示させたいときは、どのようにしたらよいでしょうか？

A メニューから「表示」「拡張エディタ」を選択することで表示できます。また、コマンドバーに「WEDIT」と入力すると、拡張エディタを表示できます。

Q ブラウザとSASアプリケーションサーバー間に、セキュリティ機能を設定したいと考えています。どのような設定が可能ですか？

A ブラウザとWebサーバー間、またWebサーバーとSASアプリケーションサーバー間で、さまざまなセキュリティを設定することができます。

SSLの暗号化をしたHTTPSをサポートしているので、データを暗号化できます。ファイアウォールを設定している場合でも、Webサーバーからアプリケーションサーバーの往復時に使用するポート番号を同一のものにできます。

設定方法:

broker.cfg ファイルの使用するサービスの中で次の記述を有効にします。

```
FullDuplex True
```

SASアプリケーションサーバー上で稼動するプログラムを使用して、ブラウザから入力したユーザーIDとパスワードにより、そのサーバー上の認証を行なうことが可能です(初期設定は、SASアプリケーションサーバーの起動ユーザー権限になっています)。

設定方法:

1. 起動時に使用する「appsrv.sas」(設定によって名前は異なる)内の APPSRV プロシジャに「AUTHオプション」「ADMINPWオプション」を追加します。

```
PROC APPSRV AUTH=HOST ADMINPW='admin' ... ;
(プログラム行) ;
RUN ;
```

- AUTHオプションは「NONE(デフォルト)あるいはHOST」の指定
- ADMINPWオプションはAUTHオプション指定時は必須です。

2. 同じく「appsrv.sas」内の APPSRV プロシジャに「REQUESTステートメント」を追加します。

```
PROC APPSRV AUTH=HOST ADMINPW='xxxx' ... ;
(プログラム行) ;
REQUEST LOGIN=SAMPLE.WEBLOGIN.SAS ;
(プログラム行) ;
RUN ;
```

マシンの認証に必要な「ユーザー名」「パスワード」を与えるための入力フォームを起動するために必要です。

この情報は同一セッション内でのみ有効です。

WindowsNTの場合、ブラウザから指定するユーザーの権限に、「高度なユーザー権利」の「バッチジョブとしてログオン」が割り当てられていなければなりません。

また、PROC APPSRV によってこの機能を使用するためには、「高度なユーザー権利」の「オペレーティングシステムの一部として機能」権限が割り当てられている必要があります。

Q 大規模なデータで正確検定を行うと、何時間実行しても実行中のまま結果が戻ってこない場合があります。なにか回避策はありませんか？

A SASシステム バージョン 8eから、大規模な問題の正確検定に関して、次のようなオプションがFREQ、NPAR1WAYプロシジャのEXACTステートメントに追加されました。

•MAXTIME= オプション

処理時間の上限を設定するオプションです。設定された時間より計算時間が長くなる場合には、途中で計算をストップさせます。バッチ処理などで正確検定を行う場合、実行時間を制限できるようになります。

•MCオプション

コンピュータの疑似乱数(モンテカルロシミュレーション)によって正確検定を行なう(正確検定のp値の近似値を求める)ためのオプションです。すべてのデータを数え上げる正確検定の計算が不可能に近い場合に、利用できます。

MAXTIME=オプション、MCオプションに関する詳細は、「SAS System Version 8 OnlineDoc」のFREQプロシジャ、NPAR1WAYプロシジャの章をご参照ください。

Q ワイブル分布のパラメータ推定を行いたいのですが、どのプロシジャを用いればよいのでしょうか？

A SASシステム リリース 8.2では、SAS/STATソフトウェアのLIFEREGプロシジャの利用が一番簡便な方法です。LIFEREGプロシジャでは、パラメータを最尤法により推定します。次の例をご参照ください。

プログラム例:

```
/* *****サンプルプログラム***** */
/* Scale/パラメータがs、Shape/パラメータが2の */
/* ワイブル分布に従う乱数列の発生 */
data weibull;
do i=1 to 100;
x=5*ranexp(123)**(1/2);
output;
end;
run;

/* LIFEREGプロシジャ */
proc lifereg data=weibull;
model x= /d=weibull ;
/* 分布としてD=オプションでWEIBULLを指定 */
run;
/* *****ここまで***** */
```

出力(一部抜粋)

Analysis of Parameter Estimates						
Parameter	DF	Estimate	Standard Error	95% Confidence Limits	Chi-Square	Pr > ChiSq
Intercept	1	1.5898	0.8520	1.4898 1.6898	931.89	<.0001
Scale	1	0.4943	0.8388	0.4243 0.5760		
Weibull Scale	1	4.8970	0.2549	4.4221 5.4229		
Weibull Shape	1	2.0290	0.1570	1.7982 2.9571		

上記の出力中、「Weibull Scale」、「Weibull Shape」の行が、それぞれScale(尺度)パラメータ、Shape(形状)パラメータに関する出力です。

SASシステム リリース 8.2のLIFEREGプロシジャでは、上記以外にも正規分布、指数分布、ロジスティック分布などのパラメータ推定値、標準誤差、信頼区間を計算することができます。詳細については、「SAS/STAT Software: Changes and Enhancements, Release 8.2」(注文番号: 58309)の「Chapter 6.The LIFEREG Procedure」の項をご参照ください。

Q SASシステム リリース6.12で提供されている「ライセンス情報更新プログラム」はリリース8.2にはないのでしょうか？

A SASシステム リリース 8.2では、ライセンス情報の適用方法が変更されているため、「ライセンス情報更新プログラム」は残念ながら提供されておりません。以下の要領で、適用してください。

【事前準備】

弊社から送付した「SASシステム リリース8.2」用のライセンス情報(SETINIT)を、任意のフォルダに「SETINIT.SSS」というファイル名で保存してください。

例: D:\TEMP2\SETINIT.SSS

(SETINIT.SASではなく、拡張子は「.SSS」になります)

「D:\TEMP2」は説明のため便宜的に使っています。お客様の環境によって異なります。

【拡張子「SSS」が関連付けされている場合】

「SETINIT.SSS」をWindowsのエクスプローラ等で見た場合、ファイルの種類が「SASシステムセットユニット」と表示されていれば、拡張子「SSS」に関連付けがされています。この場合、以下の方法で適用できます。

1. エクスプローラ上で「SETINIT.SSS」を選択し、「右クリック」をしてください。
2. 下記の図のように「SAS V8のライセンス情報を更新する」というメニューが表示されます。



3. 「SAS V8のライセンス情報を更新する」を選択するとライセンス情報更新処理が行われ、同じフォルダに「SETINIT.LOG」というファイルが生成されます。このファイルで結果をご確認ください。

【拡張子「.SSS」が関連付けられていない場合】

「スタート」=>「ファイル名を指定して実行」にて以下のコマンドを入力してください。

```
!sasroot%NLS%JA%SAS -SYSIN d:%temp2%setinit.sss -CONFIG
!sasroot%NLS%JA%SASV8.CFG -LOG d:%temp2%setinit.log
```

上記の「!sasroot」には実際のインストール先のフォルダを記述してください。デフォルトでインストールしていただいている場合、以下の通りになります。

```
C:%PROGRA 1%SASINS 1%SAS%V8%NLS%JA%SAS -SYSIN
d:%temp2%setinit.sss -CONFIG
C:%PROGRA 1%SASINST 1%NLS%JA%SASV8.CFG -LOG
d:%temp2%setinit.log
```

ライセンス情報更新処理が行われ、「d:%temp2%setinit.log」というファイルが生成されます。このファイルにて結果をご確認ください。

今回取り上げました内容以外にも、SAS システムに関するFAQは次のURLにてご参照いただけますので、ご利用ください。

<http://www.sas.com/japan/service/technical/faq/index.html>



SAS Training

SAS トレーニングのお知らせ

SASインスティテュートジャパンでは、2001年8月より定期トレーニングを一新し、SASシステム リリース8.2を使用したトレーニングを行っています。それらのコースに加え、この秋から定期トレーニングコースを追加します。また臨時トレーニングコースも開催します。

新トレーニングコースのご案内

この秋より、下記2コースを定期トレーニングとして開始します。

「Enterprise Minerソフトウェア操作入門」コース

受講対象 データマイニングを業務で活用される方
前提知識 「SASプログラミング: Basic I」コース(または「初級プログラミング」コース)、および「SASによるデータ解析入門」コースを受講済みか、同程度の知識のある方
学習内容 本コースでは、「データマイニング」の基本的な概念や、データマイニングを効率よく進めるための方法論である「SEMMAプロセス」について紹介します。また、SASの強力なデータマイニングツールであるEnterprise Minerソフトウェアの基本操作を、実習をまじえながら習得していただきます。
価格/日程 2日間コース 55,000円(チケット捺印数3)
 東京会場:11月19日(月)~ 20日(火)

「時系列データ解析入門」コース

受講対象 売上予測・需要予測など、時系列データを用いた将来予測を、業務で行う方
前提知識 「SASプログラミング: Basic I」コース(または「初級プログラミング」コース)、および「SASによるデータ解析入門」コースを受講済みか、同程度の知識のある方
学習内容 本コースは、時系列予測の基本的な考え方と、SAS/ETSソフトウェアや時系列予測システムを利用した分析方法を習得することを目的としています。11、指数平滑化法、およびARIMAモデル等について解説し、時系列データを用いたモデル構築の実習を行います。
価格/日程 1日間コース 38,000円(チケット捺印数2)
 東京会場:11月16日(金)

臨時トレーニングコースのご案内

皆さまのご要望にお応えして、下記3コースを臨時開催します。

「実践データハンドリング演習」コース

受講対象 より実用的なプログラミングを学びたい方
前提知識 「SASプログラミング: Basic II」コース(または「上級プログラミング」コース)を受講済みか、同程度の知識のある方
学習内容 「SASプログラミング: Basic I」「Basic II」コースの仕上げとして、明日からの業務で利用可能なSASプログラミング例を多数まじえて、実用性のあるデータハンドリングや実践的なプログラミングテクニックを解説します。
価格/日程 1日間コース 30,000円(チケット捺印数2)
 東京会場:11月9日(金) 大阪会場:12月5日(水)

「ODS実践」コース

受講対象 実践的なODS(Output Delivery System)を学習したい方
前提知識 「SASプログラミング: Basic II」コース(または「上級プログラミング」コース)、および「ODS入門」コースを受講済みか、同程度の知識のある方
学習内容 本コースでは、ODSを利用した実践的なアプリケーション(システム)を作成することを目的としています。初歩アプリケーション作成(基本操作での作成)、テンプレート機能概要、テンプレートを使用した応用アプリケーション作成、またSAS/IntrNetソフトウェアとの連携などについて解説します。
価格/日程 2日間コース 60,000円(チケット捺印数3)
 東京会場:11月26日(月)~27日(火)
 大阪会場:12月11日(火)~12日(水)

「SASによるWeb配信」コース

受講対象 SASバージョン6に関しての知識があり、バージョン8の拡張点について学習したい方、またはWeb配信をご検討中および興味のある方
前提知識 「SASプログラミング: Basic I」コース(または「初級プログラミング」コース)、「SASプログラミング: Basic II」コース(または「上級プログラミング」コース)を受講済みか、同程度の知識のある方
学習内容 本コースでは、バージョン8からの新機能であるODS(Output Delivery System)とSAS/IntrNetソフトウェアを利用して、シンクライアントを実現する方法について紹介します。またRDBMSへのアクセスやSAS/CONNECTソフトウェアによるSASシステムの非同期実行についても解説します。
価格/日程 3日間コース 100,000円(チケット捺印数5)
 東京会場:11月12日(月)~14日(水)
 なおこちらのコースをお申し込みの場合、「5ポイントチケット(95,000円)」を同時にお申し込みいただくお得です。

「SASによるWeb配信」コースにご参加の方には、ご希望によりWindows版SASシステムの最新リリース 8.2のライセンスを2ヶ月無償で貸し出しいたします。ただし、この場合弊社と貸し出しに関する契約書に捺印していただく必要があります。この件に関してご不明な点がございましたら弊社トレーニング担当までご連絡ください。

日程が決定した定期トレーニングコース

開催日程が未定だった下記2コースの詳細が決定しましたので、お知らせします。

「臨床データマネジメント」コース

前提知識 ・「SASプログラミング: Basic I」コース(または「初級プログラミング」コース)
 ・「SASプログラミング: Basic II」コース(または「上級プログラミング」コース)
 ・「マクロ言語入門」コース
 上記3コースを受講済みか、同程度の知識のある方
学習内容 製薬業の臨床開発におけるデータマネジメント業務に焦点を当て、実業務に対応した形式でSASの利用方法を紹介します。データ作成から解析データの作成までのデータ管理、およびデータ加工に関するプログラミング技法についてご紹介します。
価格/日程 2日間コース 60,000円(チケット捺印数3)
 東京会場:11月21日(水)~22日(木)

「解析表作成」コース

前提知識 ・「SASプログラミング: Basic I」コース(または「初級プログラミング」コース)
 ・「SASプログラミング: Basic II」コース(または「上級プログラミング」コース)
 ・「マクロ言語入門」コース
 上記3コースを受講済みか、同程度の知識のある方
学習内容 製薬業の臨床開発における統計解析業務に焦点を当て、実業務に対応した形式でSASの利用方法を紹介します。臨床開発業務のデータ解析業務にスポットを当て、データの読み込みから一覧表および解析表作成までのさまざまなプログラミング技法についてご紹介します。
価格/日程 2日間コース 60,000円(チケット捺印数3)
 東京会場:11月29日(木)~30日(金)

なお、上記以外にも多数のトレーニングコースがございます。コース情報については、弊社ホームページ(<http://www.sas.com/japan/training/>)をご参照いただくか、下記<トレーニング担当>までお問合せください。

トレーニング担当

T E L 03-3533-3835
 F A X 03-3533-3781
 E-mail training@jpn.sas.com

Latest Releases

最新リリース情報

PCプラットフォーム

Windows版	SASシステム リリース6.12	TS060
Windows版	SASシステム リリース8.2	TS2M0
OS/2版	SASシステム リリース6.12	TS020
Macintosh版	SASシステム リリース6.12	TS040

ミニコンピュータプラットフォーム

OpenVMS AXP版	SASシステム リリース6.12	TS020
OpenVMS VAX版	SASシステム リリース6.08	TS407

UNIXプラットフォーム

MIPS ABI版	SASシステム リリース6.11	TS040
Digital UNIX版	SASシステム リリース6.12	TS040
ABI+版	SASシステム リリース6.11	TS040
SunOS/Solaris版	SASシステム リリース6.12	TS060
HP-UX版	SASシステム リリース6.12	TS060
AIX版	SASシステム リリース6.12	TS060

メインフレームプラットフォーム

MVS版	SASシステム リリース6.09E	TS470
MSP版	SASシステム リリース6.09E	TS470
VOS3版	SASシステム リリース6.09E	TS470
CMS版	SASシステム リリース6.08	TS410

Programmers Workshop

SASプログラマーズ・ワークショップ開催のお知らせ

今回のワークショップでは、SAS/STATソフトウェアを中心に、SASシステム リリース8.2における統計系プロシジャの新機能をご紹介します。リリース6.12からの主な拡張点や、注意すべき変更点、互換性の問題など、またEnterprise Miner ソフトウェア リリース4.1 における新機能についても併せてご紹介します。

日時: 11月22日(木)14:00 ~ 17:00

会場: 株式会社SASインスティテュートジャパン 東京会場

住所: 東京都中央区勝どき1-13-1 イヌイビル・カチドキ8F
プレゼンテーションルーム

定員: 50名

日時: 11月30日(金)14:00 ~ 17:00

会場: 株式会社SASインスティテュートジャパン 大阪会場

住所: 大阪市北区堂島浜1-4-16 アクア堂島西館12F
プレゼンテーションルーム

定員: 50名

両日とも参加は無料となりますので、ぜひご参加ください。

なお、定員に限りがございますので、お早めにお申し込みください。参加ご希望の方は、「SASプログラマーズ・ワークショップ参加申込用紙」に必要事項をご記入の上、FAX、E-mailにてお申し込みください。SASプログラマーズ・ワークショップのお申し込み用紙は、下記のURLより入手できます。

<http://www.sas.com/japan/service/technical/pws.html>

その他、ご不明点、ご質問がございましたら、以下までご連絡ください。

SASプログラマーズ・ワークショップ担当

T E L 03-3533-3835

F A X 03-3533-3781

E-mail pws@jpn.sas.com

SAS Technical News Autumn 2001

発行
株式会社SASインスティテュートジャパン

テクニカルニュースに関するお問い合わせ先

テクニカルサポートグループ
TEL: 03-3533-3877
FAX: 03-3533-3781
E-mail: technews@jpn.sas.com



株式会社SASインスティテュートジャパン

www.sas.com/japan/

東京本社
〒104-0054
東京都中央区勝どき1-13-1
イヌイビル・カチドキ8F
Tel 03 (3533) 6921
Fax 03 (3533) 6927

大阪支店
〒530-0004
大阪市北区堂島浜1-4-16
アクア堂島西館12F
Tel 06 (6345) 5700
Fax 06 (6345) 5655

九州営業所
〒812-0011
福岡市博多区博多駅前3-7-35
博多ハイテックビル6F
Tel 092 (415) 6101
Fax 092 (415) 6131