The Road to Java

Lab 1

Fall Common 1998

Tom Gall and AS/400 Toolbox for Java Team

IBM Corporation

Disclaimer

This package is available for use AS IS. There is no support or service to the documentation and the code shipped with the package. IBM reserves all the rights to the lab material. This self-study material is provided for personal use. Reproduction of the material for commercial use is prohibited unless written agreement is provided by IBM.

Lab: The Road To Java

Introduction

Java is an exciting new programming language that is quickly being accepted in the computing industry. Java offers the programmer the flexibility to create full fledged applications or applets that can run inside of a World Wide Web browser.

Goal

In this lab, you will work through a series of exercises that provide you with an opportunity to create java applications that run on the AS/400 as well as Windows 95. Last you will explore the differences between a Java applet and Java application.

Tools

The tools that you will use for the exercises in this lab for the most part are quite primitive. The most very basic editor and command line compiler are utilized to create Java source code and compile your Java programs. On the AS/400 we will make use of SEU and on Windows 95 the old MS-DOS editor **edit** will be used.

Setup

During this lab you will need a userid and password for the AS/400 and you will need to know the name of the AS/400 system. The userid and password will be **JAVAxx** where **xx** is the number 01 to 99 (two digits number). This information will be given to you by your instructor. Please fill in this information below so that you have it for reference during the lab.

Please note that if a lab uses an AS/400 library, the name of that library will be the same as your userid.

My AS/400 **userid** is_____
My AS/400 **password** is_____
My AS/400 **tcp/ip address** is _____
My PC **tcp/ip address** is _____
The AS/400 **directory** is _____
The AS/400 **library** is _____
The PC **directory** is _____

The name of my AS/400 **system_name** is _____. Please write down the AS/400 system name exactly as shown by your instructor because this name is case sensitive.

Exercise 1: Creating the Hello World Application on Windows 95

What This Exercise Is About

In this exercise the student will create a simple "Hello World" Java application on Windows 95.

What You Should Be Able To Do

At the end of this exercise, you should be able to:
- Compile Java source code
- Run a Java program from the command line
- Create a Java class
- Know how to print text to the screen in a Java application

Introduction

Let's create your very first Java application. It's just a simple program that will print out the words "Hello World".

Exercise Instructions

Open An Editor

- Open an **MS-DOS** prompt. (Press the **Start** button, follow the menu to **Programs** and choose **MS-DOS** prompt)
- First we need to create a directory in which to place your work. The directory will be name supplied by your instructor.

    ```
    mkdir <directory>
    ```
- Change your current directory to the directory that you created in the previous step.
    ```
    cd <directory>
    ```
- Open the MS-DOS editor to create a new piece of Java source named HelloWorld.java. This is done by using the following command.

    edit HelloWorld.java

Enter And Compile The Source

- Type in the follow following source code. Please note the punctuation as well as the capitalization. Both are very important elements to Java's syntax.

    ```
    public class HelloWorld extends Object
    {
    public static void
    main(String args[])
        {
        System.out.println("Hello World!");
        }
    }
    ```

- Under the **File** pull down menu, choose **Save**. Then under the **File** pull down menu, choose **Exit**. You should be back at the MS-DOS command line.
- It's now time to compile your source code into Java bytecodes using the javac compiler. Issue the follow command at the MS-DOS prompt.
  `javac HelloWorld.java`
- If the source code compiles successfully without any errors, you will be presented again with another MS-DOS prompt. If there are errors in your source code, they will be displayed.
- If you have any errors in your source code re-edit the HelloWorld.java file to fix the errors. Mostly likely if you have an error, something was probably spelled incorrectly, capitalized incorrectly or punctuation was quite right.
- After you are able to compile with no errors, you will notice a new file that exists in your directory, called HelloWorld.class. Issue the '**dir**' command to see it.

Run The HelloWorld Application

- Once you are able to compile the HelloWorld.java source code without any errors or warnings, it's time to run the Application.
- At the MS-DOS prompt issue.
  `java HelloWorld`
- You should now see the message
  `Hello World!`

Exercise 2: Creating the Hello World Application On the AS/400

What This Exercise Is About

In this exercise the student will create a simple "Hello World" Java application on the AS/400.

What You Should Be Able To Do

At the end of this exercise, you should be able to:
- Compile Java source code
- Run a Java Application from the command line
- Create a Java class
- Know how to print text to the screen in a Java Application

Introduction

Let's create your second Java application. Again it's just a simple program that will print out the words "Hello World". This exercise takes the approach that you only have access to an AS/400. As an alternative to SEU you could use an editor on your PC and use client access or FTP to transfer your source into an IFS file on the AS/400. For more information about how to do that consult the URL http://www.softmall.ibm.com/as400/java/devkit/rzaha.html.

Exercise Instructions

Log On To The System

- Log onto the AS/400 using the supplied user id and password.
- Next you need to create a directory that is located in IFS to place your java source code as well as your compiled java programs.
- **mkdir ('directory')**
- Last we need to create an AS/400 library in which to place the source physical file.
- **crtlib <AS/400 library>**
- Add the library to your library list
- **addlible <AS/400 library>**

Create The Java Source Code

First you will use the AS/400 SEU tool to create a your java source file. However SEU stores the java source code inside of a source member instead of IFS. When you use the java compiler to compile your java source, it needs to be in IFS. You'll make use of the **CPYTOSTMF** command which copies the java source from your source physical file into an IFS file.
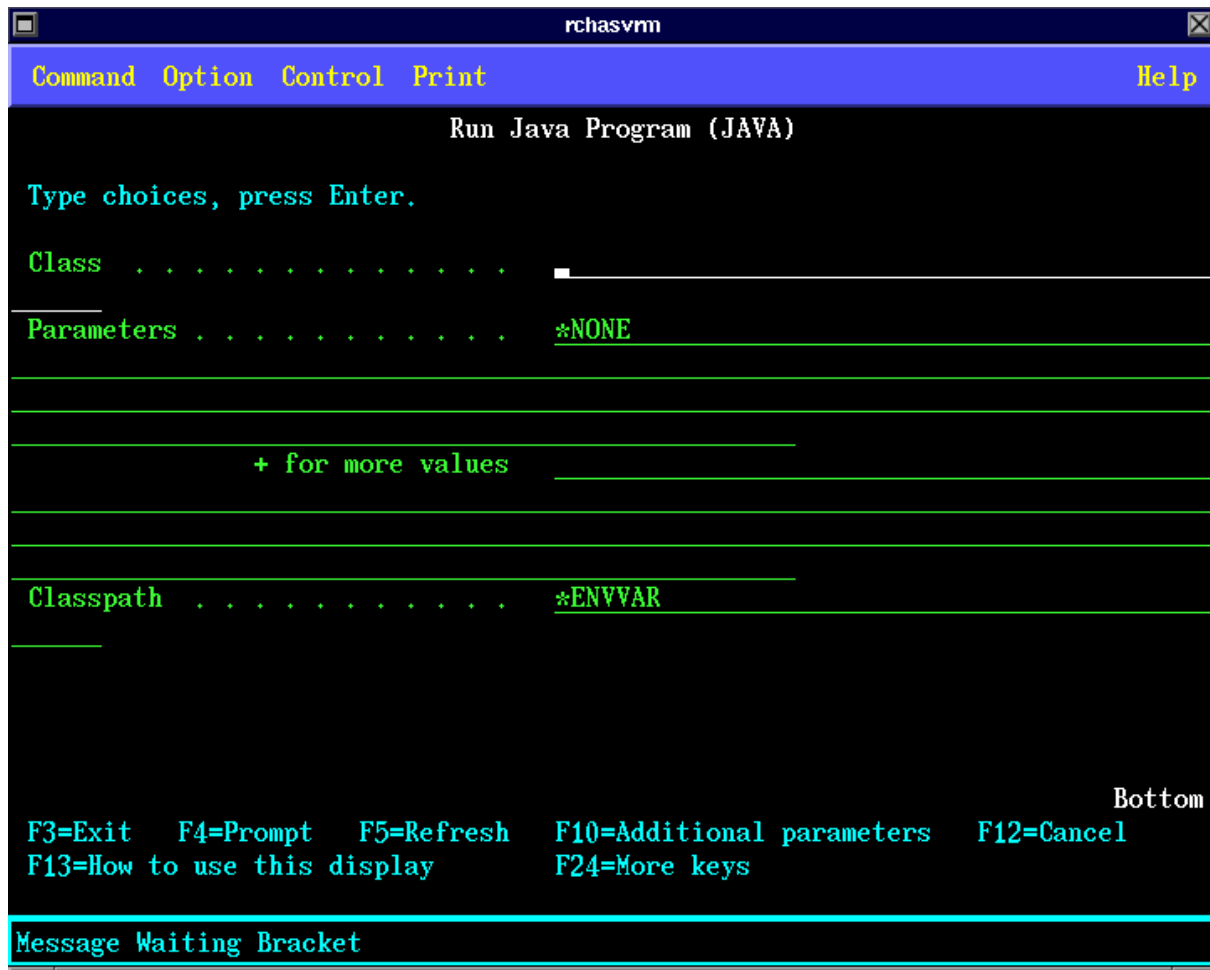
- Issue **CRTSRCPF** and press F4 to prompt on the command.

- For file, fill in jsource, for library fill in the library you will given, and last enter 'java source' in the Text 'description field', lastly press enter.
- Now issue **STRPDM**
- Take option 3 to Work with members.
- For the File field enter jsource which is the name of the physical file you just created a moment ago, and for the library, enter the library you were given.
- Now we need to create the java source code with SEU. Press the F6 key to Start Source Entry Utility.
- Enter **HELLO** as the name of the Source member, enter **txt** as the source type, and lastly put 'Hello World Example' into the Text 'description' field. Press Enter.
- Type in the follow following source code. Please note the punctuation as well as the capitalization. Both are very important elements to Java's syntax.

```
public class HelloAS400 extends Object
{
public static void
main(String args[])
        {
        System.out.println("Hello World from AS/400!");
        }
}
```

Copy And Compile The Source

- Once you have entered the source into SEU, it's time to save the source code. Hit the F3 key. Make sure 'N' is set on **Return to editing**. Press Enter.
- Now it is time to copy the source file from your library into IFS. This is accomplished with the **CPYTOSTMF** command. Type in **CPYTOSTMF** and hit F4 to prompt on it.
- Enter the following data for the following fields,

```
From database file member: /qsys.lib/<library>.lib/jsource.file/Hello.mbr
To stream file:  /directory/HelloAS400.java

Stream file code page: 819
```
- Now hit enter twice and the stream file HelloAS400.java will be created in IFS. You are now ready to compile your java program on the AS/400.
- Exit out from Work with Members Using PDM with F3.
- Exit out from the AS/400 Program Development Manager (PDM) with F3
- You need to change the current directory to where you just stored the HelloAS400.java file. Issue

```
cd ('/<directory')
```
- Now we will run the java compiler, type **RUNJVA** then hit the F4 key to prompt on the command.

```
┌─────────────────────────────────────────────────────────────────────────┐
│ ■                              rchasvrm                                ⊠ │
├─────────────────────────────────────────────────────────────────────────┤
│  Command  Option  Control  Print                                   Help  │
│                        Run Java Program (JAVA)                           │
│                                                                          │
│   Type choices, press Enter.                                             │
│                                                                          │
│   Class . . . . . . . . . . . . .   ▪_____    │
│                                                                          │
│   Parameters . . . . . . . . . . .  *NONE                                │
│                                     _____ │
│                                     _____ │
│                                     _____ │
│            + for more values        _____ │
│                                     _____ │
│                                     _____ │
│                                                                          │
│   Classpath  . . . . . . . . . . .  *ENVVAR                              │
│   _____                                                                 │
│                                                                          │
│                                                                          │
│                                                             Bottom       │
│  F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters  F12=Cancel│
│  F13=How to use this display       F24=More keys                         │
├─────────────────────────────────────────────────────────────────────────┤
│ Message Waiting Bracket                                                   │
└─────────────────────────────────────────────────────────────────────────┘
```

- To run the java compiler, fill in the following fields like so:
  ```
  Class:  sun.tools.javac.Main
  Parameters HelloAS400.java
  ```
- Now hit the enter key to run the compiler. If there are any errors in your source, they will be printed to the screen. You'll need to go back and reedit your source file with SEU to fix the errors. After you have saved your changes, be sure to recopy the source file into IFS with the **CPYTOSTMF** command.
- If you are successful and your source compiles with no errors a file called HelloAS400.class will be created inside of your directory in IFS.
- Let's take a moment to view the file that you created.

          `wrklnk`
- You will see HelloAS400.java which is the source file that you compiled with javac and you will see HelloAS400.class which is the compile java program. In the next phase of this exercise you will run this program.
- Press F3 to exit from **WRKLNK**

Run The HelloAS400 Application

- First you need to create an environment variable called CLASSPATH. This environment variable is a list of IFS directories that java searches for class files to run. Issue **ADDENVVAR ENVVAR(CLASSPATH) VALUE('/<directory>')**
- Now let's run the HelloAS400 java application. Issue
        `RUNJVA HelloAS400`

- You should now see the message
        `Hello World from AS/400!`

Exercise 3: AS/400 Java Programs

What This Exercise Is About

In this exercise you will examine the **CRTJVAPGM**, **DSPJVAPGM** and **DLTJVAPGM** commands on the AS/400.

What You Should Be Able To Do

At the end of this exercise, you should be able to:
- Use **CRTJVAPGM** to make Java Program go faster
- Use **DSPJVAPGM** to analyze how a java program was created
- Use **DLTJVAPGM** to delete a Java Program

Introduction

In order to make Java run faster on the AS/400 a Java Program is associated with each class file, ZIP file or JAR file. The Java Program is a preverified compiled version of a class file. The optimization level you specify with **CRTJVAPGM** determines how much of a speed increase you will realize in your Java Application. The Java Program isn't visible in IFS where java class/jar files are stored. Tools like **CRTJVAPGM**, **DSPJVAPGM** and **DLTJVAPGM** allow you to interact with the Java Program.

A Java Program may be created one of two ways. Either by explicitly using the **CRTJVAPGM** command on a class/jar file, or on first use of the jar/class file when it is used by a java application. In this exercise we will look at some of the options of how Java Programs are created.

DSPJVAPGM will display the attributes to the named class/jar file in IFS.

DLTJVAPGM will destroy the java program that is associated with a class/jar file in IFS.

Exercise Instructions

CRTJVAPGM, DSPJVAPGM

- Log onto the AS/400.
- Enter
  ```
  ADDENVVAR ENVVAR(CLASSPATH) VALUE('/<directory>')
  ```

- Execute
  ```
  cd('/<directory')
  ```

- Now you will use the java application HelloAS400 that you created in exercise 2. As was mentioned in the introduction java programs are created when a class file is used for the first time. Since you have already executed this java application, there is already a java program for the class file HelloAS400.class. Let's take a look at it. Issue

  ```
  DSPJVAPGM CLSF(HelloAS400.class)
  ```

- Notice the various fields, most importantly the optimization field. Optimization level 10 is the default when a java program is created. Let's make our java program faster.
- Issue **CRTJVAPGM** and hit F4 to prompt on it.
- Notice that optimization level 40 is the highest level you can go. Also of note is that optimization level 10 or *INTERPRET is best when you are debugging your application. Optimization level 40 is best for speed.
- Also notice that you may enable performance collection hooks. When you place these hooks into your java programs, you may use the Performance Explorer to collect performance information on your java applications.
- Enter 40 for the optimization level and press enter.
- Now return your HelloAS400 application by entering

```
java HelloAS400
```

- The first thing you probably will notice is that your java application isn't running noticeably faster. Here's why. The class file HelloAS400.class is only one of many class files are that used when your java application is run! The other class files are located in two jar files named java.zip and sun.zip in the IFS directory /QIBM/ProdData/Java400/lib. Let's see what optimization level they are at. Issue

```
DSPJVAPGM CLSF('/QIBM/ProdData/Java400/lib/java.zip')
```

DLTJVAPGM

- When you delete a class file from IFS, if there was a java program associated with the class file, it will be deleted as well. But what if you want to save the class file and only get ride of the java program associated with it. To do this, there is the **DLTJVAPGM** command. Let's try it out now on your HelloAS400 application.
- Issue

```
DLTJVAPGM CLSF('HelloAS400.class')
```

- Your java program should now be deleted. Let's use **DPSJVAPGM** to double check. Issue

```
DSPJVAPGM CLSF('HelloAS400.class')
```

Exercise 4: Create Your Own Home Page

What This Exercise Is About

Before we are able to create any Java Applets, you will need a home page on which to place them. In this lab you will create an HTML document for the Java Applets that you will create and reuse in the next exercises.

What You Should Be Able To Do

At the end of this exercise, you should be able to:
- Create a web page using Netscape

Introduction

World Wide Web pages are written using a language called HyperText Markup Language better know as HTML. Learning HTML is comparable to learning another programming language. Thankfully HTML editors have entered in the market in recent years. These editors allow you to create your very own home page without having to learn any HTML what-so-ever. Composing a home page becomes as easy as writing a letter in your favorite word processor.

Exercise Instructions

Start Netscape and Composer

- Double Click on the **Netscape Communicator** icon that is located on your desktop
- From the **Communicator** pull down menu in Netscape choose the **Page Composer** menu item.

You are now ready to start creating your own home page. Pictured above is the Netscape Composer.
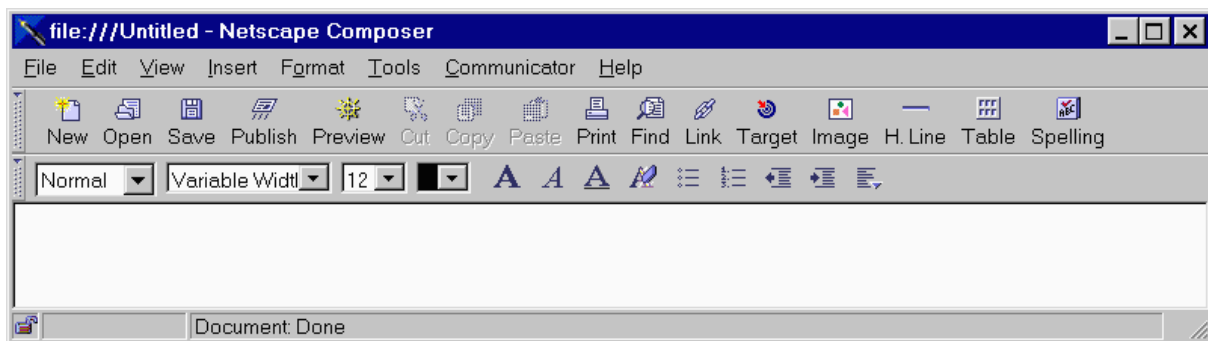
Adding Text To Your Web Page

- Let's put a title on this web page. We'll want to put this in a bigger head line font. Select in the left most drop down list box **Heading 1**. (Hint: It's located right under the icon labeled "**New**")
- Type in the editor the following text "**My Home Page**" and hit **return**.

Adding A Link In Your Web Page To Another Web Page

• Now we will make a link to the AS/400 home page. Press the **Link** button from the toolbar.
• Fill in the Dialog as pictured and click on "**OK**"
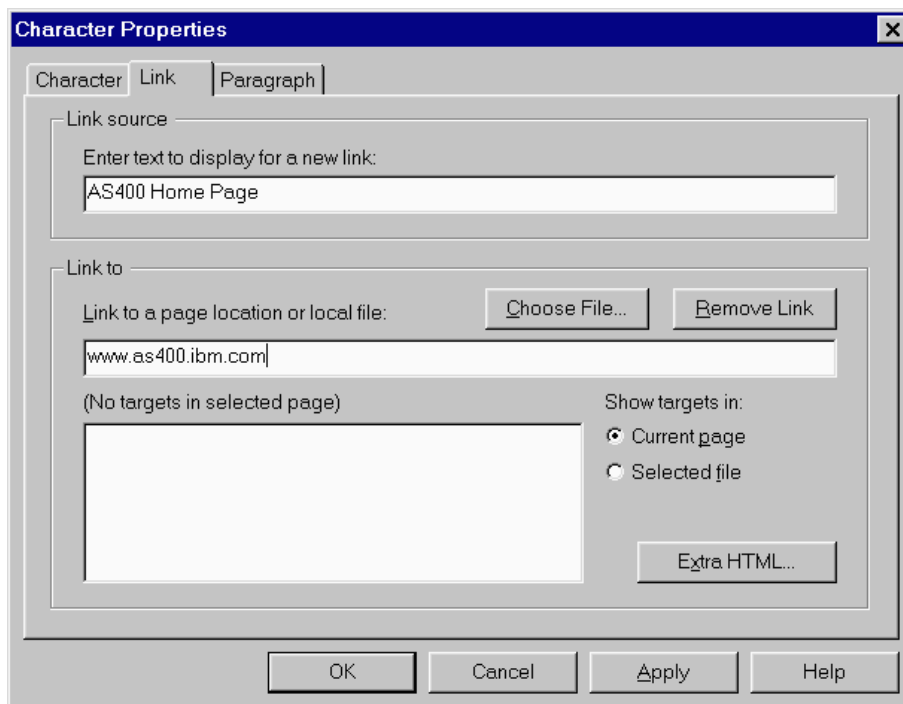
Now your web page should look something like this:
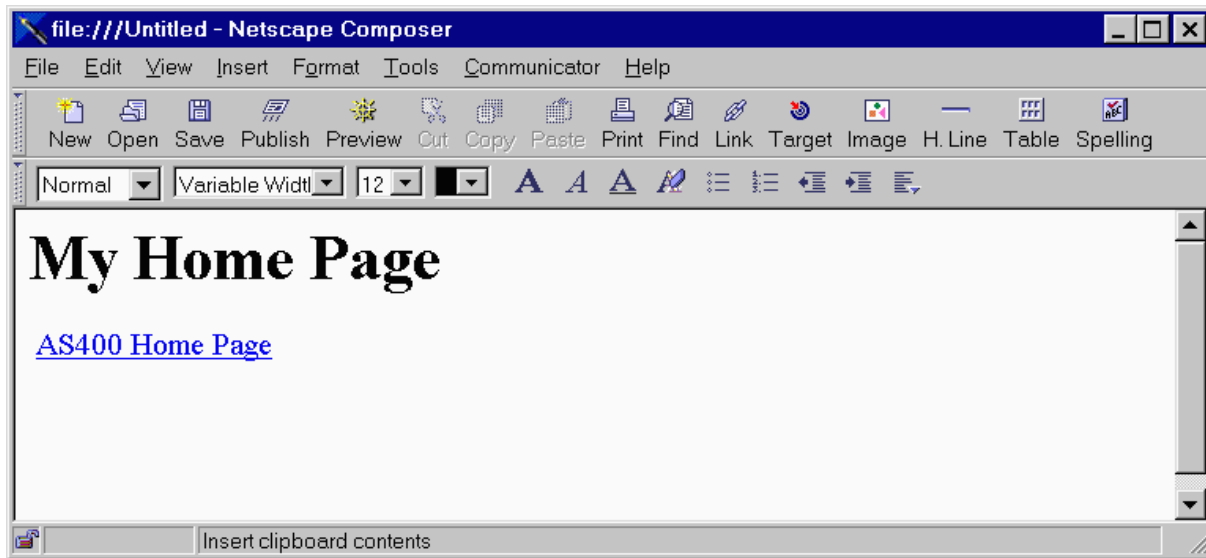
Saving Your Web Page
Last you need to save your home page, as we will need it for Exercise 3 and 4 of this lab.

- Click on **Save** from the Toolbar. Make sure to save your Home page in the same directory as your you have placed your HelloWorld java code from Exercise 1.
- Name your HTML file **HomePage** and Click on the **OK** button.
- Next you will be presented with a dialog to give your page a title. Click on **OK** again.

Exercise 5: Creating The Hello World Applet

What This Exercise Is About

As you will recall Applets are Java programs that are dynamically downloaded over the World Wide Web and run in your browser. In this exercise we will create a very simple Hello World Applet and place it on the home page you created in Exercise 2.

What You Should Be Able To Do

At the end of this exercise, you should be able to:

- Create an Applet
- Compile an Applet with javac
- Connect an Applet inside of an existing HTML page
- Test a Web Page with an Applet

Introduction

The source code to create an Applet is very similar to the HelloWorld application that you created earlier. When you create a Java Applet, you must subclass the java.applet.Applet class. This means you are going to use the java.applet.Applet class as a reusable part and extend it to do what you want. In this case you will make it print out the word "Hello World" in the browser.

Create The HelloWorldApplet Source File

- Open an **MS-DOS** prompt. (Press the **Start** button, follow the menu to **Programs** and choose **MS-DOS** prompt)
- Change your current directory to the directory that is given for this lab. Your lab instructor should have given this to you.
- Open the MS-DOS editor to create a new piece of java source named HelloWorldApplet.java. This is done by using the following command.

```
edit HelloWorldApplet.java
```

Enter And Compile The Source

- Type in the follow following source code. Please note the punctuation as well as the capitalization.

```
import java.applet.*;
import java.awt.Graphics;

public class HelloWorldApplet extends java.applet.Applet
   {
   public void
   paint(Graphics g)
        {
        g.drawString("Hello World!", 5 ,25);
        }
```
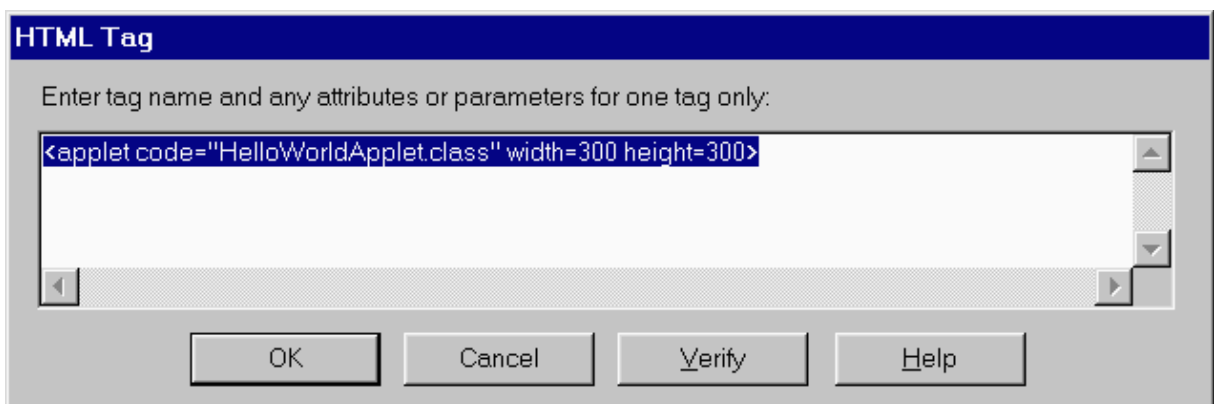

                    }
- Under the File pull down menu, choose Save. Then under the File pull down menu, choose quit. You should be back at the MS-DOS command line.
- It's now time to compile your source code into Java bytecodes using the javac compiler. Issue the follow command at the MS-DOS prompt.
- `javac HelloWorldApplet.java`
- If the source code compiles successfully without any errors, you will be presented again with another MS-DOS prompt. If there are errors in your source code, they will be displayed.
- If you have any errors in your source code re-edit the HelloWorldApplet.java file to fix the errors. Mostly likely if you have an error, something was probably spelled incorrectly, capitalized incorrectly or punctuation was not quite right.
- After you are able to compile with no errors, you will notice a new file that exists in your directory, called HelloWorldApplet.class. Issue the '**dir**' command to see it.

Insert The HelloWorldApplet Into Your Home Page

- Start Netscape
- From the **Communicator** pull down menu click on the **Page Composer** menu item
- In the Netscape Composer window, from the **File** pull down menu click on the **Open Page...** menu item. Click on the **Choose File...** button. Select the home page you created in exercise 3. It should be saved in the directory the lab instructor gave you with the name **HomePage**. Click on the **Open** button once you have selected the file. Last click on the **Open** button in the Open Page dialog. Your home page should appear in the editor.
- Pick a location you would like the Applet to appear on your home page. (After the link to the AS/400 home page would be a good spot). Click once to place the cursor in the location you have chosen.

HTML Tag

Enter tag name and any attributes or parameters for one tag only:

```
<applet code="HelloWorldApplet.class" width=300 height=300>
```

| OK | Cancel | Verify | Help |

- From the **Insert** pull down menu click on the **HTML Tag...** menu item.
- Fill in the dialog as pictured. Notice the value HelloWorldApplet.class which is assigned to code. This is the same file name which was the output of the javac compiler)

- Click on **OK**
- Again from the **Insert** pull down menu click on the **HTML Tag...** menu item.

HTML Tag

Enter tag name and any attributes or parameters for one tag only:

```
</applet>
```

OK   Cancel   Verify   Help
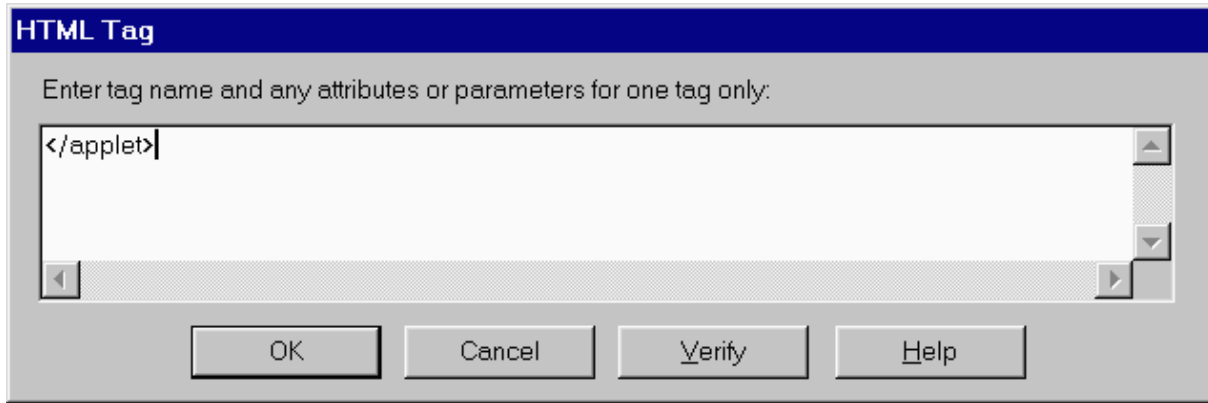
- Fill in the dialog as pictured.

- Click on the **OK** button.

Your HTML document should now look like this picture.
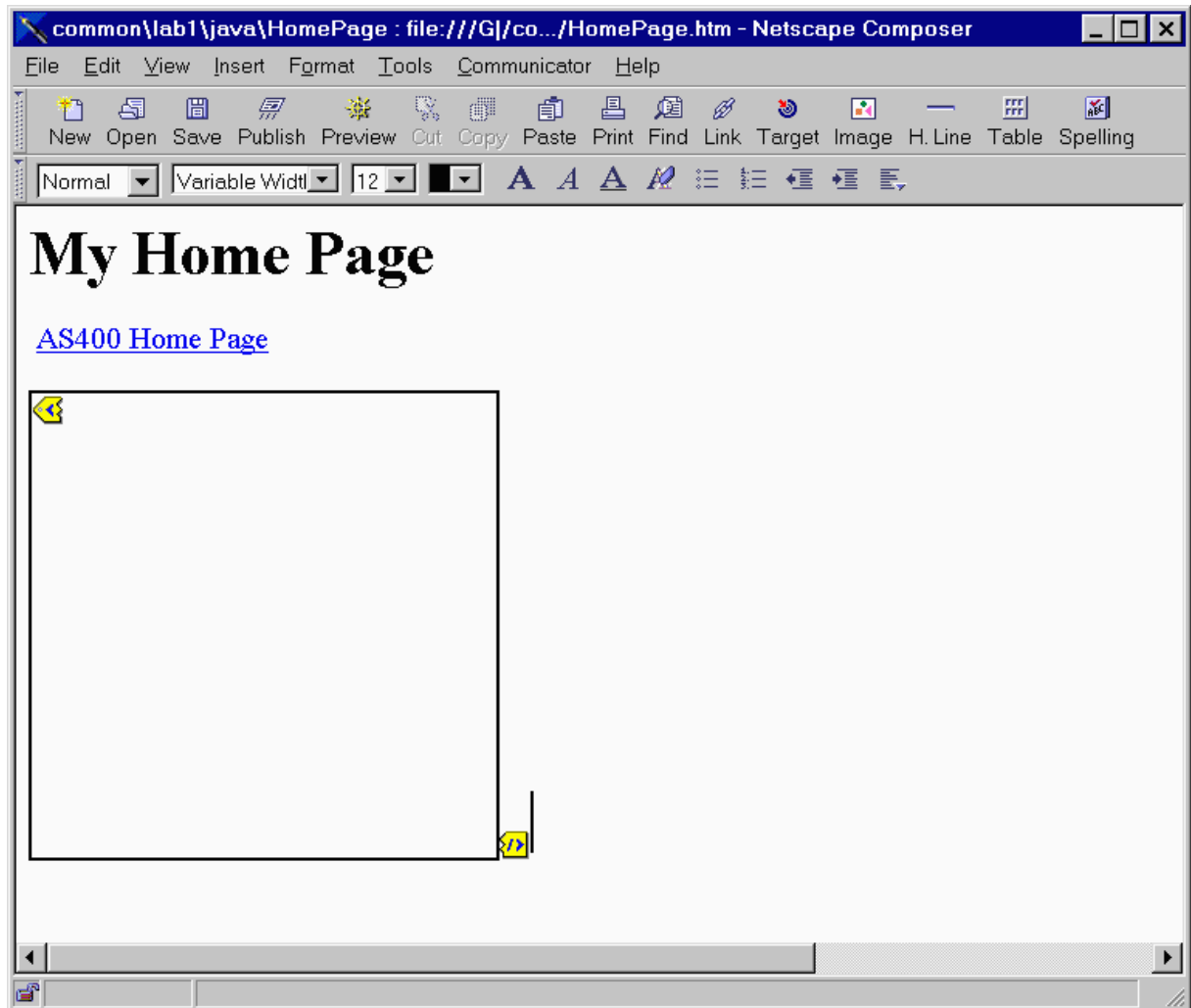
Test The HelloWorldApplet

- From the **File** pull down menu click on the **Browse Page** menu item. You will be prompted if you would like to save your changes. Click on **yes**.
- A new Netscape window should appear with your home page in it. After a moment the HelloWorldApplet should start up and display it's friendly message.

Bonus Lab: Running a Java Application with a GUI on the AS/400 but use it on your PC

What This Exercise Is About

In V4R2 Remote AWT is included with Java. Remote AWT allows you to run a Java Application that has a Graphical User Interface on the AS/400 but the window will appear on your PC. In this exercise the student will create a simple "Hello World" Java Application that has a Graphical User Interface, run it on the AS/400 but use the application on your PC.

What You Should Be Able To Do

At the end of this exercise, you should be able to:
- Run a Java Application with a GUI from the command line
- Know the steps to setup your PC so you can display AS/400 Java Applications.

Introduction

Let's create your second Java application. Again it's just a simple program that will print out the words "Hello World". This exercise takes the approach that you only have access to an AS/400. As an alternative to SEU you could use an editor on your PC and use client access or FTP to transfer your source into an IFS file on the AS/400. For more information about how to do that consult the URL http://www.softmall.ibm.com/as400/java/devkit/rzaha.html.

Exercise Instructions

Log On To The System

(If you did not already do this in the previous lab exercises)

- Log onto the AS/400 using the supplied user id and password.
- Next you need to create a directory that is located in IFS to place your java source code as well as your compiled java programs.
- mkdir ('directory')
- Last we need to create an AS/400 library in which to place the source physical file.
- crtlib <library>
- Add the library to your library list
- addlible <library>

Create The Java Source Code

First you will use the AS/400 SEU tool to create a your java source file. However SEU stores the java source code inside of a source member instead of IFS. When you use the java compiler to compile your java source, it needs to be in IFS. You'll make use of the CPYTOSTMF command which copies the java source from your source physical file into an IFS file.

- Issue **CRTSRCPF** and press F4 to prompt on the command.

- For file, fill in jsource, for library fill in the library you will given, and last enter 'java source' in the Text 'description field', lastly press enter.
- Now issue **STRPDM**
- Take option 3 to Work with members.
- For the File field enter jsource which is the name of the physical file you just created a moment ago, and for the library, enter the library you were given.
- Now we need to create the java source code with SEU. Press the F6 key to Start Source Entry Utility.
- Enter HelloGUI as the name of the Source member, enter **txt** as the source type, and lastly put 'Hello World GUI Example' into the Text 'description' field. Press Enter.
- Type in the follow following source code. Please note the punctuation as well as the capitalization. Both are very important elements to Java's syntax.

```
import java.awt.*;
import java.awt.event.*;

public class HelloGUIAS400 extends Frame
{
public HelloGUIAS400(String windowName)
      {
      super(windowName);  // sets the name of the window

      addWindowListener(
         new WindowAdapter()  // creates an anonymous class
            {
            // handles closing events, when the user requests to
            // close the window, the application exits
            public void
            windowClosing(WindowEvent e)
               {
               System.exit(0);
               }
            }
        );
      setSize(300,300);   // sets size of the window
      show();             // show the window on the screen
      }
public static void
main(String args[])
      {
      HelloGUIAS400 win=new HelloGUIAS400("Hello World Window");
      }
}
```

Copy And Compile The Source

- Once you have entered the source into SEU, it's time to save the source code. Hit the F3 key. Make sure 'N' is set on **Return to editing**. Press Enter.
- Now it is time to copy the source file from your library into IFS. This is accomplished with the **CPYTOSTMF** command. Type in **CPYTOSTMF** and hit F4 to prompt on it.
- Enter the following data for the following fields,

```
From database file member:
                /qsys.lib/<library>.lib/jsource.file/HelloGUI.mbr
To stream file:  /directory/HelloGUIAS400.java
Stream file code page: 819
```

- Now hit enter twice and the stream file HelloGUIAS400.java will be created in IFS. You are now ready to compile your java program on the AS/400.
- Exit out from Work with Members Using PDM with F3.
- Exit out from the AS/400 Program Development Manager (PDM) with F3
- You need to change the current directory to where you just stored the HelloGUIAS400.java file. Issue

```
cd ('/<directory')
```

- Now we will run the java compiler, type **RUNJVA** then hit the F4 key to prompt on the command.

```
┌─────────────────────────────────────────────────────────────────────┐
│ ■                          rchasvrm                              ⊠  │
├─────────────────────────────────────────────────────────────────────┤
│  Command  Option  Control  Print                              Help  │
│                     Run Java Program (JAVA)                         │
│                                                                     │
│   Type choices, press Enter.                                        │
│                                                                     │
│   Class  . . . . . . . . . . .    ■_____        │
│                                                                     │
│   Parameters . . . . . . . . . .  *NONE                             │
│                                                                     │
│                                                                     │
│                                                                     │
│              + for more values    _____             │
│                                                                     │
│                                                                     │
│   Classpath  . . . . . . . . . .  *ENVVAR                           │
│                                                                     │
│                                                                     │
│                                                                     │
│                                                                     │
│                                                               Bottom│
│   F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel │
│   F13=How to use this display      F24=More keys                    │
│                                                                     │
│  Message Waiting Bracket                                            │
└─────────────────────────────────────────────────────────────────────┘
```
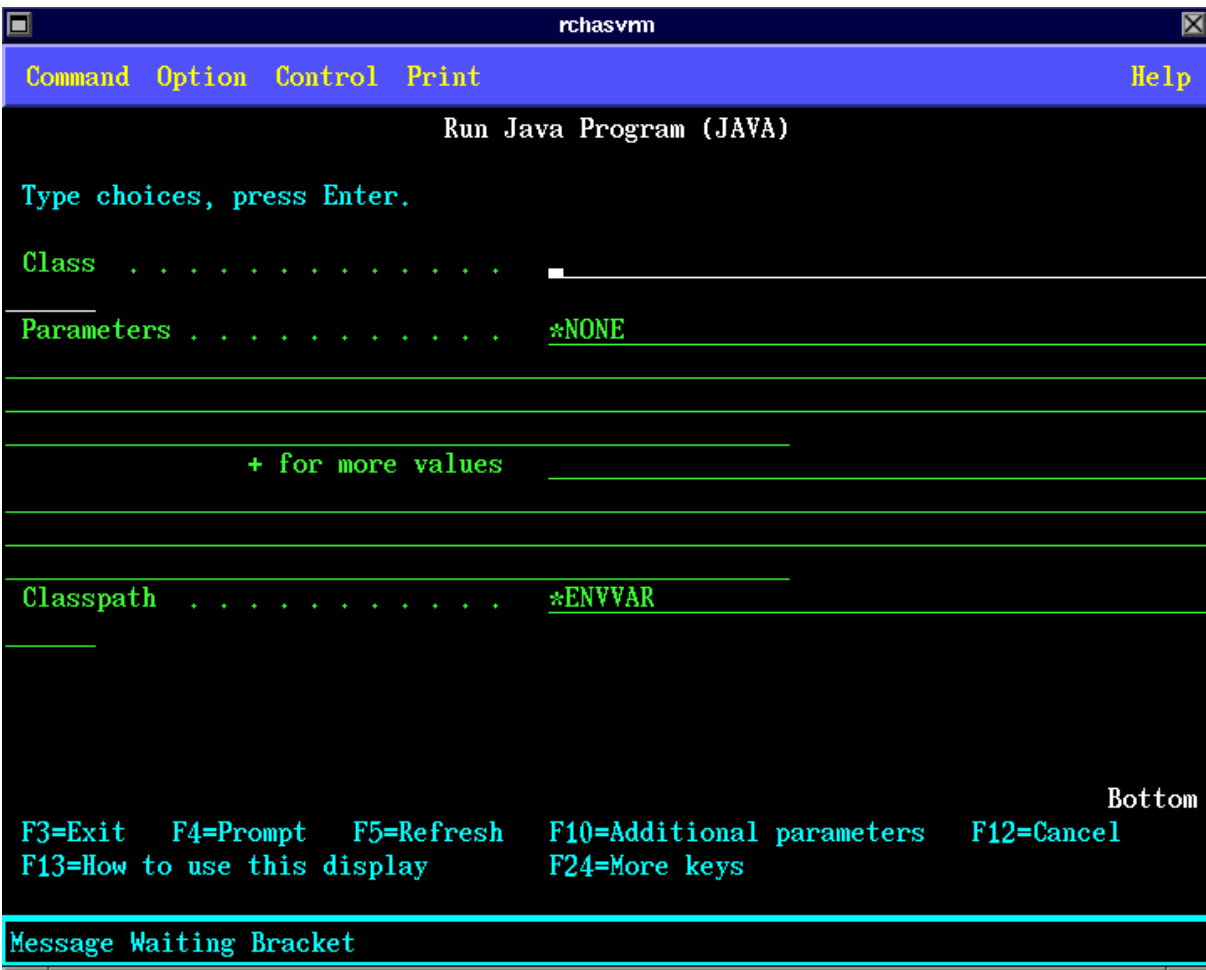
- To run the java compiler, fill in the following fields like so:
  Class: sun.tools.javac.Main
  Parameters HelloGUIAS400.java
- Now hit the enter key to run the compiler. If there are any errors in your source, they will be printed to the screen. You'll need to go back and reedit your source file with SEU to fix the errors. After you have saved your changes, be sure to recopy the source file into IFS with the **CPYTOSTMF** command.
- If you are successful and your source compiles with no errors a file called HelloGUIAS400.class will be created inside of your directory in IFS.
- Let's take a moment to view the file that you created.

  wrklnk
- You will see HelloGUIAS400.java which is the source file that you compiled with javac and you will see HelloGUIAS400.class which is the compile java program. In the next phase of this exercise you will run this program.
- Press F3 to exit from **WRKLNK**

Run The HelloGUIAS400 Application

In order to run the Java Application yet have the user interface of the application display on your PC, you will need to first off run a server on your PC that will allow the Java Application to display itself on your PC. Second you will have to actually start the application on the AS/400.

- On the PC issue the following command:
  ```
  java java.awt.Awt400ServerImpl <PC tcp/ip address>:32869
  ```
- You are ready to run the Java Application HelloGUIAS400 on the AS/400. Issue **RUNJVA** and prompt on the command with PF4.
- Fill in the following fields like so:
  Class: HelloGUIAS400
  Classpath: /<directory>
- Page down to the properties portion of the command and enter:
  awt.toolkit with the property value of **java.awt.CToolkit**
- Enter + for values and add the following property
  RmtAwtServer with the property value of **<PC tcp/ip address>:32869**
  Press enter and the window should appear on your PC in moments.
  You might get an error message on the AS/400 panel about Find native method library exception. Ignore it.
- After the window has appeared click on the x button in the upper right corner of the window to exit the application.

Congratulations! You've finished the bonus lab!