

## Compiling, Packaging, and Running the Application

Now that you have examined the sample application in detail, it's time to compile, package, and run the application. First, though, download and install the Struts and Tomcat software if you have not already done so. Tomcat is a free servlet engine available for download from the Internet and will be used in the examples in this book for running Struts applications. Tomcat is also the reference implementation for the JSP and servlet specifications from Sun. Of course, you don't have to use Tomcat to run the examples, but it is the only method described by this book. So, to follow along, it is strongly suggested that you use Tomcat.

Each of the following sections is dedicated to a step of the process. First, you will set up the Struts and Tomcat software. Next, you'll compile the application. Then, you'll package the application in a standard Web Archive file. Finally, you'll see how to deploy and run the application with Tomcat.

### Downloading and Installing Struts and Tomcat

As mentioned, both Struts and Tomcat are freely available for download from the Internet. Following are the Web sites for each:

- **Struts** <http://jakarta.apache.org/struts/>
- **Tomcat** <http://jakarta.apache.org/tomcat/>

After you have downloaded the Struts and Tomcat software distributions, you will need to choose a directory to install them to. After selecting a directory, extract the files of each distribution to that directory. For example, if you choose to install the distributions in a directory called **c:\java**, then the Struts files would be located at **c:\java\jakarta-struts-1.1** and Tomcat would be installed at **c:\java\jakarta- tomcat-4.1.27**.

### Compiling the Application

The Mini HR application is comprised of several files; however, only the Java source code files need to be compiled before you package and run the application. Because the Java source code files use the servlet and Struts APIs, you need to add these libraries to your Java classpath. You could do this by updating your CLASSPATH environment variable. Alternatively, you can just specify the classpath when you compile the Mini HR application.

In addition to the files that you created and reviewed earlier in this chapter, you also need to copy the following files to the **c:\java\MiniHR\WEB-INF\lib** directory. These **.jar** files contain the Struts and associated library class files that are necessary for the Mini HR application to run once it is packaged as a **.war** file.

```
c:\java\jakarta-struts-1.1\lib\commons-beanutils.jar c:\java\jakarta-struts-1.1\lib\commons-  
collections.jar c:\java\jakarta-struts-1.1\lib\commons-lang.jar c:\java\jakarta-struts-  
1.1\lib\commons-logging.jar c:\java\jakarta-struts-1.1\lib\commons-validator.jar c:\java\jakarta-  
struts-1-1\lib\digester.jar c:\java\jakarta-struts-1.1\lib\fileupload.jar c:\java\jakarta-struts-  
1.1\lib\jakarta-oro.jar c:\java\jakarta-struts-1.1\lib\struts.jar
```

Add the following files to the **c:\java\MiniHR\WEB-INF\tlds** directory:

```
c:\java\jakarta-struts-1.1\lib\struts-bean.tld c:\java\jakarta-struts-1.1\lib\struts-html.tld  
c:\java\jakarta-struts-1.1\lib\struts-logic.tld
```

Assuming that you have installed Struts at **c:\java\jakarta-struts-1.1**, installed Tomcat at **c:\java\jakarta-tomcat-4.1.27**, and placed the Mini HR application files at **c:\java\MiniHR**, the following command line will compile the Mini HR application when run from the **c:\java\MiniHR** directory:

```
javac -classpath WEB-INF\lib\commons-beanutils.jar;          WEB-INF\lib\commons-  
collections.jar;          WEB-INF\lib\commons-lang.jar;          WEB-INF\lib\commons-  
logging.jar;          WEB-INF\lib\commons-validator.jar;          WEB-  
INF\lib\digester.jar;          WEB-INF\lib\fileupload.jar;          WEB-INF\lib\jakarta-  
oro.jar;          WEB-INF\lib\struts.jar;          C:\java\jakarta-tomcat-  
4.1.27\common\lib\servlet.jar          WEB-  
INF\src\com\jamesholmes\minihhr\*.java          -d WEB-INF\classes
```

Notice that you must specify the path to each **.jar** file explicitly. Of course, if you update CLASSPATH, this explicit specification is not needed. You should also notice that the compiled code will be placed into the **WEB-INF\classes** directory, as specified by the **-d WEB-INF\classes** section of the command line. Remember from the earlier discussion that this is the standard Web Archive directory that J2EE servlet containers will look in for compiled Web application code.

## Packaging the Application

Because Struts applications are standard J2EE Web applications, this application will be packaged using the standard J2EE Web Archive format. Packaging the application as a **.war** file allows the application to be easily deployed on any J2EE-compliant servlet engine with ease. Because you arranged the files for the Mini HR application in the standard Web Archive directory structure, packaging them into a **.war** file is straightforward.

Following is the command line for creating a **MiniHR.war** file, assuming that you run the command from the Mini HR application directory (**c:\java\MiniHR**):

```
jar cf MiniHR.war *
```

After you run the command, a **MiniHR.war** file will be created and ready for deployment.

## Running the Application

Once you have packaged your application, running it is as simple as placing the Web Archive file into Tomcat's **webapps** directory and then starting up Tomcat. By default, Tomcat starts up on port 8080, thus the server can be accessed at **http://localhost:8080/**. To access the Mini HR application, point your browser to **http://localhost:8080/MiniHR/**. You'll notice that the name of your Web Archive file is used for the URL of your application. Because you packaged the Mini HR application in a file called **MiniHR.war**, **/MiniHR/** is used for the application's URL.

When you first access the **http://localhost:8080/MiniHR/** URL, **index.jsp** will be run, because it was specified as the Welcome File in the **web.xml** deployment descriptor. From the opening page, select the Search for Employees link. The Search page allows you to search for employees by name or social security number. If you do not enter any search criteria, an error message will be shown on the page. Similarly, if you enter an invalid social security number, an error message will be shown on the page after you click the Search button.

Figures 2-2 through 2-5 show Mini HR in action.



Figure 2-2: The opening screen

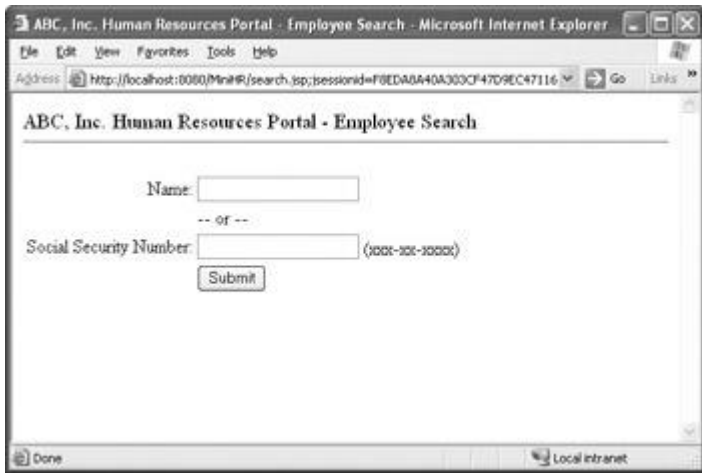


Figure 2-3: The Employee Search screen



Figure 2-4: The Employee Search screen with a validation error

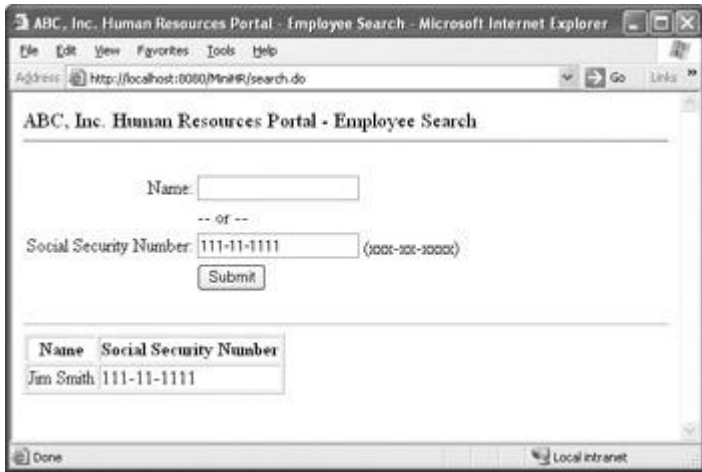


Figure 2-5: The Employee Search screen with search results