**实验二 第一个Java EE Web应用及学习使用Session和Cookie**

# 实验目的

1. 学习使用eclipse开发Java EE Web应用程序的方法；
2. 学习Servlet的开发方法和Servlet在DD文件中如何配置；
3. 学习service()方法的编写方法，学会HttpServletRequest和HttpServletResponse的使用方法；
4. 学习Servlet上下文的使用方法。
5. 学习Web容器Session管理机理；
6. 了解Session寿命周期；
7. 学习使用用户自定义Cookie；
8. 学习Session监听器。

# 实验内容

## 第一个Java EE Web应用

1. 建立开发环境：JDK+Eclipse+JBoss；
2. 开发一个只有一个Servlet的Java EE Web应用，在页面上打印当前时间；
3. 开发一个简单的啤酒选择应用；
4. 开发一个下载文件的Java EE Web应用；
5. 使用Servlet上下文。

## 学习使用Session和Cookie

1) 开发一个简单的购物车；

2) 开发一个简单的用户登录提示系统；

3) 设定Session生命周期；

4) 开发一个下载文件的Java EE Web应用；

5) 开发一个Session监听器。

# 实验仪器、设备

1. PC机最低配置：2G Hz以上CPU；1G以上内存；1G自由硬盘空间；
2. JDK1.6.0以上；
3. Eclipse Ganymede；
4. Tomcat 8。

# 实验原理

1. B/S编程模式；
2. 动态Web编程；
3. Servlet寿命周期管理；
4. Servlet请求和相应；
5. Servlet容器上下文。
6. B/S编程模式；
7. 动态Web编程；
8. Session管理机制；
9. Session寿命周期。

# 实验步骤

## 建立开发环境

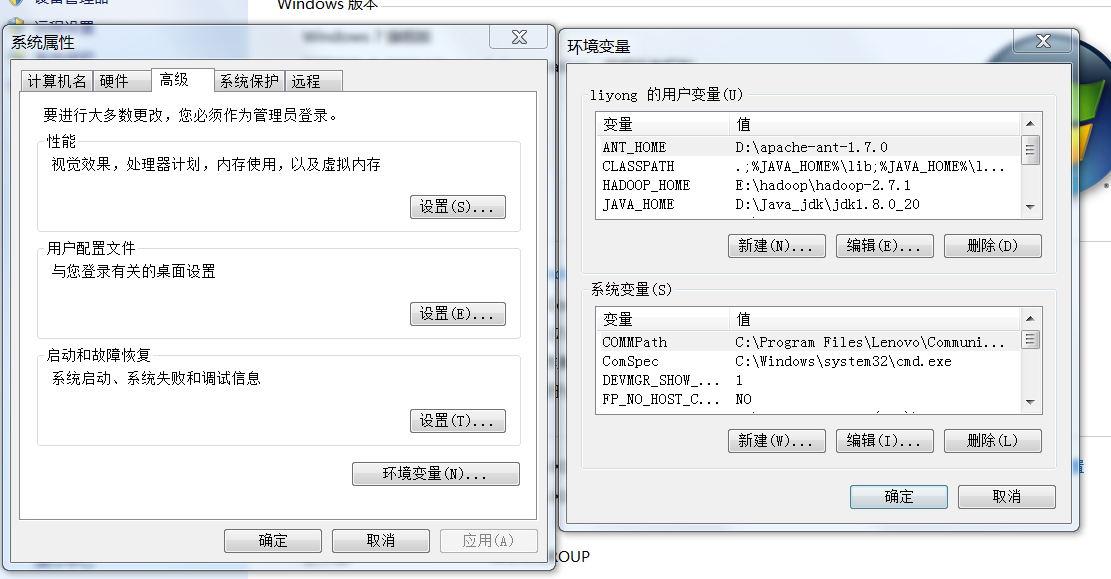
### 安装JDK1.8

1. 先查看你的实验用机上是否安装有其他版本的JDK，若没有则进行下列步骤，若已安装，则先卸载它（若已安装的JDK版本号为1.8.0.X，可不必卸载，跳过本步骤不再安装JDK）；
2. 在FTP中找到JDK 1.8的安装文件jdk-1\_5\_0\_14-windows -i586-p.exe，双击该文件，按照提示信息安装JDK和JRE。注意区分自己机器是32位还是64位；
3. 在控制面板中选取“系统”，打开“系统属性”窗口；





1. 选择“高级”Tab页，点击“环境变量”按钮，在“环境变量”窗口中“新建”一个用户变量：JAVA\_HOME，其值设置为JDK的安装目录。

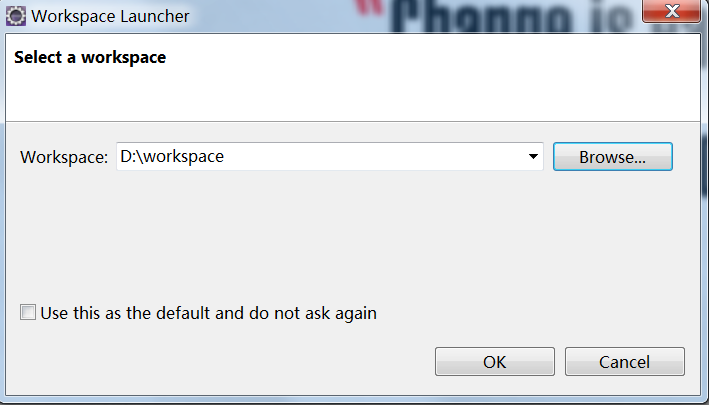


### 安装Tomcat 8应用服务器和eclipse开发环境

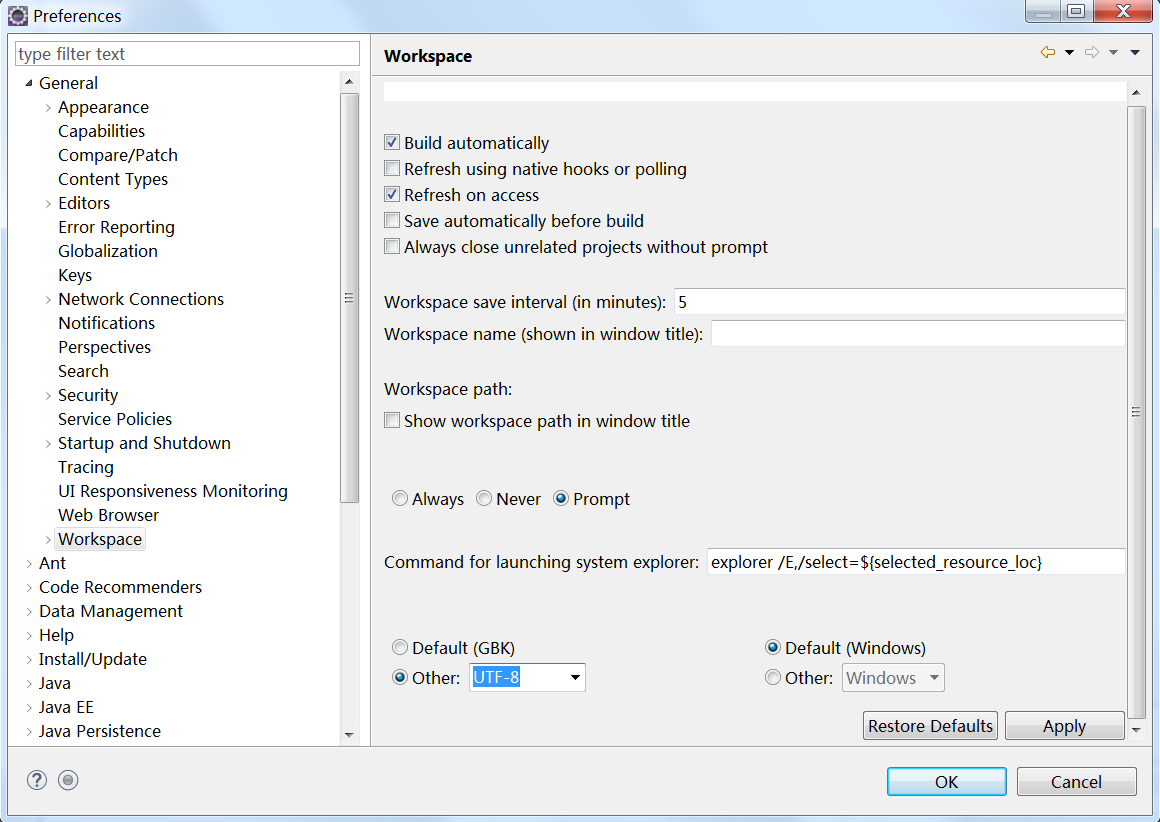
1. 在FTP中找到apache-tomcat-8.0.46-windows-x64.zip，解压缩该文件到硬盘，假设解压目录为TOMCAT\_HOME，解压目录中不要含有中文和空格；
2. 在FTP中找到eclipse-jee-mars-R-win32-x86\_64.zip，解压缩该文件到硬盘，假设解压目录为ECLIPSE\_HOME，解压目录中不要含有中文和空格。

### 设置开发环境

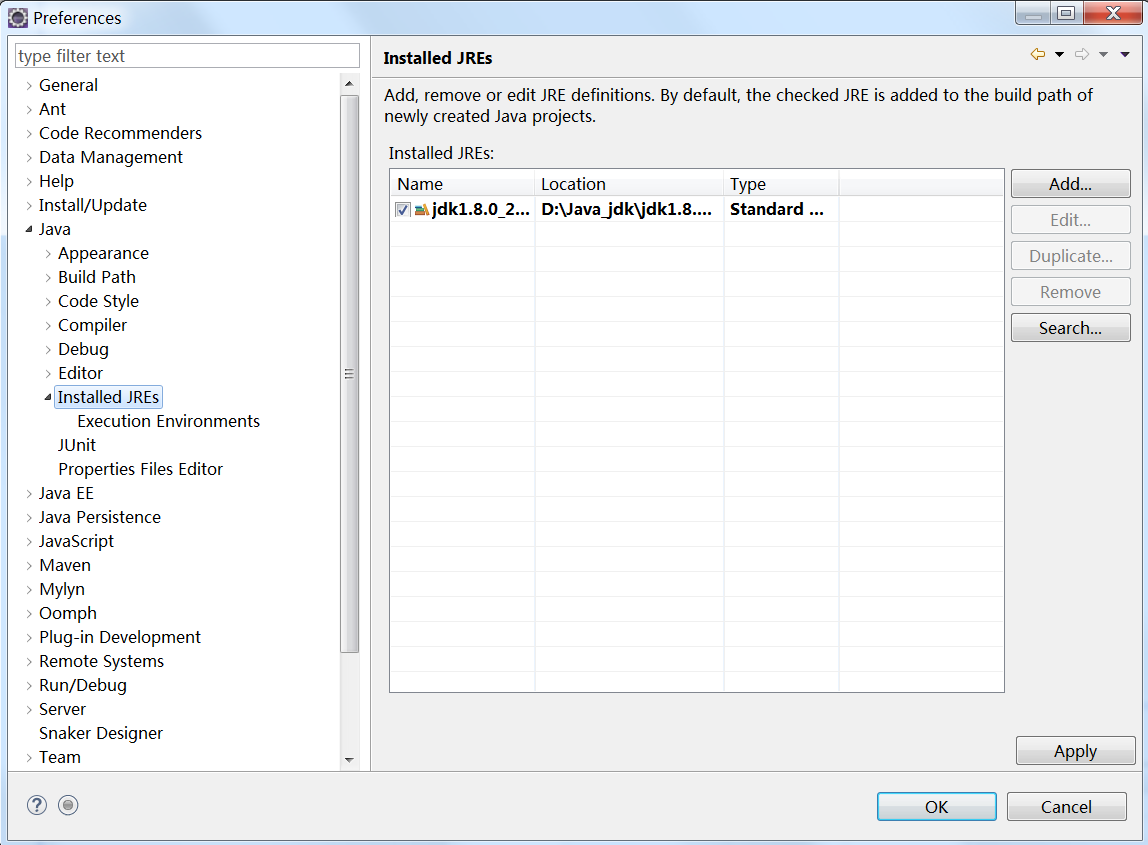
1. 双击ECLIPSE\_HOME\eclipse.exe，运行eclipse集成开发环境。注意设定你自己的workspace目录，目录中不要含有中文和空格。



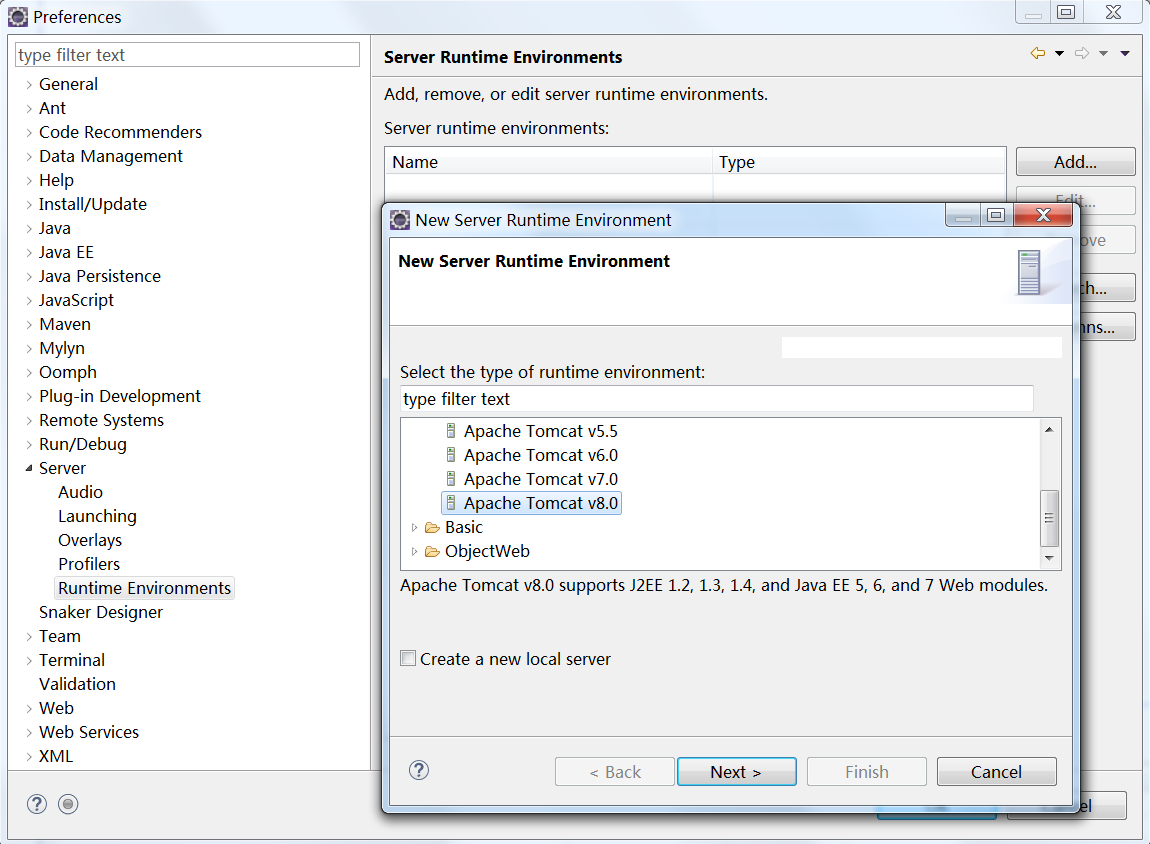
1. 选取Window🡪Preferences…弹出Preferences设置窗口，在其中的左栏树上选取General/Workspace，设置字符集为UTF-8。

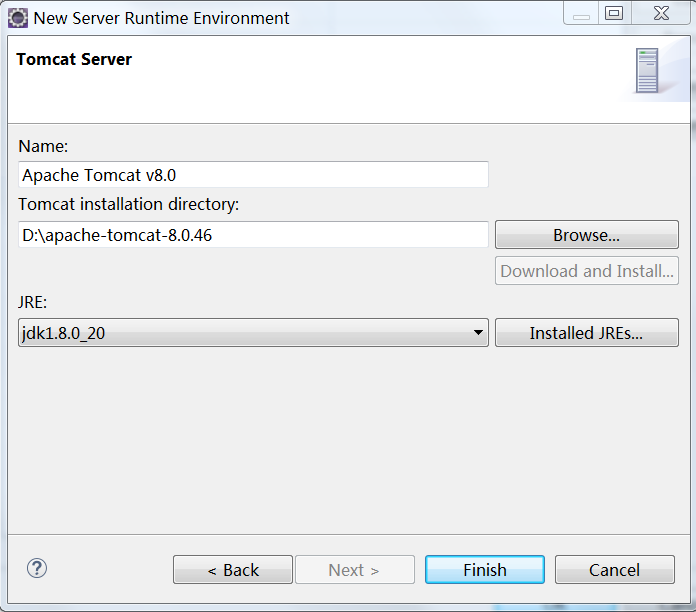


1. 选取Window🡪Preferences…弹出Preferences设置窗口，在其中的左栏树上选取Java/Installed JREs，察看是否设定了JRE，如果没有，则设置你安装的JRE目录，建议选择JDK版本便于进行Debug和源码学习。

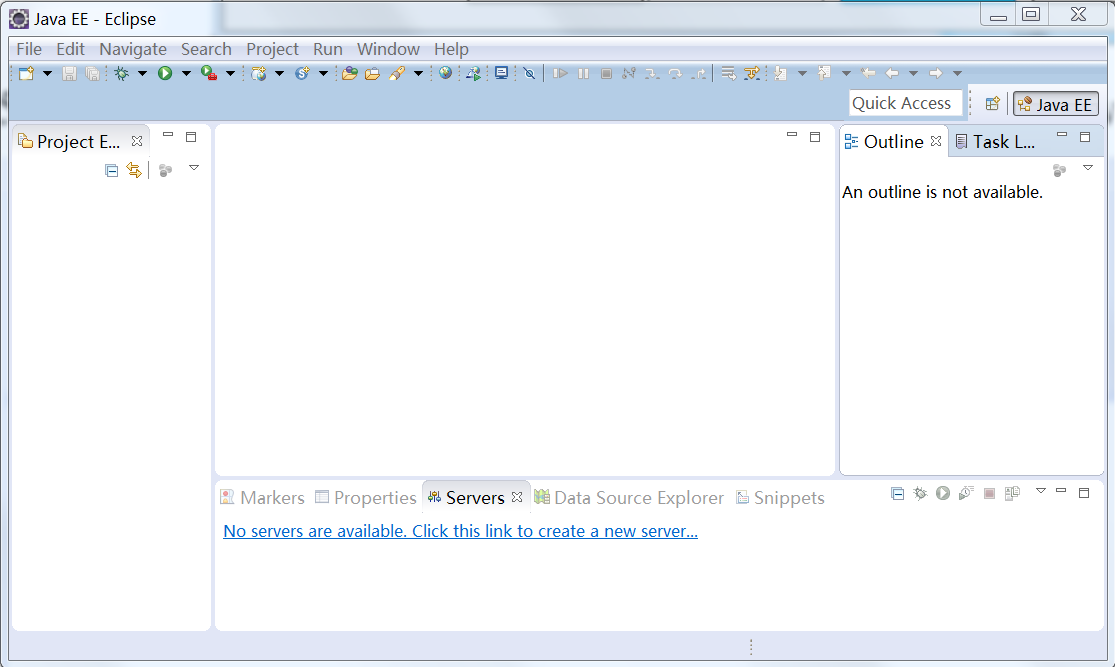


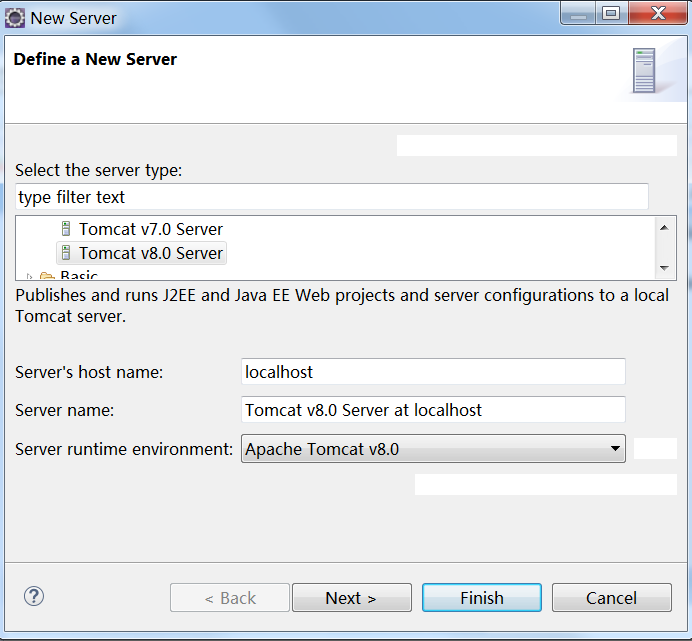
1. 选取Window🡪Preferences…弹出Preferences设置窗口，在其中的左栏树上选取Server/Runtimes Environments，选择Tomcat 8作为Java EE应用服务器，在下一步设定Application Server Directory为TOMCAT\_HOME；



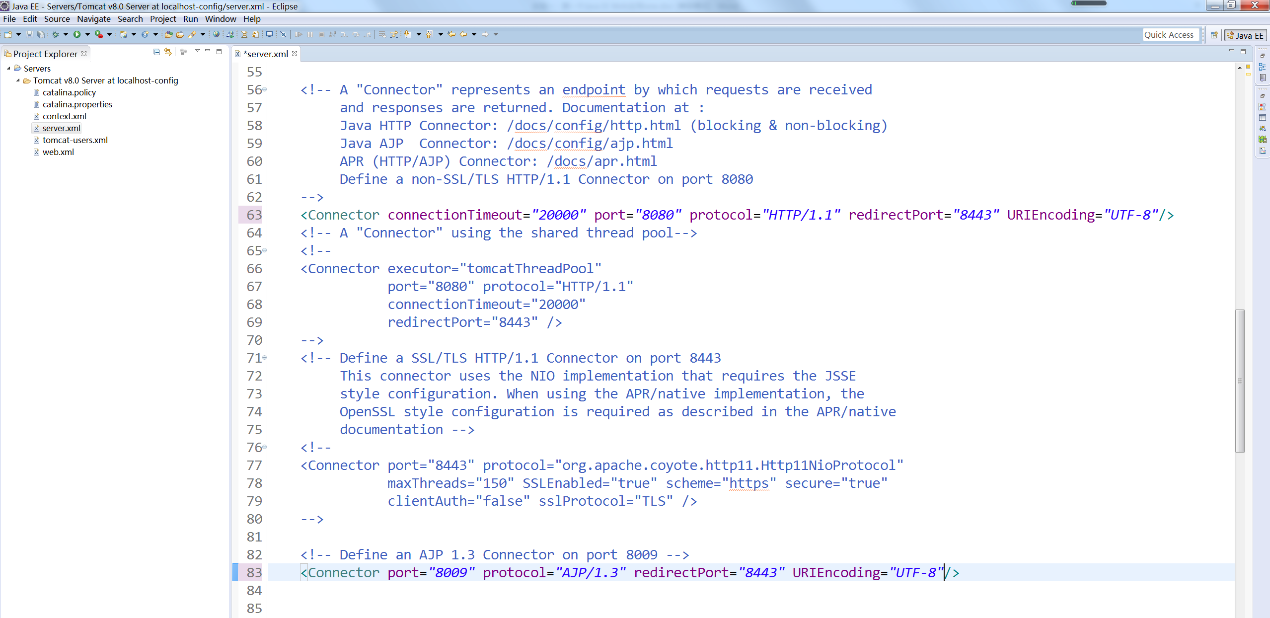


1. 在Eclipse的server窗口中，点击Create a new server…，弹出Tomcat设置窗口，选择上一步设置的Tomcat。





1. 在project窗口出现Server节点，打开server.xml，在2个Connector节点添加URIEncoding="UTF-8"属性，避免中文参数问题。



1. 完毕！

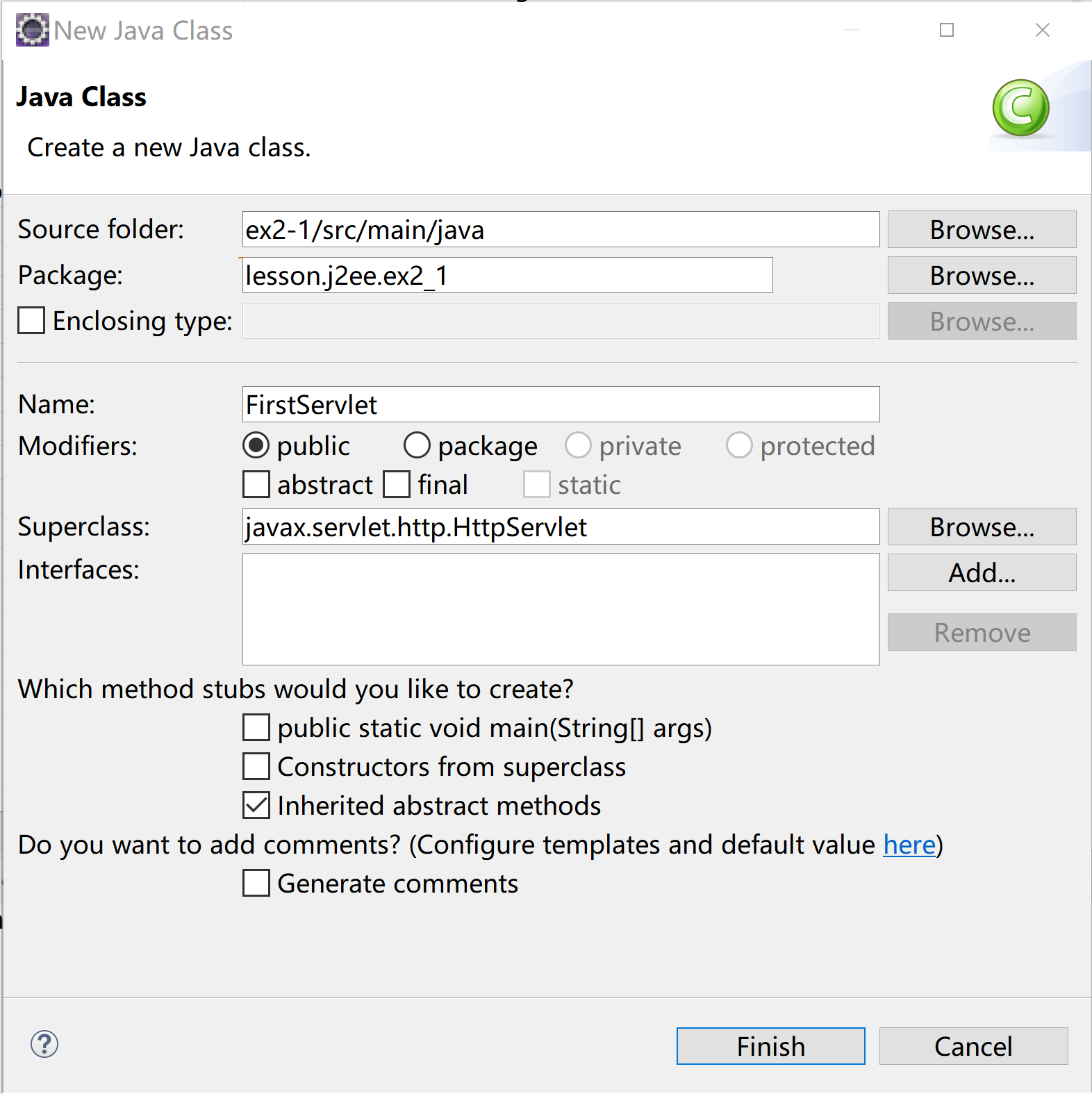
## 开发仅有一个Servlet的Web应用

### 新建一个动态Web项目

1. 打开Eclipse workbench，选择菜单File🡪New🡪Project…打开New Project向导窗口；
2. 选择Web/Dynamic Web Project，Next，命名新的动态Web项目名为ex2-1，Target Runtime为Tomcat 8；
3. 点击下一步，在Project Facets中配置Dynamic Web Module版本为2.4，完成。

### 开发FirstServlet

1. 在Project Explorer中的Java Resources: src文件夹上点击鼠标右键，选择New🡪Class，在弹出窗口中键入下图中内容；



1. 编写FirstServlet.java；
2. 双击Deployment Descriptor: <web app>，编辑web.xml，使之成为web.xml。

## 编译、部署、运行和使用Web应用

### 编译、部署、运行Web应用

* 1. 在ex2-1项目上点击鼠标右键，在弹出菜单上选取Run As🡪Run on Server；
  2. 在Run on Server窗口，选择Choose an existing server，并选择Tomcat V8.0 Server at localhost，下一步。查看是否ex2-1被配置到Tomcat服务器上。完成，自动实现部署并运行。

### 在浏览器上执行Web应用

在IE浏览器中键入地址：<http://localhost:8080/ex2-1/Serv1>，运行第一个Servlet。

## 开发一个处理Post方法HTTP请求的Servlet

1. 根据第三章课件中第13页POST方法的Web表单代码，编写一个HTML页面，作为捕获用户提交数据的请求页面，命名为[select.html](file:///C:\Users\Eric-Feng's%20X1\AppData\Local\Temp\Temp1_JavaEE实验1.zip\JavaEE瀹為獙1\ex1\select.html)，放到WebContent目录下；
2. 编写处理POST请求数据的Servlet：[SelectBeerServlet.java](C:\\Users\\Eric-Feng's X1\\AppData\\Local\\Temp\\Temp1_JavaEE实验1.zip\\JavaEE瀹為獙1\\ex1\\SelectBeerServlet.java)，在其doPost( )方法中将提交的color数据显示在响应页面上，并配置相应的DD。
3. 按照5.3节的方法编译、部署、运行和使用Web应用。

## 自己动手完成以下工作

1. 根据第三章课件中第17页的代码，编写一个向客户端发送非HTML数据的Servlet：CodeReturn.java。其中发送的jar文件为：[bookCode.jar](C:\\Users\\Eric-Feng's X1\\AppData\\Local\\Temp\\Temp1_JavaEE实验1.zip\\JavaEE瀹為獙1\\ex1\\bookCode.jar)，将它放到WebContent目录下的resources子目录中。
2. 根据第四章课件中第6页、第7页和第8页的内容，编写一个使用Servlet初始化参数和上下文初始化参数的Servlet: TestInitParams.java，并配置其DD，在其中既包含<init-param>又包含<context-param>。部署并运行它，查看运行结果。

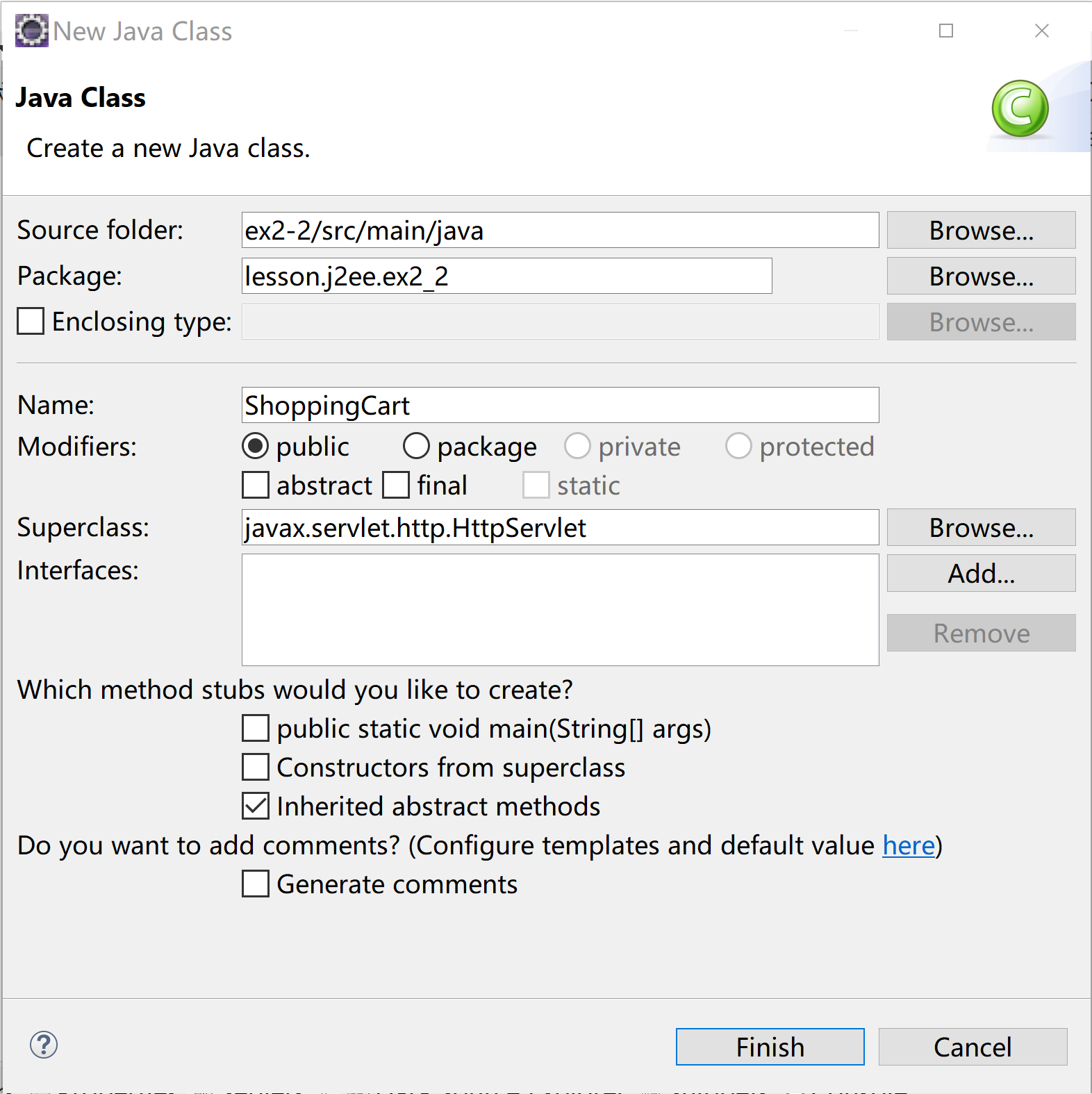
## 开发一个简单的购物车

### 新建一个动态Web项目

1. 打开Eclipse workbench，选择菜单File🡪New🡪Project…打开New Project向导窗口；
2. 选择Web/Dynamic Web Project，Next，命名新的动态Web项目名为ex2-2，Target Runtime为Tomcat v8.0；
3. 点击下一步，在Project Facets中配置Dynamic Web Module版本为2.4，完成。

### 开发ShoppingCart

1. 在Project Explorer中的Java Resources: src文件夹上点击鼠标右键，选择New🡪Class，在弹出窗口中键入下图中内容；



1. 编写ShoppingCart.java；
2. 双击Deployment Descriptor: <web app>，编辑web.xml，使之成为web.xml。

### 开发图书显示页面

* 1. 在Project Explorer中的WebContent文件夹上点击鼠标右键，选择New🡪HTML，命名为bookstore.html，完成；
  2. 编辑bookstore.html内容，使之成为bookstore.html。

## 编译、部署、运行和使用Web应用

### 编译、部署、运行Web应用

* 1. 在ex2-2项目上点击鼠标右键，在弹出菜单上选取Run As🡪Run on Server；
  2. 在Run on Server窗口，选择Choose an existing server，并选择Tomcat v8.0 Server at localhost，下一步。查看是否ex2-2被配置到Tomcat服务器上。完成，自动实现部署并运行。

### 在浏览器上执行Web应用

在IE浏览器中键入地址：http://localhost:8080/ex2-2/bookstore.html，运行选购图书页面。。

## 开发一个简单的用户登陆提示系统

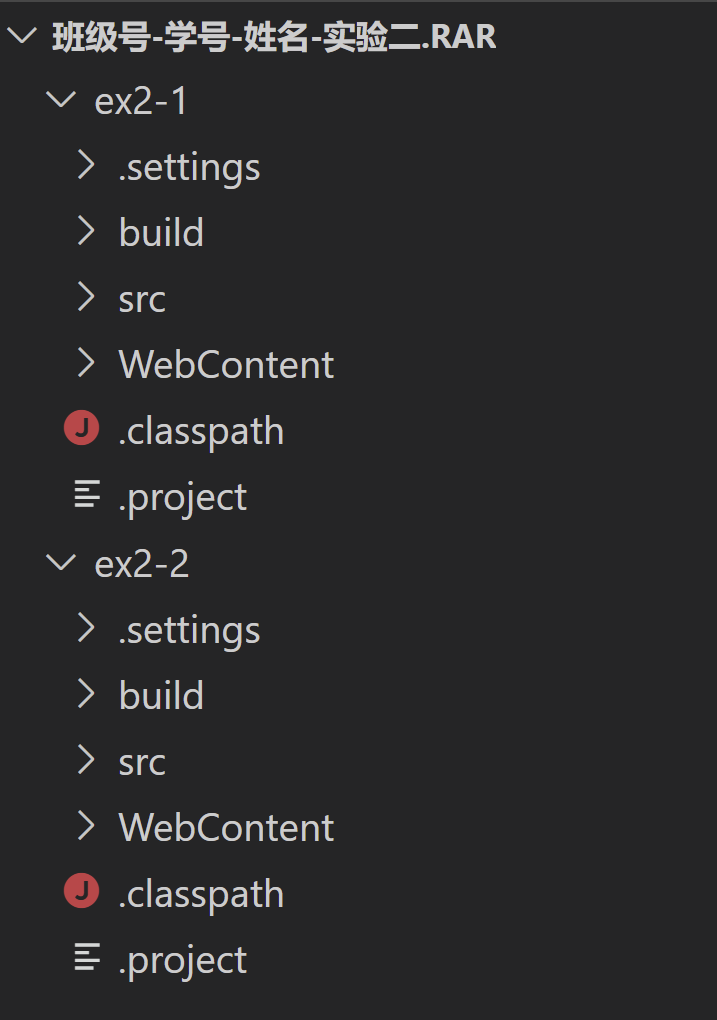
1. 根据第五章课件，编写一个HTML页面，作为捕获用户提交数据的请求页面，命名为UserLogin.html，放到WebContent目录下；
2. 编写Servlet：Login.java，在其doPost( )方法中将用户名称加入Cookie中，然后显示出来，并配置相应的DD。
3. 按照5.7节的方法编译、部署、运行和使用Web应用。

## 自己动手完成以下工作

1. 根据第五章课件中第13页给出的前两种销毁session的方法，对以上购物车应用进行修改，设定session寿命为3分钟，然后观察修改后的运行效果；
2. 参考第五章课件中第18页的代码为以上购物车应用添加一个HttpSessionAttributeListener监听器：BookAddedListener.java，用于监听添加session属性item1, item2, …以及itemCount；监听itemCount属性的更新，并配置相应的DD。在控制台上打印出item1, item2, …和itemCount的值；
3. 在用户登录系统中加入密码设置，要求用Cookie存储密码，并能实现自动登录。

# 提交要求

1. 提交源程序代码，并在程序中写java doc说明；
2. 打包文件名格式：班级号-学号-姓名-实验二.rar；
3. 打包文件夹结构如下图。



# 实验注意事项

1. 看清实验指导书后再动手；
2. 按实验指导书动手做实验的过程中要勤于思考，与课堂授课内容比照理解。

# 思考题

1. 一个Servlet中是否能够同时有doGet()和doPost()方法？做一个实验证明你的结论。
2. Session和Cookie的区别。