**实验6 Servlet实验二**

**一、实验目的**

1、理解ServletContext对象作为Web应用共享全局量的含义

2、通过Servlet数据库操作

3、理解注解Annotation

**二、实验内容**

1、ServletContext常用方法练习；

2、Servlet实现JDBC连接；

3、使用注解Annotation标注配置信息

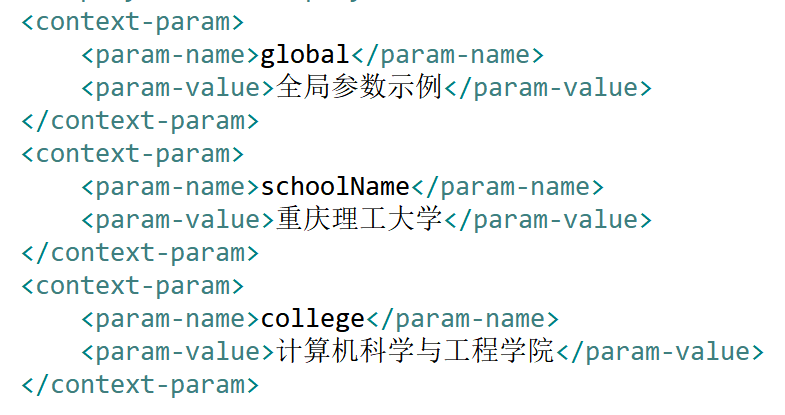
**三、实验步骤**

1. **ServletContext对象常用方法练习**

* **获取Web应用的初始化信息（见实验5）**

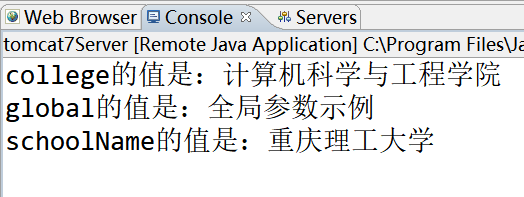
**（/CourseEx/ContextTest)**

获取web.xml中多个参数的名字和值，若配置文件中参数如下所示：



获取Context对象：ServletContext ctx = this.getServletContext();

通过ctx.**getParameterNames()**返回一个Enumeration类型的值，通过该对象的hasMoreElements()进行判断并遍历，通过String pname = (String) params.nextElement()得到参数名，通过ctx.getInitParameter()得到参数值，输出获取的所有参数名和对应的值。



* **多个Servlet对象共享数据**

新建两个不同的Servlet，在一个Servlet中通过ServletContext.setAttribute()设置一个全局属性counter，在另一个Servlet中通过ServletContext.getAttribute()获得这个属性，体会**数据共享**

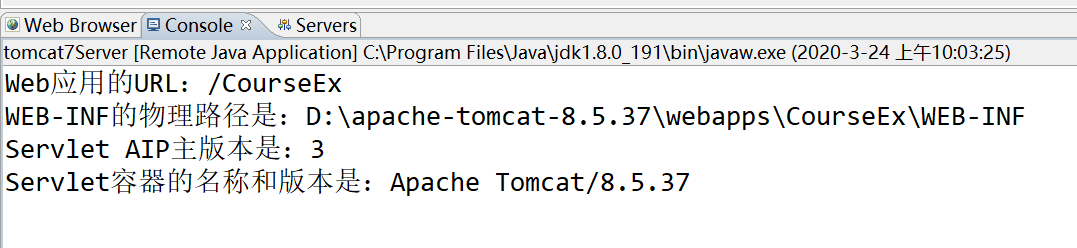
（/CourseEx/SetServletContext + /CourseEx/GetServletContext）

* **获取Web应用的基础信息和文件信息**

新建一个Servlet（/CourseEx/ContextTest）， 在控制台中输出当前Web应用的根路径、名称、服务器中文件的物理路径、Tomcat版本等

获取ServletContext对象 application

|  |  |  |
| --- | --- | --- |
| **序号** | **方法名** | **说明** |
| 1 | String getContextPath() | 返回当前Web应用的根路径 |
| 2 | String getRealPath(String path) | 返回指定路径在服务器文件系统中的绝对路径 |
| 3 | int getMajorVersion() | Servlet AIP主版本 |
| 4 | String getServerInfo() | Servlet容器的名称和版本 |
| 5 | String getServletContextName() | 返回当前Web应用的名称 |

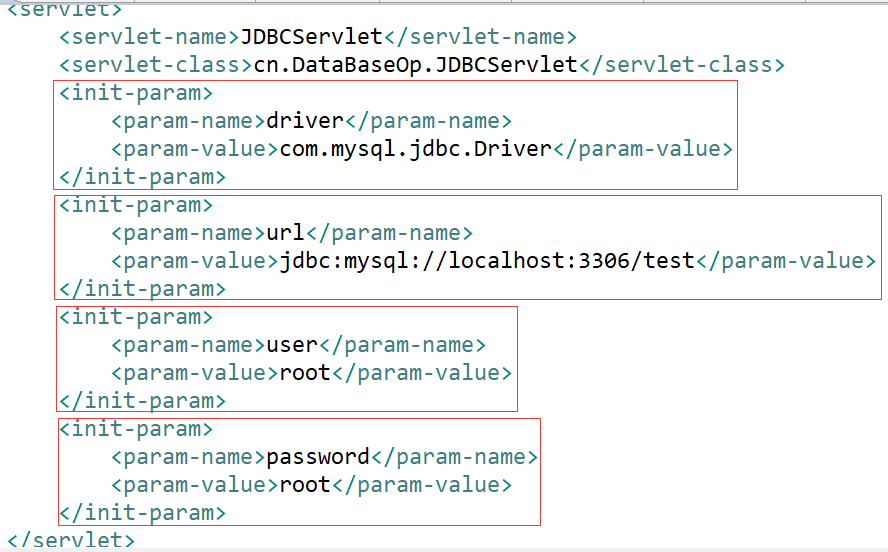


1. **Servlet实现JDBC连接**

创建JDBCServlet，继承HttpServlet，编写如下方法：

(/CourseEx/JDBCServlet)DB JavaBean

* 定义用于存储驱动、连接、用户名、密码的String类型变量；
* 配置web.xml文件，将数据库连接字符串、用户名和密码存入其中



对应的连接字符串

* 在**Init()**方法中获取上述参数的值

使用getInitParameter()

* 在**doGet()**方法中实现数据库的操作 加载 连接 SQL 结果 操作 关闭

可在方法的开始处设置页面的中文编码格式：

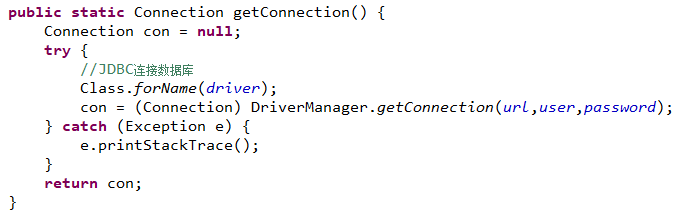
response.setContentType("text/html;charset=utf-8");

步骤参见数据库连接JDBC，至少实现增删改查中的基本操作，如向表中插入数据，或查询数据并显示在页面上。

* 可以对数据库的基本操作进行简单封装，如连接、关闭、增、删、改、查等

例如：创建Conncetion()方法用于数据库的连接，并返回该连接；

创建close()关闭对象和连接

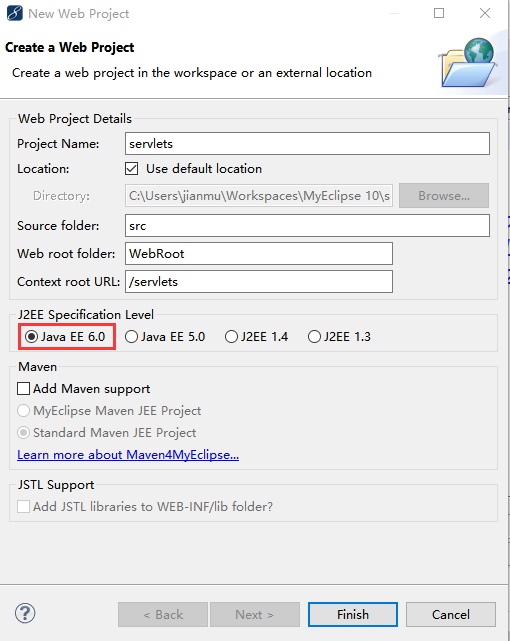


1. **使用注解Annotation标注配置信息**

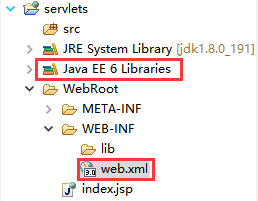
从Servlet3.0开始不必在web.xml文件中配置Web组件，而使用Annotation标注来配置相关信息。以下练习通过注解标注来设置Servlet名称、url和初始化参数。

### （1）创建WebProject

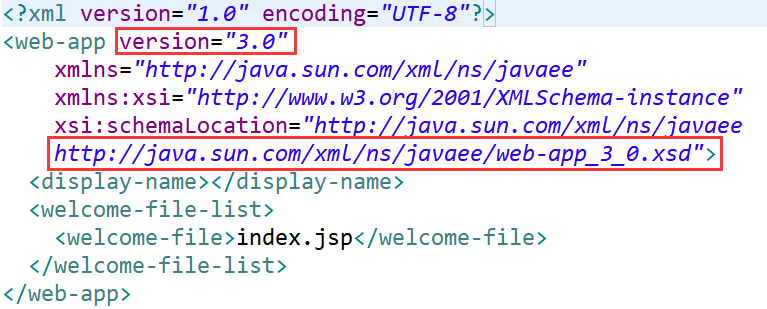
使用MyEclipse10新建一个Web Project，选择Java EE6.0，如下图：



创建好的项目如下图所示：



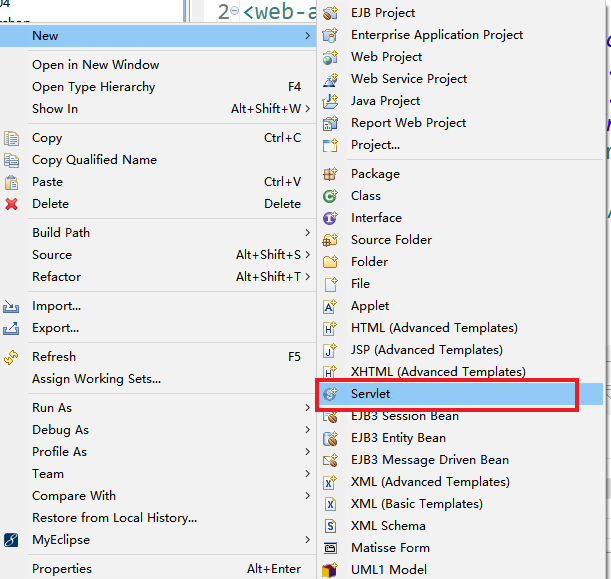
项目的web.xml文件内容如下：

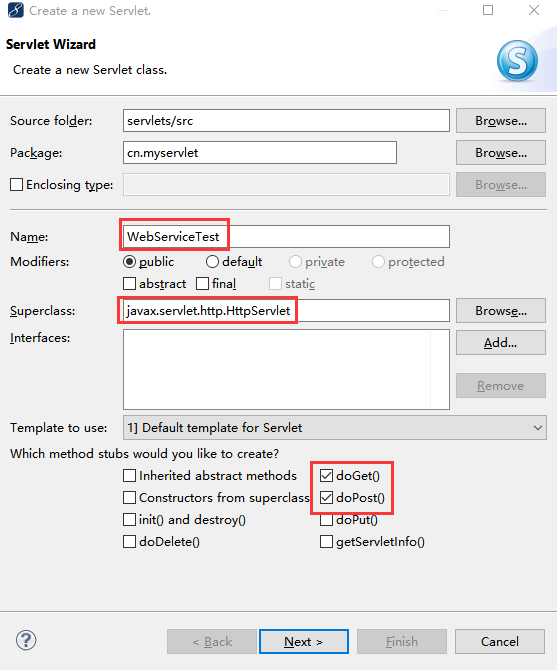


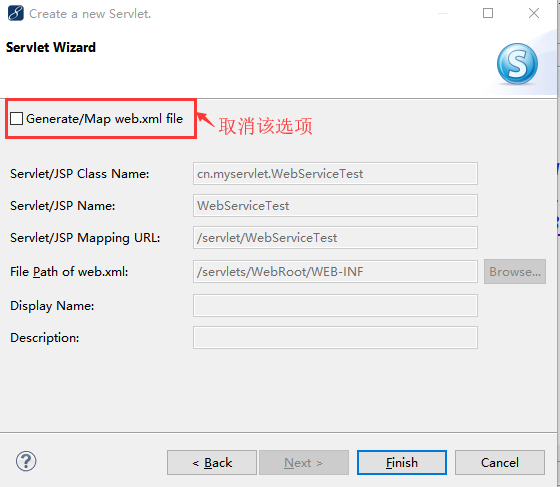
注意web.xml文件中的version信息，**version="3.0"**和**http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd**这就是表示当前使用的是Servlet3.0的版本。

### （2）开发Servlet3.0程序

* 新建一个Servlet，写上Servlet的Name，如下所示：







取消“Generate/Map web.xml file”选项，这样建好一个Servlet后，在web.xml文件中不会自动生成任何关于这个Servlet的描述信息。

#### 使用注解描述Servlet

@WebFilter

在Servlet3.0中，可以使用@WebServlet注解将一个继承于javax.servlet.http.HttpServlet的类标注为可以处理用户请求的Servlet。

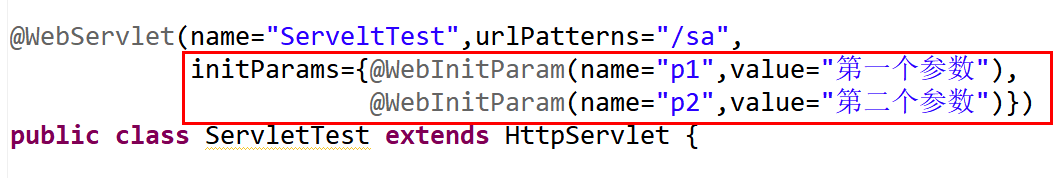
|  |  |  |
| --- | --- | --- |
| @WebServlet注解的相关属性 | | |
| 属性名 | | 类型 | 描述 |
| description | | String | Servlet的描述信息 |
| displayName | | String | Servlet的显示名称 |
| LoadOnStartup | | int | Servlet加载顺序 |
| **initParams** | | **WebInitParam[]** | **一组Servlet的初始化参数{(),() }** |
| **name** | | **String** | **Servlet的名称** |
| **urlPatterns** | | **String[]** | **Servlet的访问URL** |
| **value** | | **String[]** | **Servlet的访问URL** |



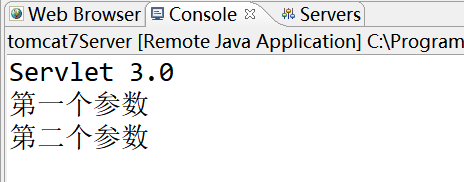
Servlet的访问URL是Servlet的必选属性，才能通过浏览器进行访问，可以选择使用urlPatterns或者value定义。

在浏览器中输入“http://localhost:8080/servlets/webservlet”即可运行，在控制台输出

#### 使用注解获取初始化参数



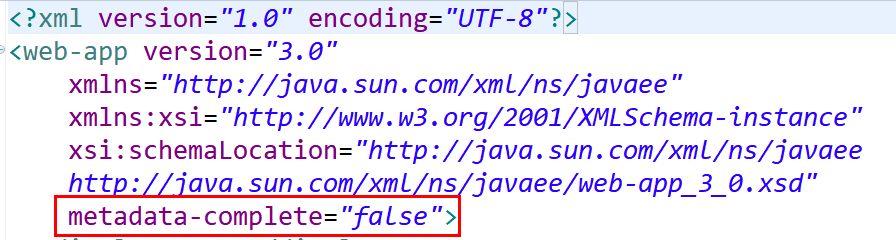
在控制台中输出：



使用注解后有可能出现的问题：

1、500：jdk版本不匹配

2、404：web.xml文件中增加属性



【附】修改Servlet后不重启Tomcat的方法

在Tomcat/conf下的server.xml文件中，在<Host>下配置项目信息<Context>，设置reloadable=”true”即可

