

《基于 JavaEE 的高级程序设计》实验——JSP

实验要求

采用 JSP 技术实现一个简单的登录功能。

实验目标

- 1、能使用 MVC 架构模式实现登录功能。
- 2、能使用 JDBC 连接 MySQL 数据库。
- 3、熟练掌握 Eclipse 的使用，使用调试功能，完成系统调试。
- 4、能使用 Maven 创建项目。
- 5、能使用 Annotation 编写 Servlet。

实验步骤

1、准备实验工具

JDK: JDK8 以上版本

Eclipse: 推荐使用 2019-12 版本

MySQL: 5.5 版本

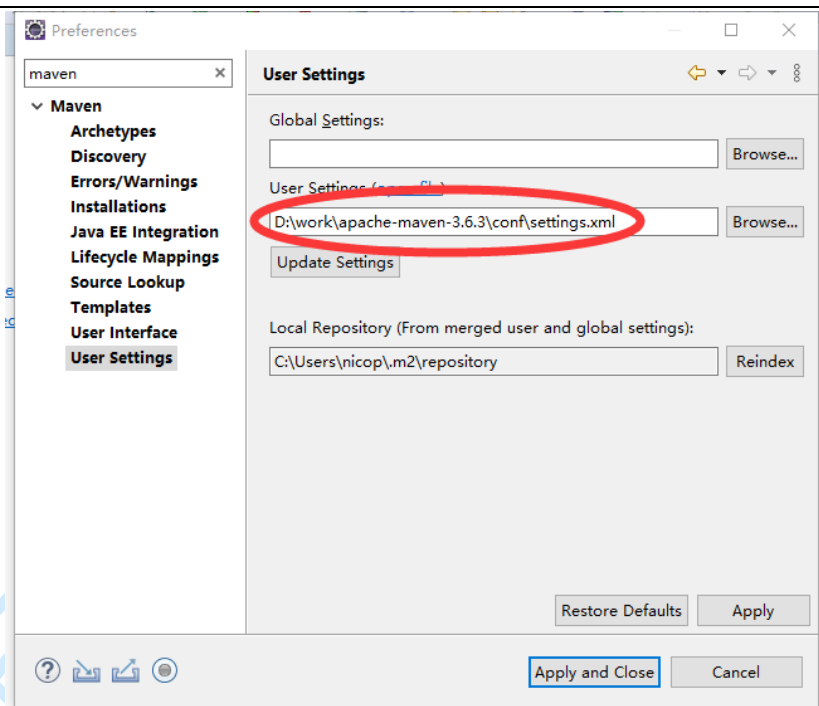
Tomcat: 8.5 版本

注意: Maven 的基本配置

- (1) 解压 Maven 到制定路径下，注意路径不要有空格和中文字符，比如 d:\work 下
- (2) 修改 Maven 的镜像地址，打开 Maven 路径下的 conf 文件夹，在 settings.xml 中增加阿里云 Maven 镜像：

```
.....
<mirror>
  <id>aliyunmaven</id>
  <mirrorOf>*</mirrorOf>
  <name>阿里云公共仓库</name>
  <url>https://maven.aliyun.com/repository/public</url>
</mirror>
.....
```

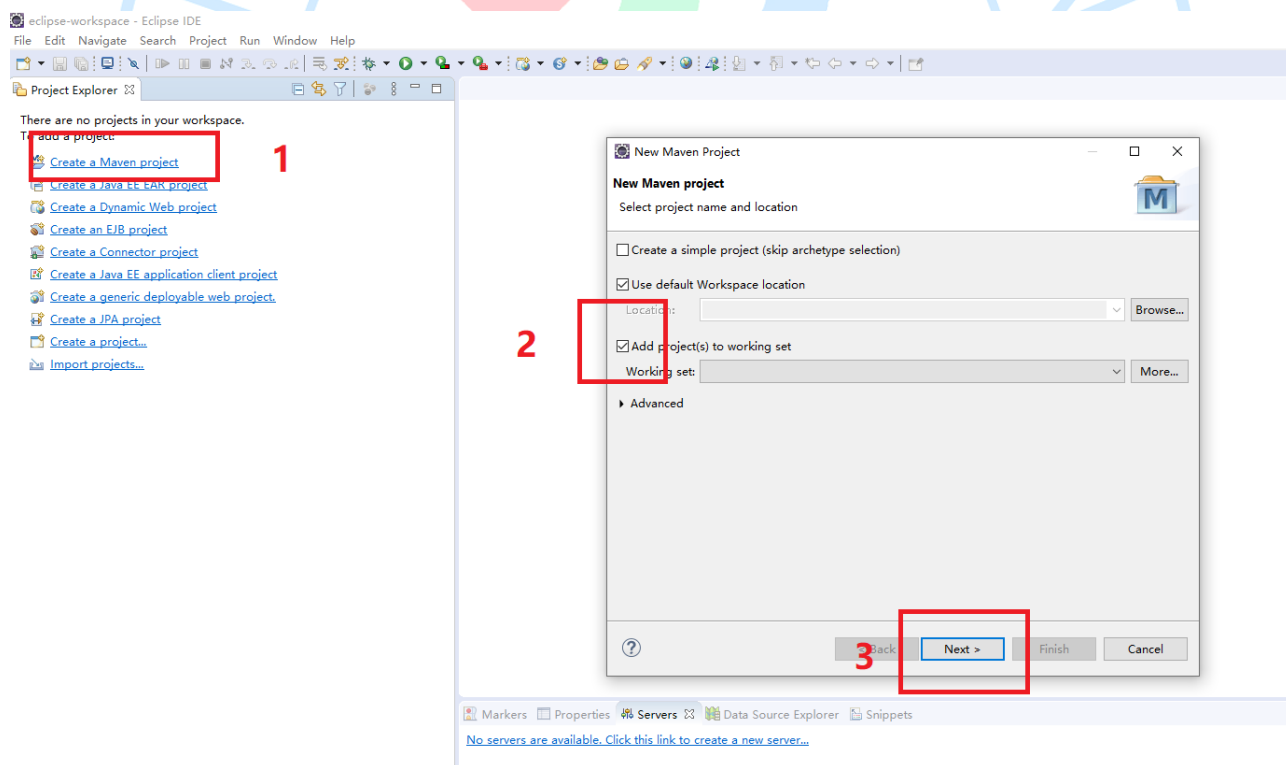
- (3) 在 Eclipse 中配置 Maven 的配置文件，并点击 “Update Settings”:



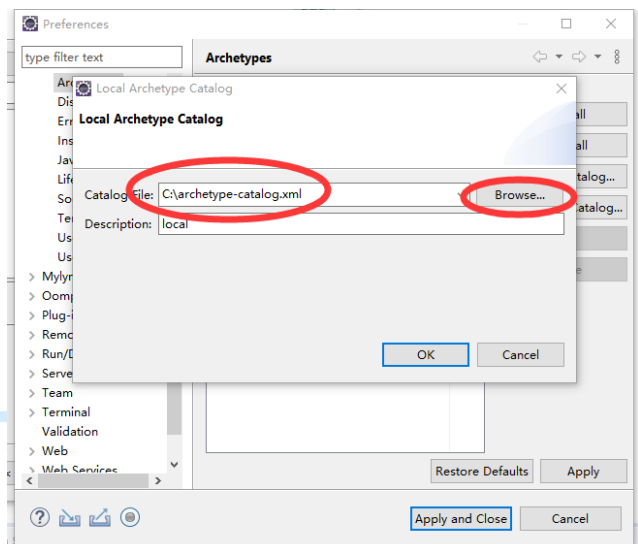
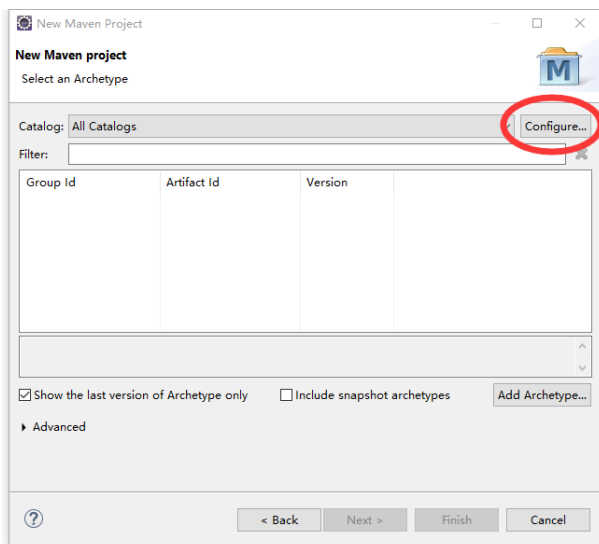
2、Eclipse 中创建 Java Web 项目（推荐 Maven 创建）

（1）创建 Maven 项目

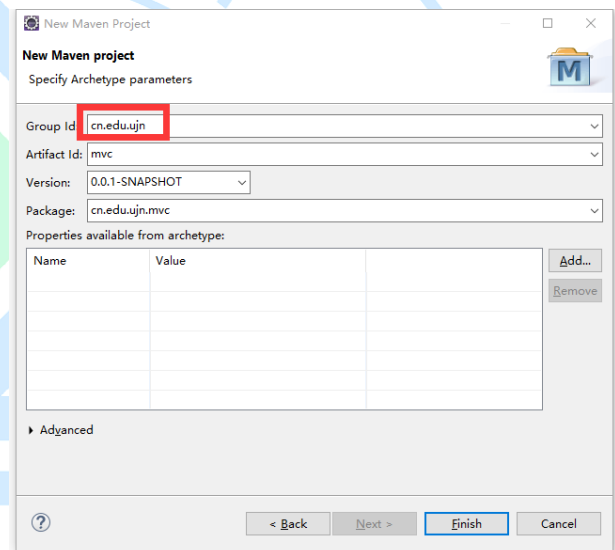
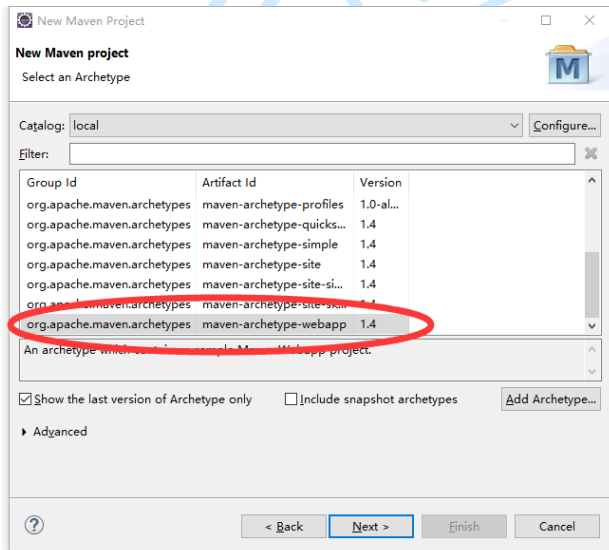
点击“Create a Maven project”：



注意：如果在“Select an Archetype”一步时间较长，可以选择加载本地 archetype 文件（推荐使用我给大家提供的 archetype-catalog-liukun.xml）：



选择“org.apache.maven.archetypes”下的“maven-archetype-webapp”：



(2) 修改 Maven 项目属性

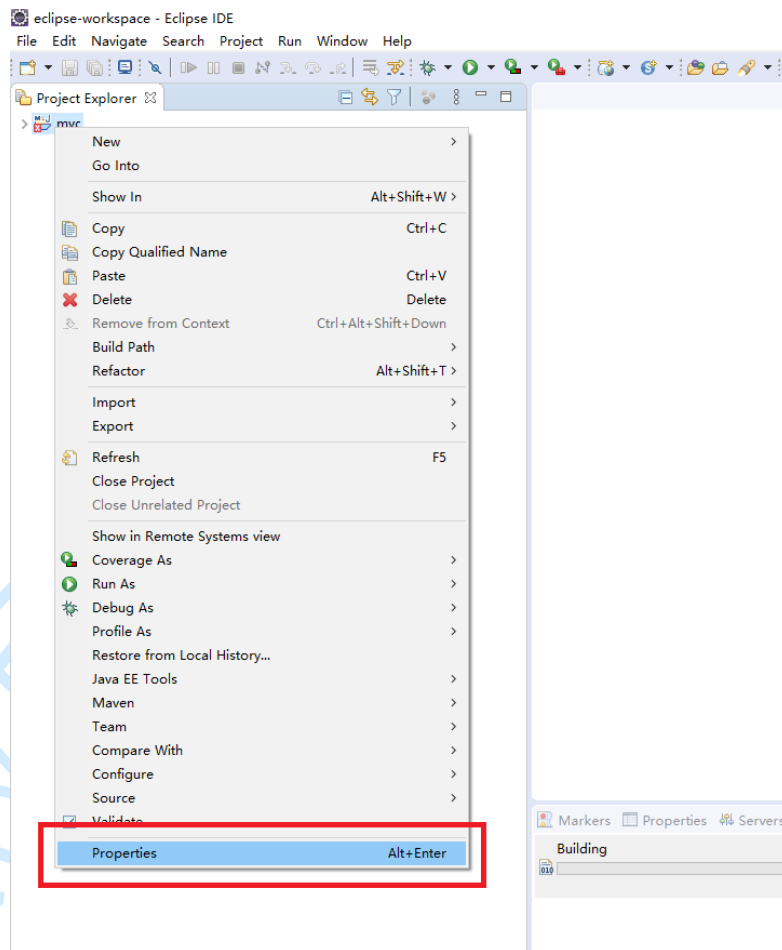
注意：创建完项目后需更改项目的属性

更改 Java Build Path 中的 JDK 为 1.8（Eclipse 自带）

选定 Targeted Runtimes 为 Tomcat v8.5

更改 Java Compiler 中的 JDK 为 1.8

更改 Project Facets 中的 JDK 为 1.8



修改完项目属性后，项目的错误会消失。

3、添加 login.jsp 和 ok.jsp

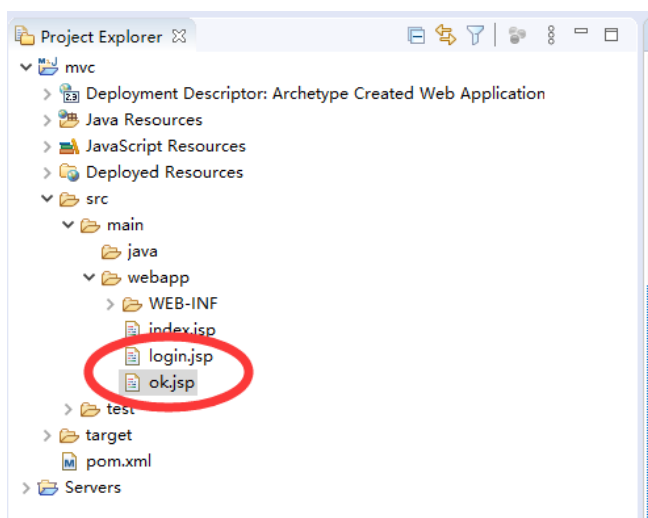
- (1) 在 Maven 配置文件中添加 JDBC 驱动的依赖
打开 pom.xml，添加；

```
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>5.1.48</version>
</dependency>
```

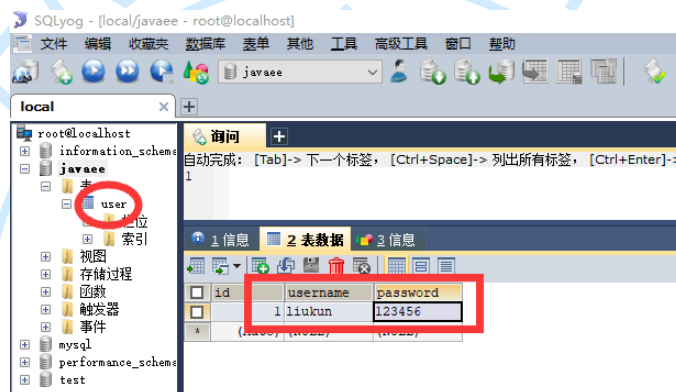
```
<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.11</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>5.1.48</version>
  </dependency>
</dependencies>
```

- (2) 拷贝 JSP 页面到制定路径

注意：Maven 的 Web 项目的 webapp 路径在 src/main/webapp 下：



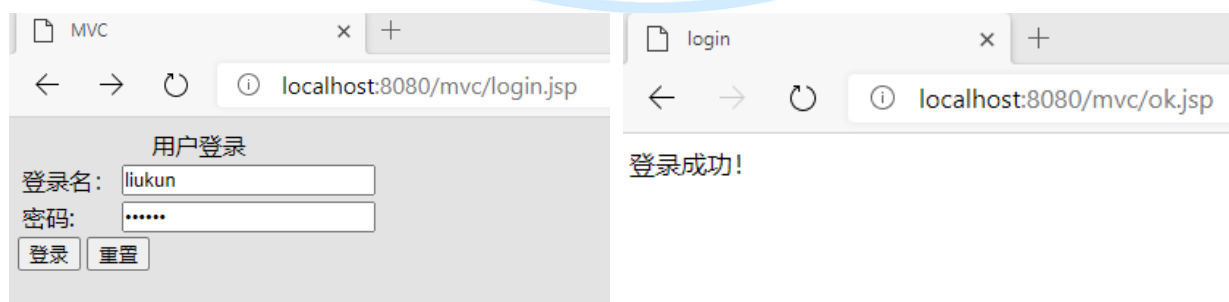
- (3) 创建数据库
打开 SQLyog，导入 javaee.sql:



- (4) 修改 ok.jsp 中的数据库用户名密码

```
.....
Class.forName("org.gjt.mm.mysql.Driver");
String url = "jdbc:mysql://localhost:3306/javaee";
Connection con = DriverManager.getConnection(url, "root", "root");
Statement s = con.createStatement();
ResultSet rs = s.executeQuery("select * from user where username='"
    + username + "' and password='" + pwd + "'");
.....
```

- (5) 测试程序是否能运行



- 4、采用 MVC 架构模式改写程序
Servlet 参考代码:

```

package cn.edu.ujn.mvc.c;

import java.io.IOException;
import java.sql.*;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet("/login")
public class LoginServlet extends HttpServlet {

    @Override
    protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws
ServletException, IOException {
        System.out.println("login-----");
        String username = req.getParameter("username");
        String pwd = req.getParameter("pwd");
        try {
            Class.forName("org.gjt.mm.mysql.Driver");
            String url = "jdbc:mysql://localhost:3306/javaee";
            Connection con = DriverManager.getConnection(url, "root", "root");
            Statement s = con.createStatement();
            ResultSet rs = s.executeQuery("select * from user where username='"
+ username + "' and password='" + pwd + "'");
            if (rs.next()) {
                resp.sendRedirect("success.jsp");
            } else {
                resp.sendRedirect("login.jsp");
            }
        } catch (Exception e) {
            // TODO: handle exception
        }
    }
}

```

5、扩展

- (1) 思考：密码怎么保存？
- (2) 思考：这个程序符合面向对象的“依赖倒置”原则吗？