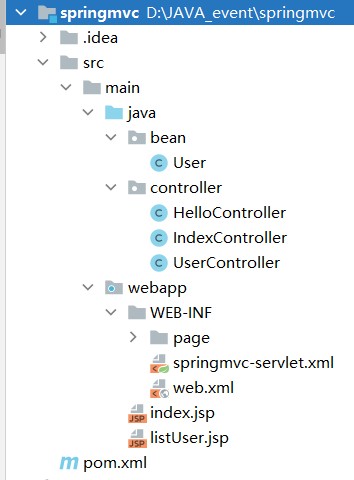
## SpringMVC

选择以webapp模板创建maven项目

### 项目结构



### web.xml配置

web.xml提供路径与servlet的映射关系,

<servlet> 标签下的 <servlet-name> 与 <servlet-mapping> 标签下的 <servlet-name> 必须一样,

<servlet-name>与<servlet-class>可以不一样，但是为了便于理解与维护，一般都会写的一样。

|  |
| --- |
| <!DOCTYPE web-app PUBLIC  "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"  "http://java.sun.com/dtd/web-app\_2\_3.dtd" >  <web-app>  <display-name>Archetype Created Web Application</display-name>  <servlet>  <load-on-startup>10</load-on-startup> //标签代表servlet会和tomcat一起启动，10表示servlet启动顺序  <servlet-name>HelloServlet</servlet-name>  <servlet-class>com.iweb.HelloServlet</servlet-class>  </servlet>  <servlet-mapping>  <servlet-name>HelloServlet</servlet-name>  <url-pattern>/hello</url-pattern>  </servlet-mapping>  </web-app> |

### 以下几种方式是使用get方式提交的

form默认的提交方式

如果通过一个超链访问某个地址

如果在地址栏直接输入某个地址

ajax指定使用get方式的时候

**以下几种方式是使用post方式提交的**

在form上显示设置 method="post"的时候

ajax指定post方式的时候

### response设置编码

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

## JSP

### 依赖

|  |
| --- |
| <dependency>  <groupId>javax.servlet</groupId>  <artifactId>javax.servlet-api</artifactId>  <version>3.0.1</version>  <scope>provided</scope>  </dependency>  <dependency>  <groupId>javax.servlet.jsp</groupId>  <artifactId>jsp-api</artifactId>  <version>2.1</version>  <scope>provided</scope>  </dependency> |

### JSP4个作用域

pageContext 当前页面

requestContext 一次请求

sessionContext 当前会话

applicationContext 全局，所有用户共享

### jsp九个内置对象

request,response,out 请求、响应、输出

pageContext, session,application 当前页面、当前会话、当前全局

page,config,exception 当前对象、servlet初始参数、异常

### Filter过滤器

|  |
| --- |
| public class FirstFilter implements Filter {  @Override  public void destroy() {}  @Override  public void doFilter(ServletRequest req, ServletResponse res, FilterChain chain)  throws IOException, ServletException {  HttpServletRequest request = (HttpServletRequest) req;  HttpServletResponse response = (HttpServletResponse) res;    String ip = request.getRemoteAddr();  String url = request.getRequestURL().toString();  SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");  Date d = new Date();  String date = sdf.format(d);  System.out.printf("%s %s 访问了 %s%n", date, ip, url);  chain.doFilter(request, response);  }  @Override  public void init(FilterConfig arg0) throws ServletException {}  } |

配置web.xml

|  |
| --- |
| <filter>  <filter-name>FirstFilter</filter-name>  <filter-class>filter.FirstFilter</filter-class>  </filter>  <filter-mapping>  <filter-name>FirstFilter</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping> |