# Java spring mvc配置

## 安装eclipse

* 配置新增文件Window🡪perspective🡪customize perspective🡪short cuts
* 新增一个dynamic web project项目, 配置文件在WebContent目录下的WEB-INF目录下的web.xml
* 页面文件放在WebContent目录下
* 右键项目🡪build path🡪Configure build path, 修改Default out folder目录为TestWeb/WebContent/WEB-INF/classes

**项目布署**

* 进入apache-tomcat根目录下的conf文件夹下, 打开tomcat-users.xml文件, 在tomcat-users节点下添加如下内容<role rolename="manager-gui"/><user username="tomcat" password="tomcat" roles="manager-gui"/>, 内容根据情况而定, 以便可以通过web进入tomcat后台管理程序(<http://localhost:8080>), 点击manager app进入后面管理界面
* 在conf文件夹下添加Catalina/localhost/你的虚拟目录名.xml, 如Catalina/localhost/MyJspWeb.xml, 里面的内容添加<?xml version="1.0" encoding="utf-8"?><Context docBase="C:\\Users\\zengjunyi\\workspace\\TestWeb\\WebContent" reloadable="true" />, 注意Context节点名称大小写, 此时可以刷新tomcat后台管理程序, 可以看到程序已启动, <http://localhost:8080/MyJspWeb/MyJsp.jsp>, 注意jsp路径区分大小写

**其它配置**

1. 程序乱码

* 设置tomcat/conf/server.xml, Connector节点, 添加属性useBodyEncodingForURI=”true”
* 页面中使用request.setCharacterEncoding(“UTF-8”);

1. Jsp页面乱码

* Window 🡪 Preferences 🡪 General 🡪 Workspace, 最下面Text file encoding改为UTF-8
* Window 🡪 Preferences 🡪 Web 🡪 Jsp, 最下面Text file encoding改为UTF-8

1. 配置打开文件夹
2. Run 🡪 External Tools--External Tools Configurations, 双击program, 修改Name, 如打开文件夹, Location为C:/WINDOWS/explorer.exe, Arguemnts为${container\_loc}, 切换到common页签, 勾上External Tools.
3. Maven settings,

<?xml version="1.0" encoding="UTF-8"?>

<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0 http://maven.apache.org/xsd/settings-1.0.0.xsd">

<localRepository>F:\公司工作区\maven</localRepository>

<pluginGroups>

</pluginGroups>

<proxies>

</proxies>

<servers>

</servers>

<mirrors>

</mirrors>

<profiles>

</profiles>

</settings>

# 其它配置

1. 导出war包

* 右键项目选择export 🡪 War file, 选择运行时, 勾上Export source files

1. 添加全局乱码修复filter

<filter>

<filter-name>EncodingFilter</filter-name>

<filter-class>org.springframework.web.filter.CharacterEncodingFilter</filter-class>

<init-param>

<param-name>encoding</param-name>

<param-value>UTF-8</param-value>

</init-param>

<init-param>

<param-name>forceEncoding</param-name>

<param-value>true</param-value>

</init-param>

</filter>

<filter-mapping>

<filter-name>EncodingFilter</filter-name>

<url-pattern>/\*</url-pattern>

</filter-mapping>

1. Filter 过滤掉js问题, 在web.xml里添加如下的配置

<servlet-mapping>

<servlet-name>default</servlet-name>

<url-pattern>\*.css</url-pattern>

</servlet-mapping>

<servlet-mapping>

<servlet-name>default</servlet-name>

<url-pattern>\*.gif</url-pattern>

</servlet-mapping>

<servlet-mapping>

<servlet-name>default</servlet-name>

<url-pattern>\*.jpg</url-pattern>

</servlet-mapping>

<servlet-mapping>

<servlet-name>default</servlet-name>

<url-pattern>\*.js</url-pattern>

</servlet-mapping>

1. 修复返回json乱码的问题

<mvc:annotation-driven>

<mvc:message-converters register-defaults="true">

<bean class="org.springframework.http.converter.StringHttpMessageConverter">

<property name="supportedMediaTypes" value="application/json;charset=utf-8" />

</bean>

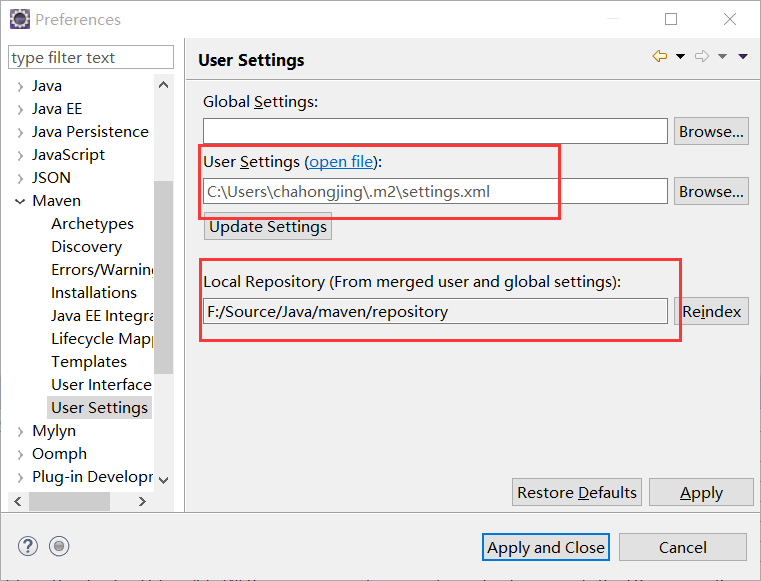
</mvc:message-converters>

</mvc:annotation-driven>

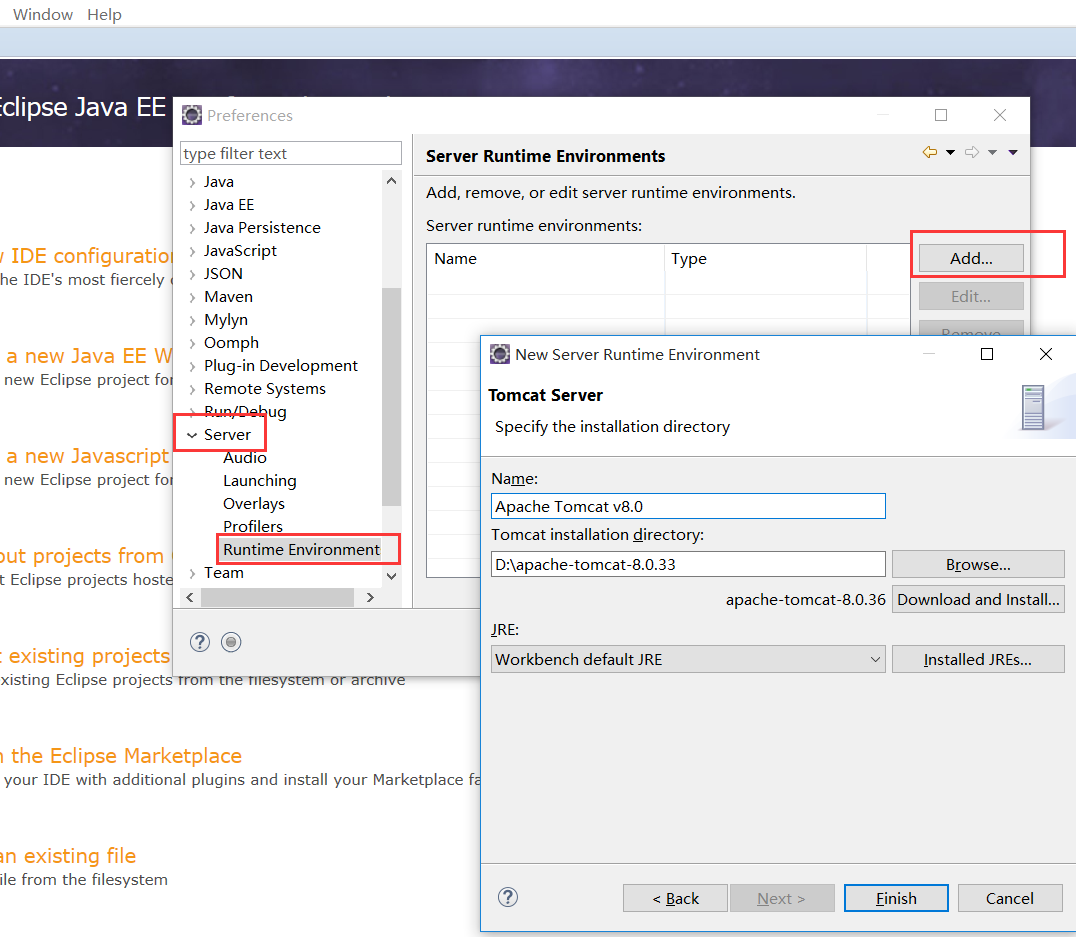
# Eclipse运行maven web项目

## 配置maven和tomcat

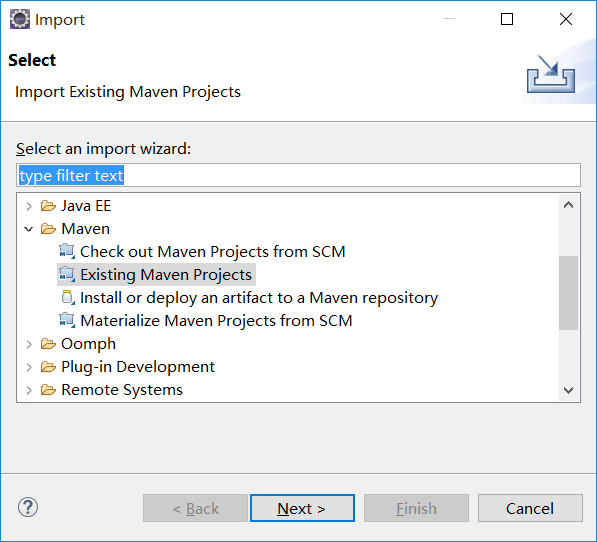
在window🡪preferences中找到maven项，设置maven仓库



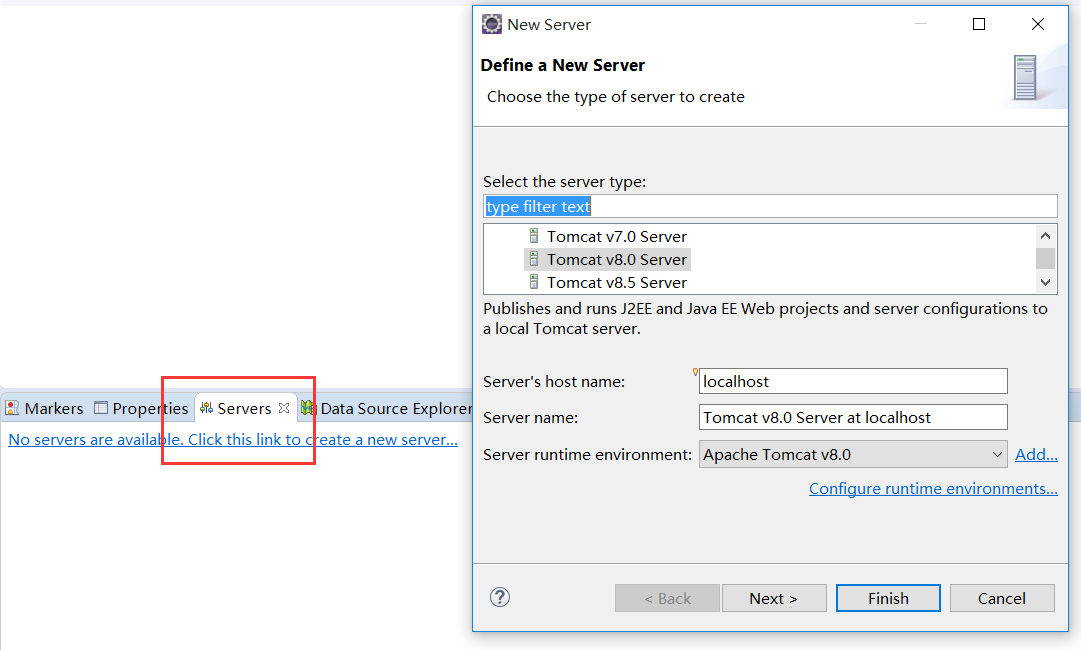
在找到server项中的runtime environment，添加tomcat服务器，如果后续服务器配置出现问题，可在这里删除重新配置。



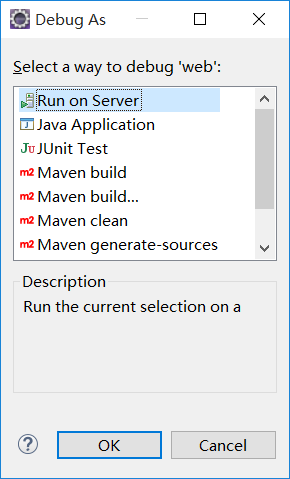
在File🡪Import导入项目，选择已存在的maven项目，确定后选择项目pom文件所在的目录（如果有多个pom文件，要全部勾选）



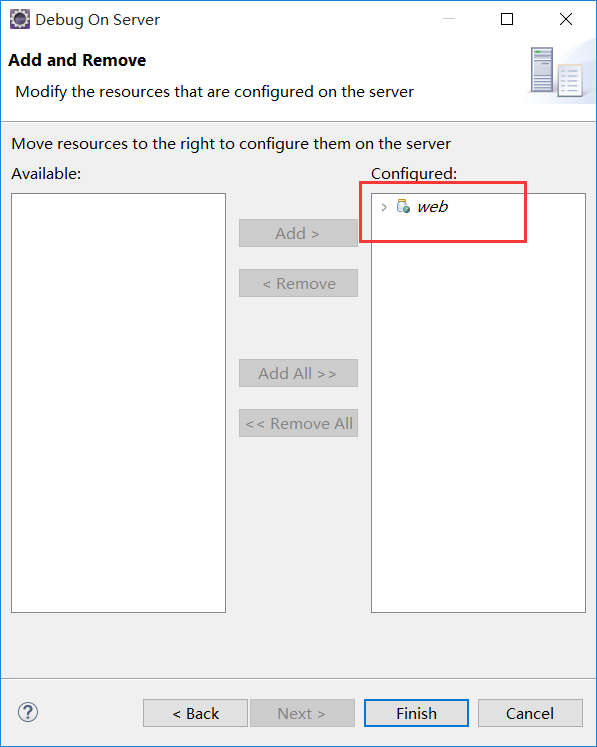
在server栏中添加服务器



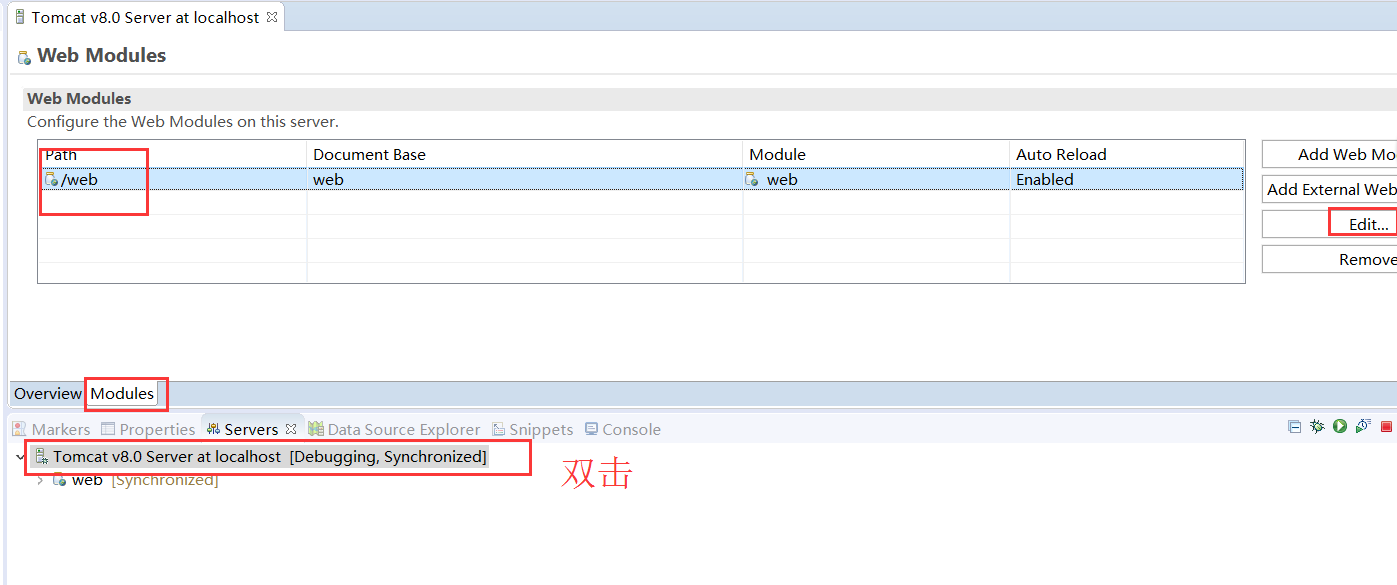
Debuge运行项目，选择run on server



在弹出的框选择next, 选择要运行的项目（移到右边）



双击server，个性web项目发布的context



如果选择的profile选择不正确，则可以在web项目右键🡪属性，在弹出框里找到maven，输入要使用的profile

