myeclipse+maven搭建web项目方法一

**一 前言**

用maven我们能做什么，有什么好处？

用maven，我们可以方便统一的进行jar包管理，jar包版本升级，快速项目构建以及分模块开发等相关功能。maven使用得好，项目开发速度就会成倍的提升。总之，项目越大，越能体现maven的优势。

**二 软件环境**

myeclipse8.6.1

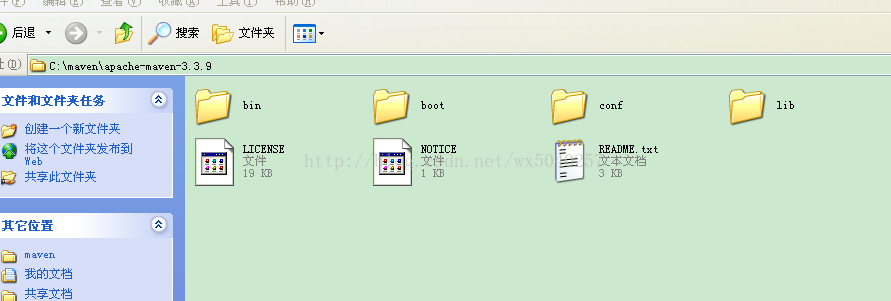
apache-maven-3.3.9

JDK1.7

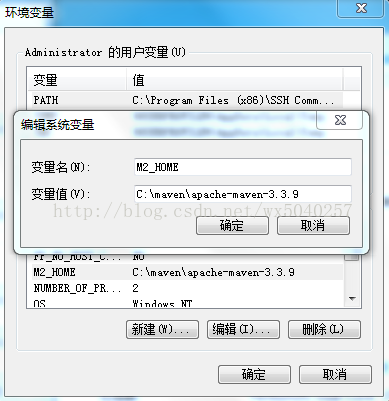
apache-tomcat-7.0.29。

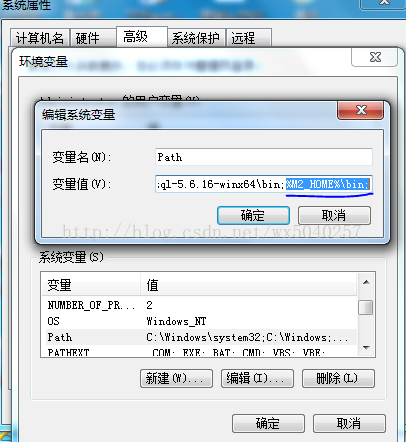
**三maven安装配置**

解压缩apache-maven-3.3.9-bin.tar.gz到本地，如图所示：



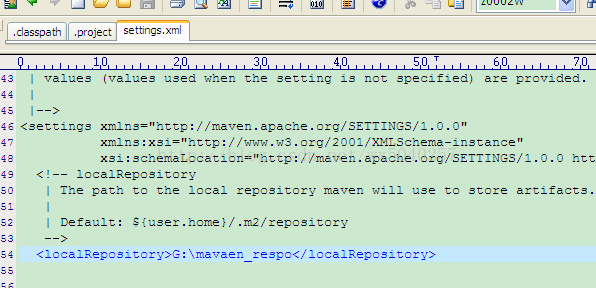
配置环境变量(如果你不想脱离开发环境运行项目，此处非必须)：



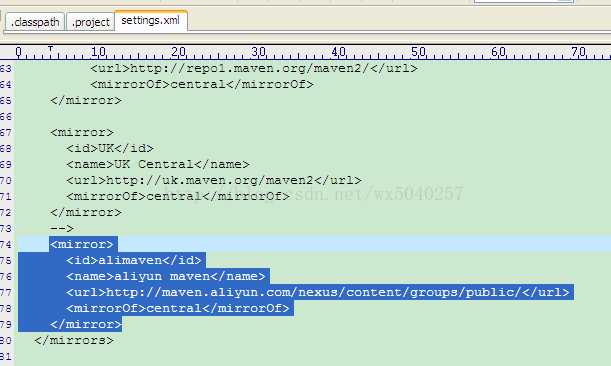


打开conf/settings.xml文件做如下配置

(1)配置本地仓库，如图

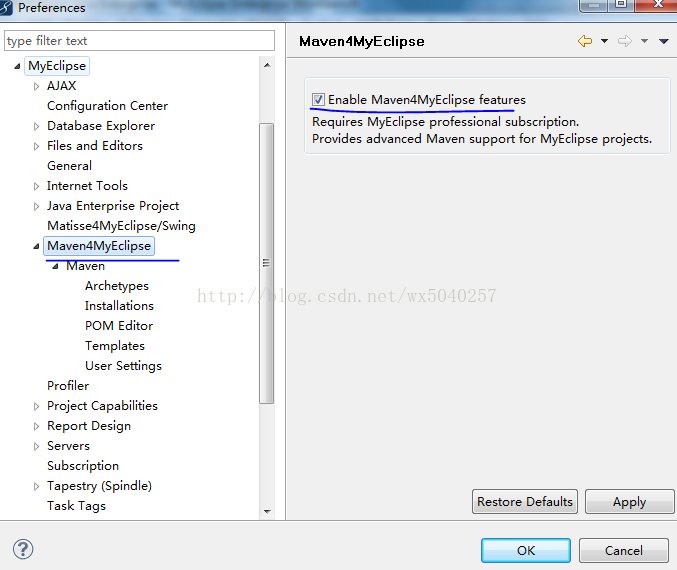


(2)配置中央仓库,即配置jar包的下载镜像(建议配置阿里巴巴的镜像，下载速度超快)，如图

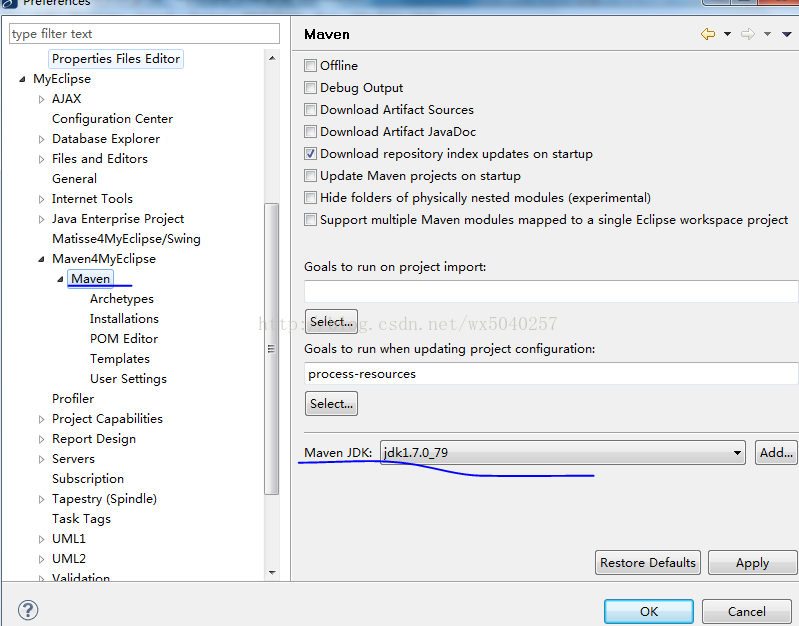


(3)打开myeclipse，设置maven相关参数

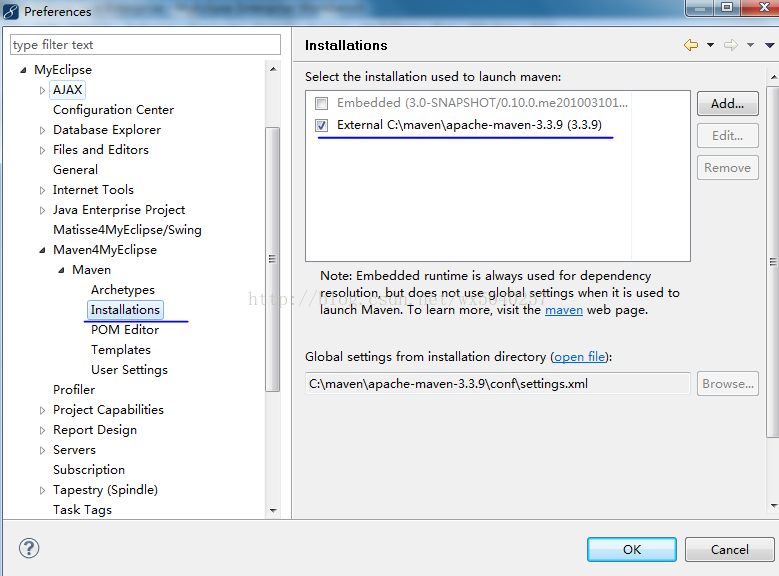
首先启用maven功能，如图所示:



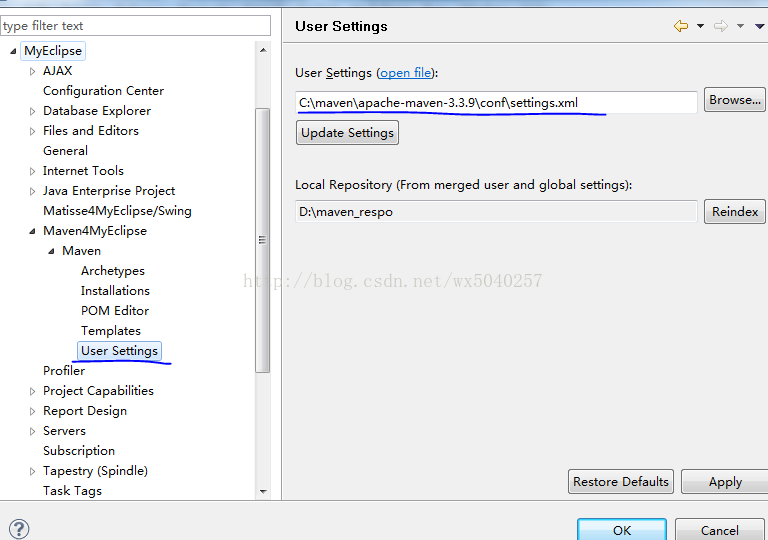
然后配置jdk：



然后配置Installations,点击add按钮，添加一个本地安装的maven，如图所示：



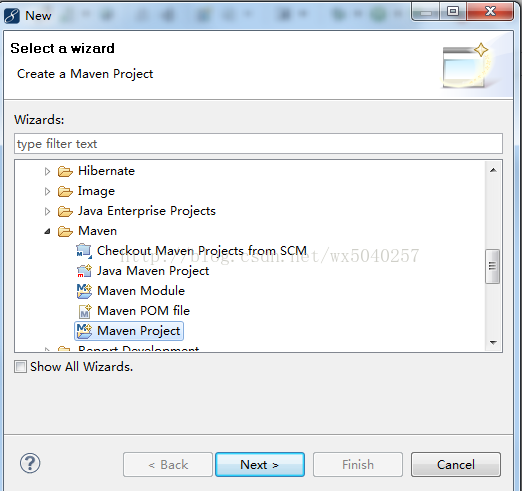
然后配置UserSettings:



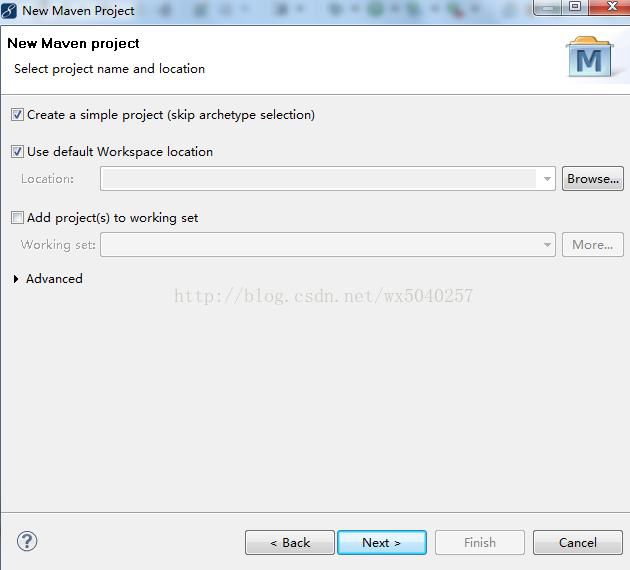
Ok,配置工作就此结束

**四 MyEclipse+Maven创建web项目**

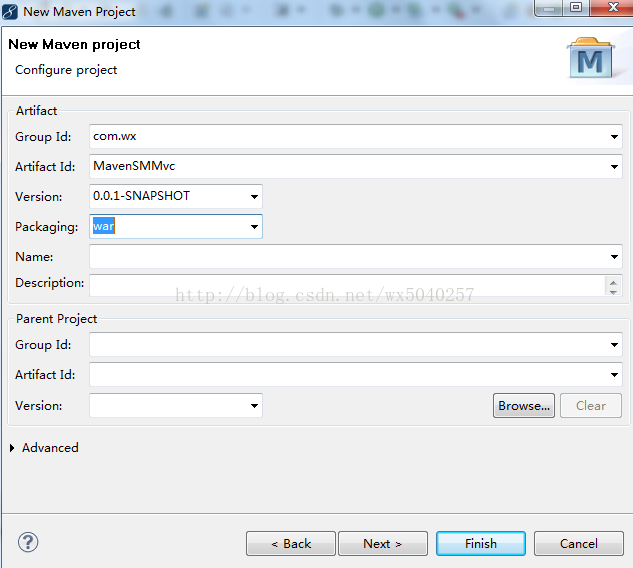
1、开启myeclipse，右键new——》other，如下图找到maven project



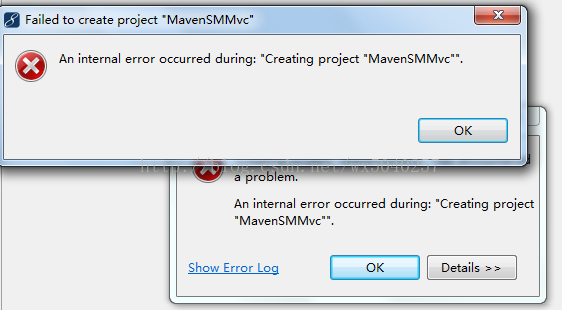
2、选择maven project，显示创建maven项目的窗口，勾选如图所示，Create a simple project括号后面的skip archetype selection表示跳过骨架，骨架表示一些创建模板，这里我们不使用模板更加简单。



3、输入maven项目的基本信息，如果是web工程，在Packaging栏我们要选择war，如下图所示：



4、点击完成会报错，如下所示，不过没有影响：



创建完成之后项目项目上显示的jdk还是1.4，我们可以在pom.xml里面配置jdk的升级插件，

# ( 更改jdk版本的方法见《[Maven maven-compiler-plugin 编译问题](http://www.cnblogs.com/zxguan/p/7847105.html)

》)

~~如下：~~

~~<build>~~

~~<plugins>~~

~~<plugin>~~

~~<groupId>org.apache.maven.plugins</groupId>~~

~~<artifactId>maven-compiler-plugin</artifactId>~~

~~<version>3.5.1</version>~~

~~<configuration>~~

~~<source>1.7</source>~~

~~<target>1.7</target>~~

~~<encoding>utf-8</encoding>~~

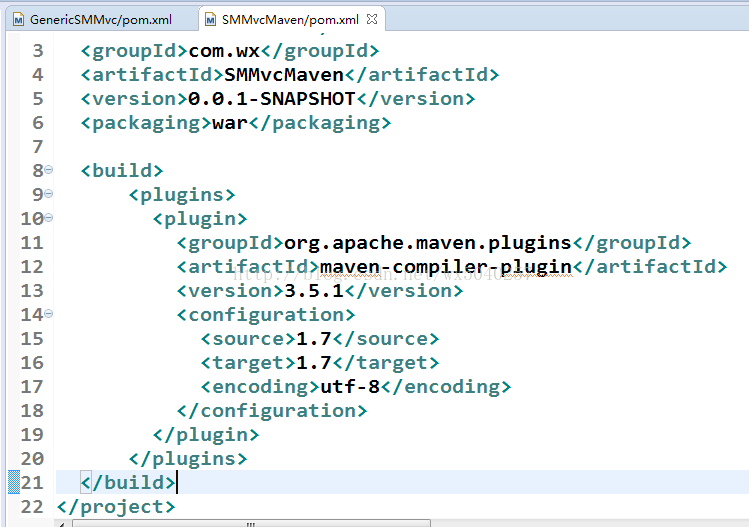
~~</configuration>~~

~~</plugin>~~

~~</plugins>~~

~~</build>~~

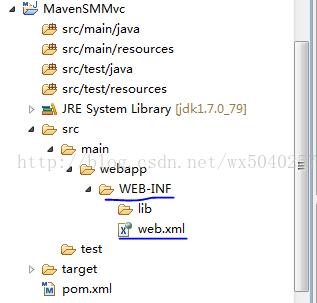
~~如图所示：~~

~~~~

~~我们配置完成后，需要执行一次更新项目配置的动作。选中项目 --> 右键 -->Maven4MyEclipse --> Update Project Configuration。这时候，又会报错，不过也没影响。更新完成之后， jdk版本也更改过来了，如图所示：~~



5.这时候src/main/webapp目录下还没有web.xml,从其它web工程拷贝过来即可(直接拷贝web-inf文件夹)，如图：



6、选择在pom.xml中添加jar包依赖

例如添加spring-core,应该是

<dependency>

    <groupId>org.springframework</groupId>

    <artifactId>spring-core</artifactId>

    <version>3.2.2.RELEASE</version>

</dependency>

不知道写怎么办？ 百度搜索maven spring-core repository    然后选择旁边的版本号,如图所示：



点击3.2.2，就可以直接复制出来了



附录：spring+springmvc+mybatis整合开发包依赖配置

===========================================================

<dependencies>

    <!--spring3.2.2 start-->

    <dependency>

    <groupId>org.springframework</groupId>

    <artifactId>spring-core</artifactId>

    <version>3.2.2.RELEASE</version>

    </dependency>

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-beans</artifactId>

        <version>3.2.2.RELEASE</version>

    </dependency>

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-context</artifactId>

        <version>3.2.2.RELEASE</version>

    </dependency>

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-aop</artifactId>

        <version>3.2.2.RELEASE</version>

    </dependency>

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-expression</artifactId>

        <version>3.2.2.RELEASE</version>

    </dependency>

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-web</artifactId>

        <version>3.2.2.RELEASE</version>

    </dependency>

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-webmvc</artifactId>

        <version>3.2.2.RELEASE</version>

    </dependency>

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-tx</artifactId>

        <version>3.2.2.RELEASE</version>

    </dependency>

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-jdbc</artifactId>

        <version>3.2.2.RELEASE</version>

    </dependency>

    <dependency>

        <groupId>aopalliance</groupId>

        <artifactId>aopalliance</artifactId>

        <version>1.0</version>

    </dependency>

    <dependency>

        <groupId>org.aspectj</groupId>

        <artifactId>aspectjweaver</artifactId>

        <version>1.6.2</version>

    </dependency>

    <!-- spring3.2.2end-->

    <!-- 日志包start -->

    <dependency>

        <groupId>org.slf4j</groupId>

        <artifactId>slf4j-api</artifactId>

        <version>1.7.5</version>

    </dependency>

    <dependency>

        <groupId>org.slf4j</groupId>

        <artifactId>slf4j-log4j12</artifactId>

        <version>1.7.5</version>

    </dependency>

    <dependency>

        <groupId>log4j</groupId>

        <artifactId>log4j</artifactId>

        <version>1.2.17</version>

    </dependency>

    <dependency>

        <groupId>commons-logging</groupId>

        <artifactId>commons-logging</artifactId>

        <version>1.1.3</version>

    </dependency>

    <!-- 日志包end -->

    <!-- mybatis相关包 start -->

    <dependency>

        <groupId>org.mybatis</groupId>

        <artifactId>mybatis</artifactId>

        <version>3.2.2</version>

    </dependency>

    <dependency>

        <groupId>org.mybatis</groupId>

        <artifactId>mybatis-spring</artifactId>

        <version>1.2.0</version>

    </dependency>

    <dependency>

        <groupId>asm</groupId>

        <artifactId>asm</artifactId>

        <version>3.3.1</version>

    </dependency>

    <dependency>

        <groupId>cglib</groupId>

        <artifactId>cglib</artifactId>

        <version>2.2.2</version>

    </dependency>

    <!-- mybatis相关包 end -->

    <!-- mysql连接驱动包 -->

    <dependency>

        <groupId>mysql</groupId>

        <artifactId>mysql-connector-java</artifactId>

        <version>5.1.26</version>

    </dependency>

    <!-- jstl标签库 -->

    <dependency>

        <groupId>jstl</groupId>

        <artifactId>jstl</artifactId>

        <version>1.2</version>

    </dependency>

    <!-- 字节码解析 -->

    <dependency>

        <groupId>org.javassist</groupId>

        <artifactId>javassist</artifactId>

        <version>3.17.1-GA</version>

    </dependency>

    <!-- 阿里巴巴json解析包 -->

    <dependency>

        <groupId>com.alibaba</groupId>

        <artifactId>fastjson</artifactId>

        <version>1.1.15</version>

    </dependency>

    <!-- 阿里巴巴druid连接池 -->

    <dependency>

        <groupId>com.alibaba</groupId>

        <artifactId>druid</artifactId>

        <version>1.0.9</version>

    </dependency>

    <!-- jsp and servlet start -->

    <dependency>

       <groupId>javax.servlet</groupId>

        <artifactId>servlet-api</artifactId>

       <version>2.5</version>

       <scope>provided</scope>

    </dependency>

    <dependency>

       <groupId>javax.servlet.jsp</groupId>

       <artifactId>jsp-api</artifactId>

       <version>2.1</version>

       <scope>provided</scope>

    </dependency>

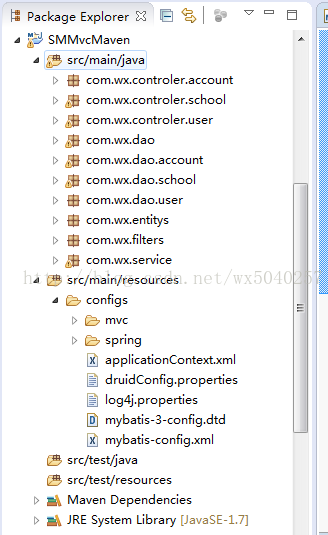
    <!--jsp and servlet end -->

  </dependencies>

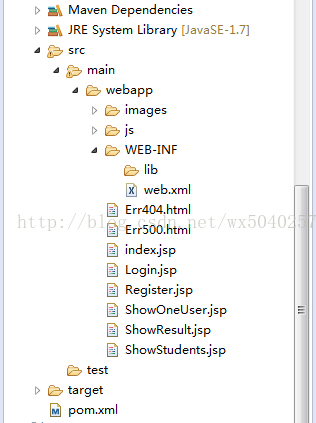
==============================================================

完成如上配置后，maven webapp项目就配置好了。

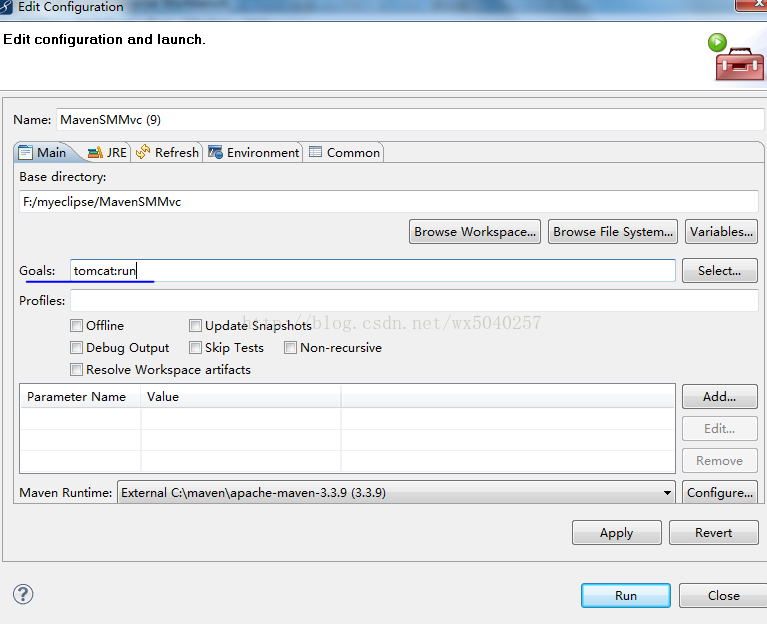
7.编写源代码及配置文件，完成后结构如图所示：



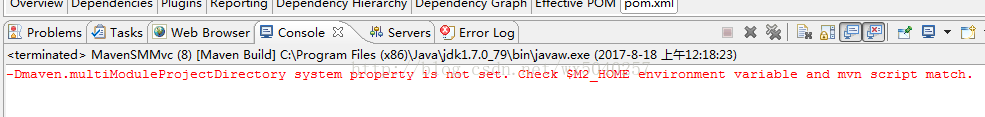
页面代码在下面：



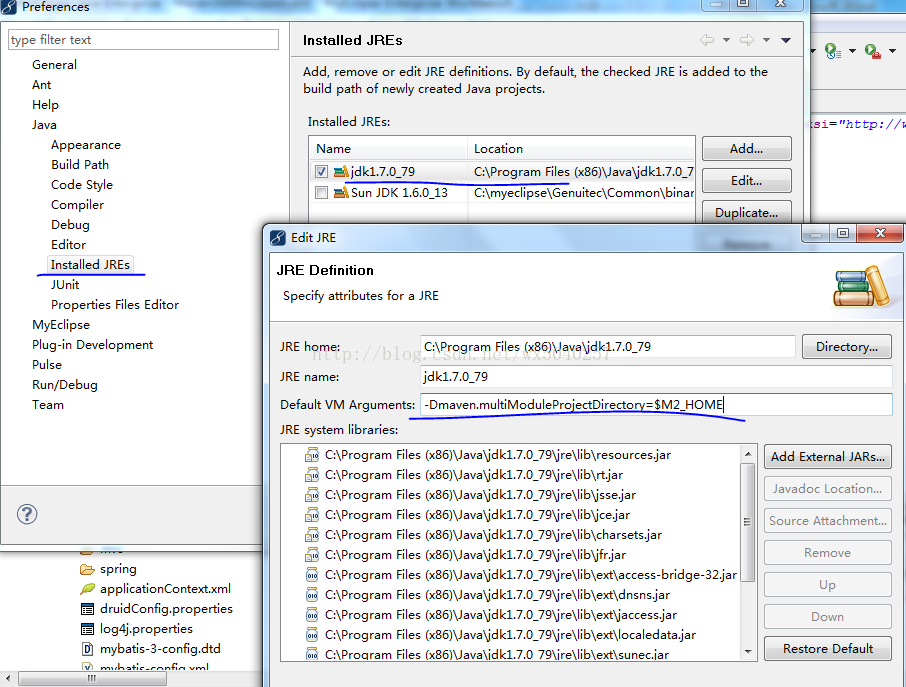
大功告成，接下来就可以运行了。在项目上点击右键，选择Run as--->Maven build…



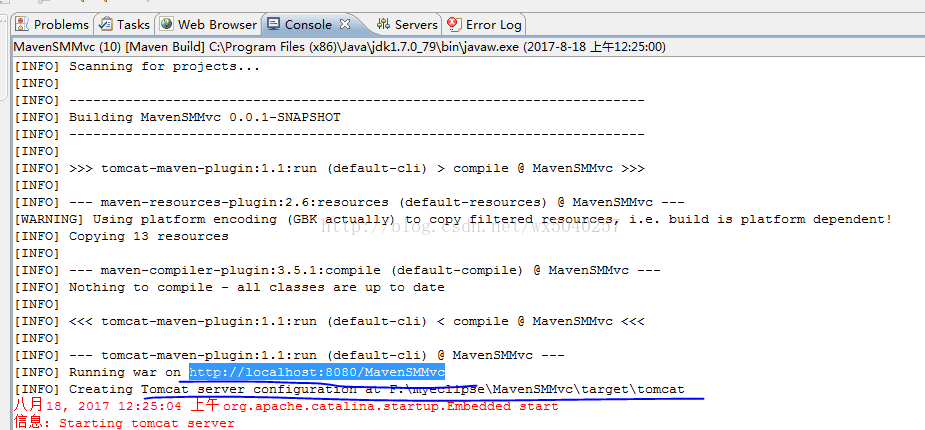
在Goals栏里面输入tomcat:run命令，单击Run按钮，报错如下：



按照提示添加环境变量$M2\_HOME,前面的步骤已经添加过，在此不再祥述,并且编辑我们的JDK配置，如图所示：



重新运行，大功告成，如图所示：



版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/wx5040257/article/details/77390655