

-----git:jiaofu

一：JDK 的安装

1.下载

进入 **Oracle** 官网，www.oracle.com 然后鼠标放在 Downloads->Java for Developers 点击下载相应的版本

2：安装，我一般选默认

3：环境配置

3.1 比如：通过计算机→属性→高级系统设置→高级→环境变量→进入配置界面

系统变量→新建 JAVA_HOME 变量

输入jdk的路径 D:\Java\jdk1.8.0_101

3.2 系统变量→编辑Path变量

在变量值最开始处输入%JAVA_HOME%\bin;%JAVA_HOME%\jre\bin; (win10 没有这样复杂，不过原理差不多)

3.3 系统变量→新建 CLASSPATH 变量

变量值填写 .;%JAVA_HOME%\lib;%JAVA_HOME%\lib\tools.jar (注意最前面有一点)

4：检验

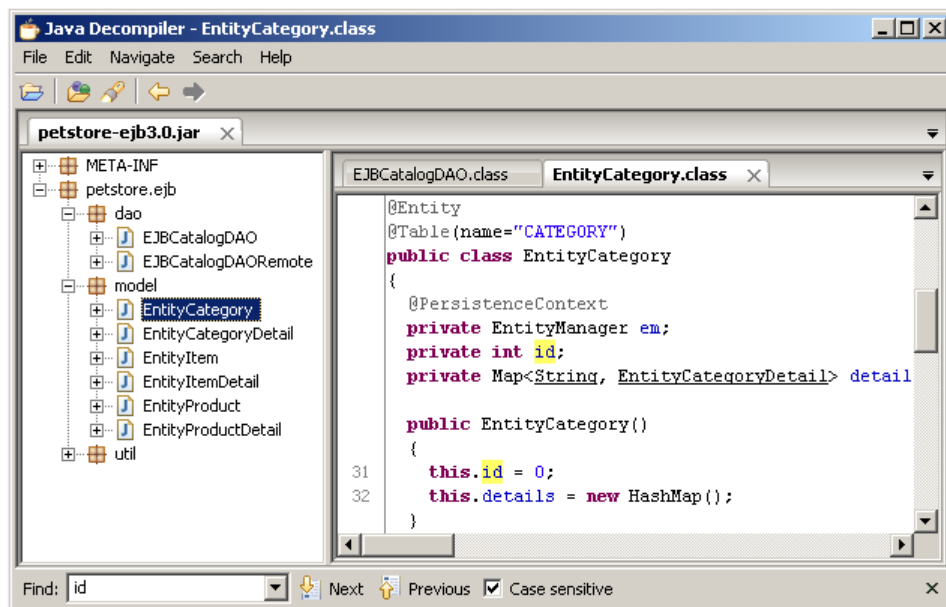
Win+R (Win7系统下) 输入java -version 回车，输出版本

信息就是可以了，然后输入 javac 回车 输出如下命令就对了

二：反编译工具

有很多，我一半用的是**JD-GUI**。

JD-GUI 是一个用 C++ 开发的 Java 反编译工具，由 Pavel Kouznetsov 开发，支持 Windows、Linux 和苹果 Mac OS 三个平台。而且提供了 Eclipse 平台下的插件 **JD-Eclipse**。JD-GUI 不需要安装，直接点击运行，可以反编译 jar, class 文件。



官方网站：<http://jd.benow.ca/>

二.Eclipse 的安装 和 Notepad++

下载链接：<http://www.eclipse.org/downloads/>

1:自动补全

最简单的修改方式是：Windows——>Preferences——>Java-->Editor-->Content Assist, 在Auto activation triggers for Java后面的文本框里只有一个“.”。现在你将其改为“.”abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ“即可。然后你再试试，会发现，现在的补全功能跟VS差不多了。你还可以在Advanced和Favorite里进行高级的设置。

2：字体设置

打开eclipse,找到window-->》点击后在下拉菜单中找到preferences-->》在左边的菜单栏中找到general, appearance在general展开后的第一行，点击展开appearance-->》找到colors and fonts,点击这样就快找到我们要的了-->》看到右边字体一列有很多的选项，我们只需要修改basic里面的-->》点击展开Basic,最下面有 Text Font 这个选项，点击-->》然后右边的Edit被激活了，点击Edit

直接安装

三.Maven 的安装

1：下载地址：<http://maven.apache.org/download.html>，打开后找到下载链接，如图：

Files

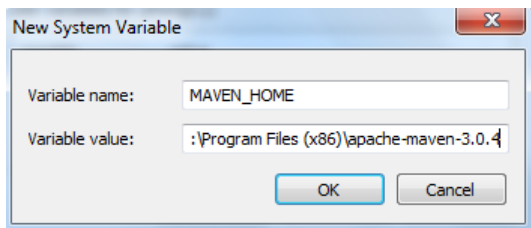
Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the [installation instructions](#). Use a source archive if you ir Maven yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to [verify the signature](#) of the release bundles against the public [KEYS](#) used by the Apache Mave

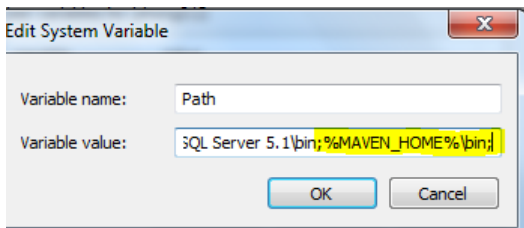
	Link	Checksum	Signature
Binary tar.gz archive	apache-maven-3.3.9-bin.tar.gz	apache-maven-3.3.9-bin.tar.gz.md5	apache-maven-3.3.9-bin.tar.gz.asc
Binary zip archive	apache-maven-3.3.9-bin.zip	apache-maven-3.3.9-bin.zip.md5	apache-maven-3.3.9-bin.zip.asc
Source tar.gz archive	apache-maven-3.3.9-src.tar.gz	apache-maven-3.3.9-src.tar.gz.md5	apache-maven-3.3.9-src.tar.gz.asc
Source zip archive	apache-maven-3.3.9-src.zip	apache-maven-3.3.9-src.zip.md5	apache-maven-3.3.9-src.zip.asc

- [Release Notes](#)
- [Reference Documentation](#)
- [Apache Maven Website As Documentation Archive](#)
- All sources (plugins, shared libraries,...) available at <https://www.apache.org/dist/maven/>
- Distributed under the [Apache License](#), version 2.0

2: 下载下来之后，解压，找个路径放进去，把bin的位置设在环境变量里，新建环境变量MAVEN_HOME



在PATH里加入maven的bin的路径



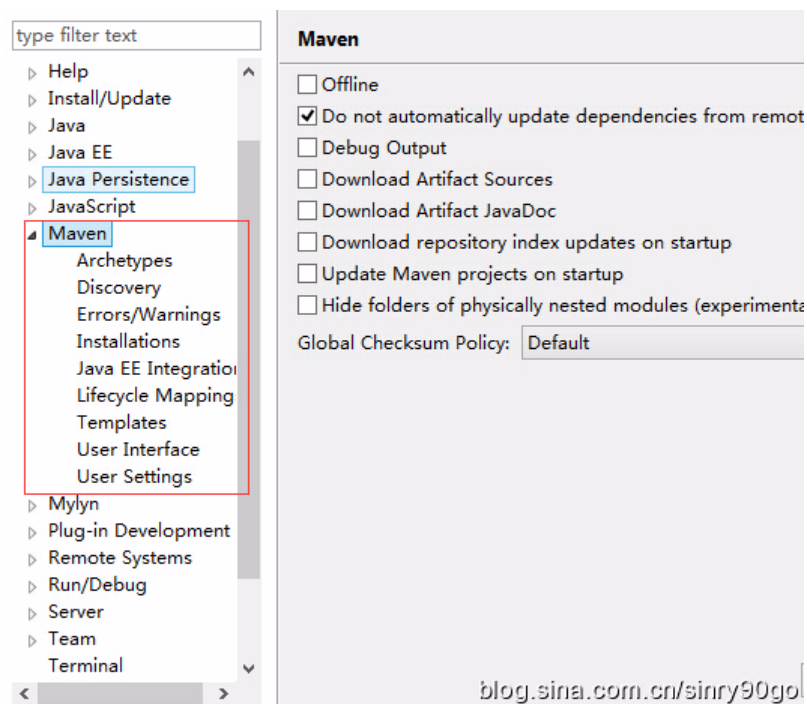
由于Maven依赖Java运行环境，因此使用Maven之前需要配置Java的运行环境。下载并安装JDK，配置JDK的环境变量JAVA_HOME，否则maven将无法使用

5 配置完毕后，在Windows命令提示符下，输入mvn -v测试一下，配置成功显示如图：

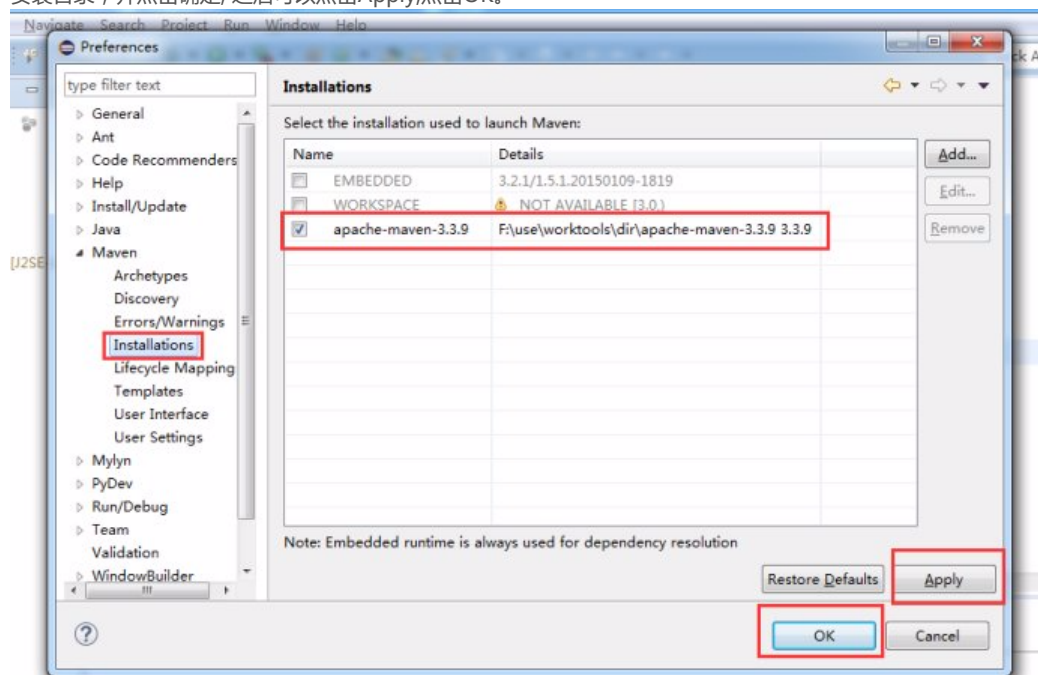
```
C:\Users\win8-11>mvn -v
Apache Maven 3.0.4 (r1232337; 2012-01-17 16:44:56+0800)
Maven home: C:\Program Files (x86)\maven\apache-maven-3.0.4\bin\..
Java version: 1.6.0_26, vendor: Sun Microsystems Inc.
Java home: C:\Program Files (x86)\Java\jdk1.6.0_26\jre
Default locale: zh_CN, platform encoding: GBK
OS name: "windows nt (unknown)", version: "6.2", arch: "x86", family: "windows"
C:\Users\win8-11>
```

3、配置maven安装目录

依次打开Window -> Preferences -> Maven，展开Maven的配置界面，如上图；



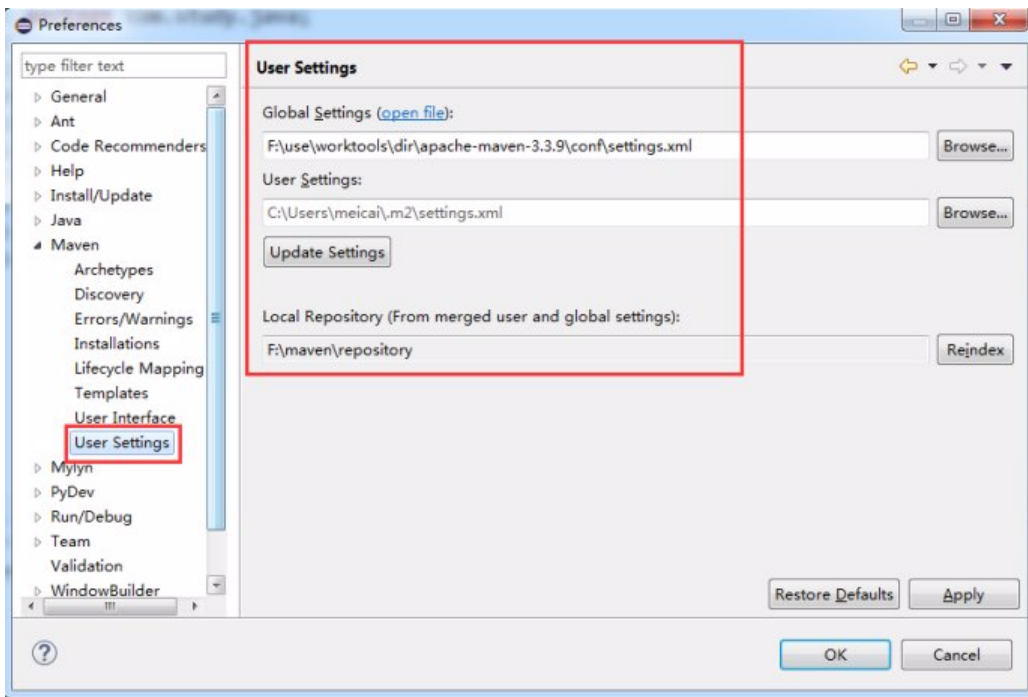
2、然后点击Installations -> add 选择maven安装目录，这里我的Maven安装目录为F:\use\worktools\dir\apache-maven-3.3.9，选择你的Maven安装目录，并点击确定，之后可以点击Apply,点击OK。



3、设置User Settings。

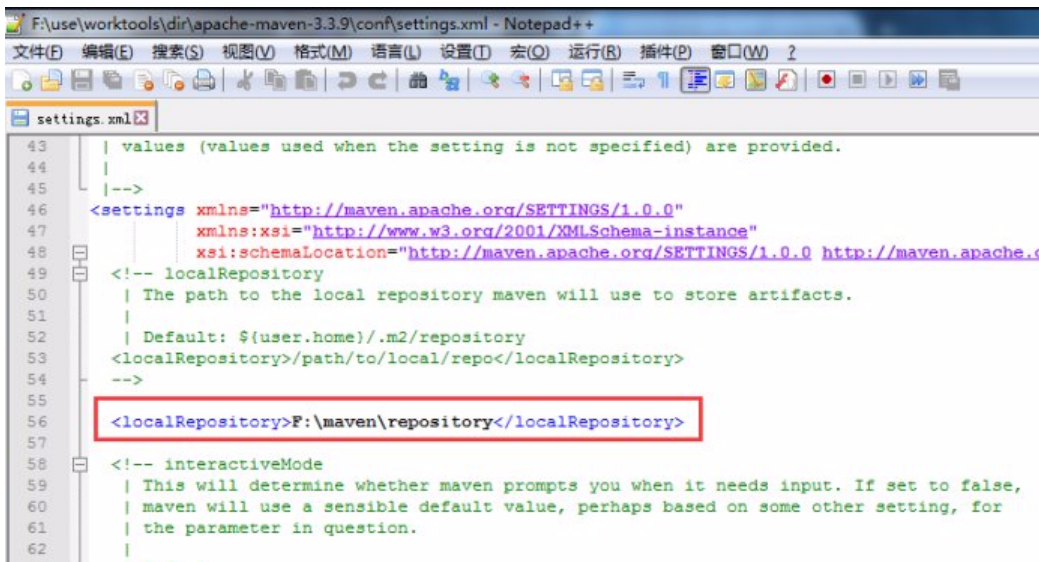
Global Settings选择maven 安装目录下conf文件夹下的settings.xml文件

注意：打开settings.xml文件检查Local Repository 项，如果为F:/maven/repository则配置成功。详细查看下一个步骤。



4、maven本地仓库配置

- 1) 打开F:\use\worktools\dir\apache-maven-3.3.9\conf\setting.xml文件，检查。。。项
- 2) 修改，这一行代表了Maven的Jar包仓库的位置。repository下放的都是项目所用到的jar包，每个Project都可以共用个repository。repository文件夹的名字可以随便取，但是一定要确保设置的仓库地址是有效的地址

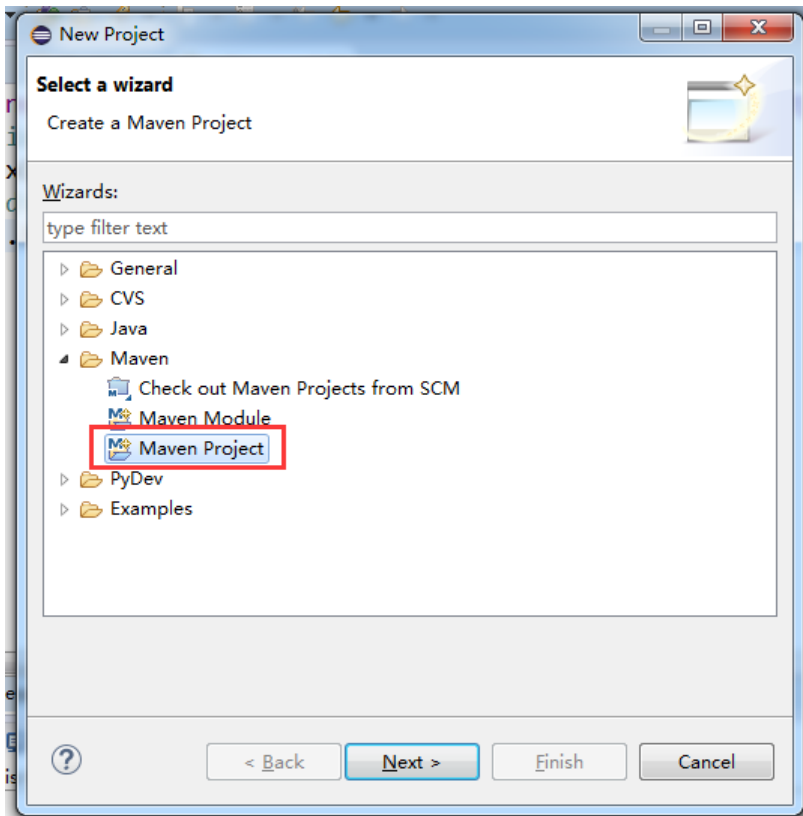


4、创建maven项目

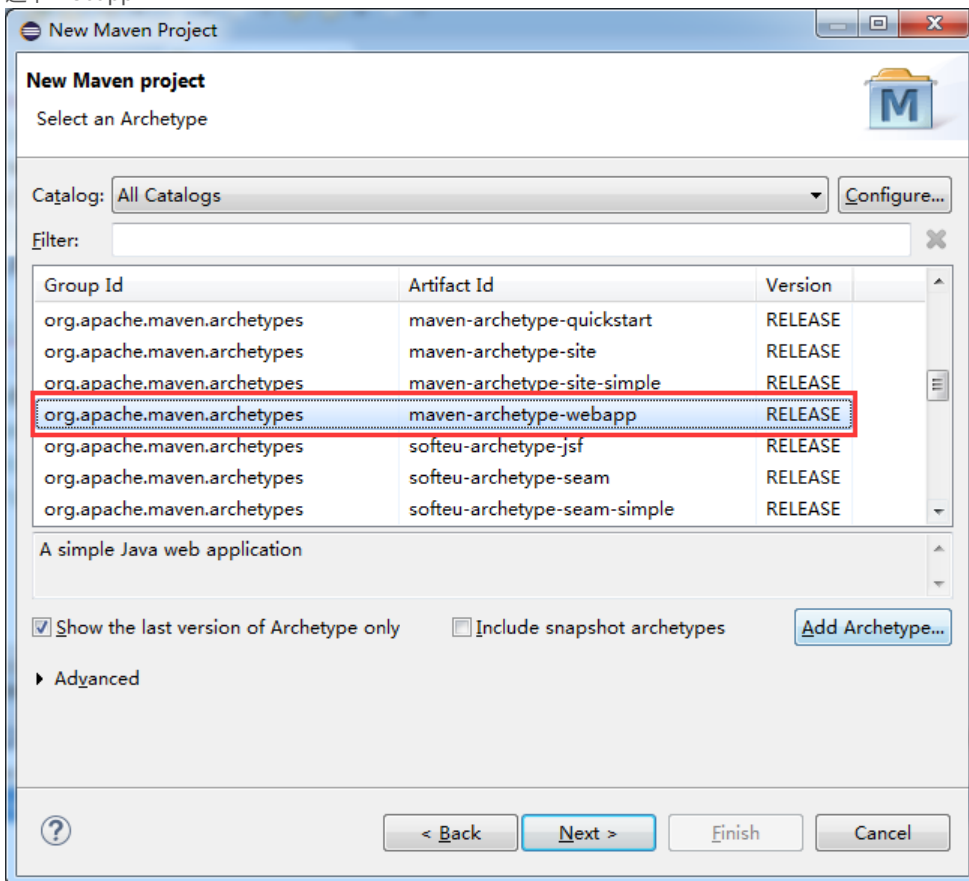
通过刚刚接触maven,我发现其主要目的就是将我们平时需要导入的jar文件啊，或者其他依赖信息啊，通过pom文件，自动去中央库加载到项目库中，避免我们手动去添加。

1、创建maven项目

File->New->Other->Maven，选择Maven Project：



选中 webapp



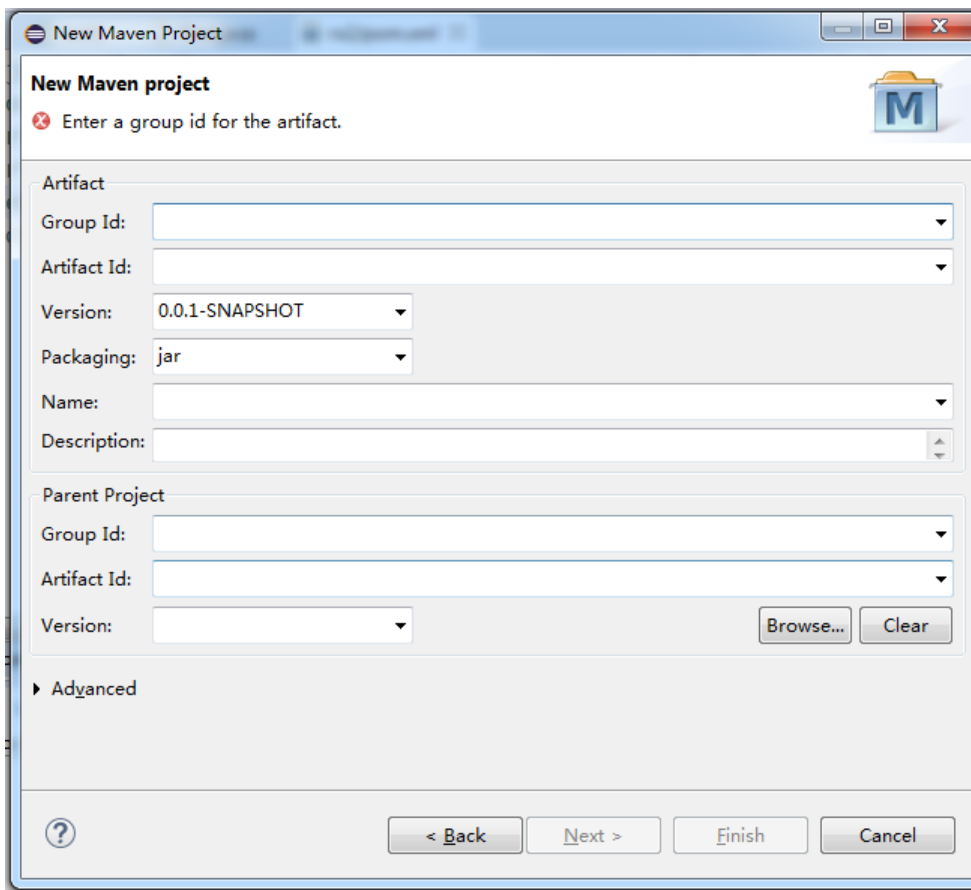
填写Group id和Artifact id，Version默认，Packaging默认为jar,Name，Description选填，其他的不填

groupId : group 翻译成中文就是组、集团的意识，groupId就是这样一个组或集团的ID标识，就拿 Apache 来说好了，Apache 大家都不陌生吧，Apache 官网上有非常多的项目，每个项目里面肯定能找到 org.apache.xxxx，而 org.apache 就是这里说groupId

artifactId : 还是拿 Apache 来说，Apache 官网上有非常多的项目，而每个项目的名字就是这里的 artifactId，换句话说，artifactId 就是你的项目名

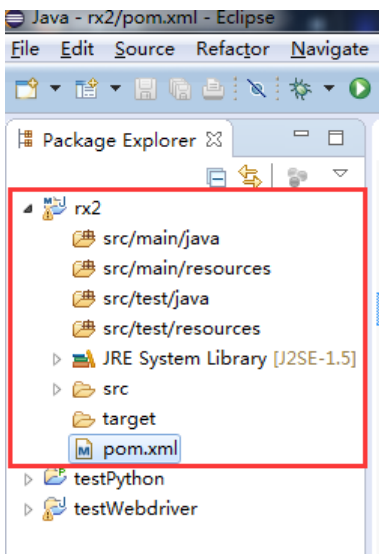
version : 这个就没有什么好解释的了吧，是项目版本的意思

package : 这个也没什么好说的，就是 groupId.artifactId，创建的时候会自动帮你补全，如果没有自动帮你补全就自己手动敲上去

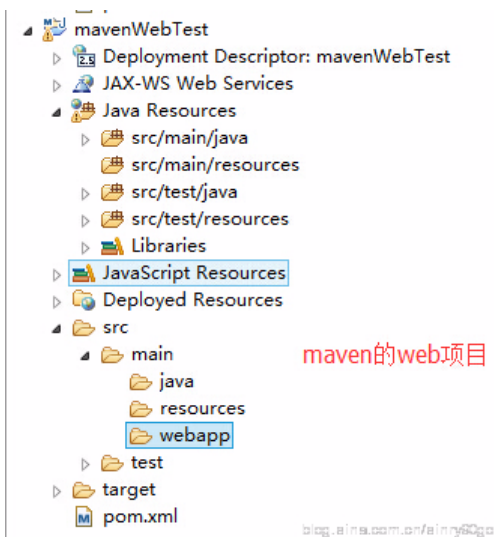


The image shows the 'New Maven Project' dialog box in Eclipse. It has a title bar with standard window controls. The main area is titled 'New Maven project' and contains a message: 'Enter a group id for the artifact.' with a red 'x' icon. Below this, there are two sections: 'Artifact' and 'Parent Project'. The 'Artifact' section has fields for 'Group Id', 'Artifact Id', 'Version' (set to '0.0.1-SNAPSHOT'), 'Packaging' (set to 'jar'), 'Name', and 'Description'. The 'Parent Project' section has fields for 'Group Id', 'Artifact Id', and 'Version', along with 'Browse...' and 'Clear' buttons. At the bottom, there is an 'Advanced' section (collapsed) and a row of buttons: '?', '< Back', 'Next >', 'Finish', and 'Cancel'.

项目的目录结构：



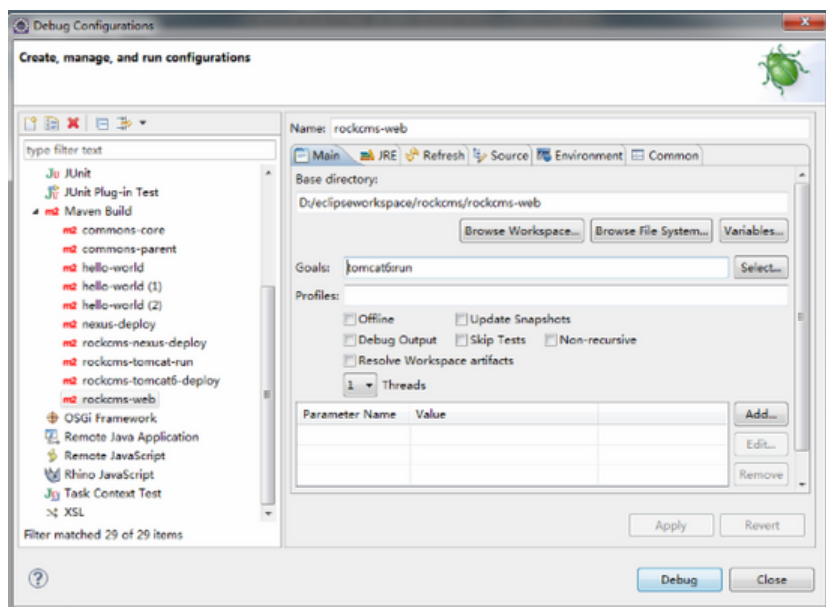
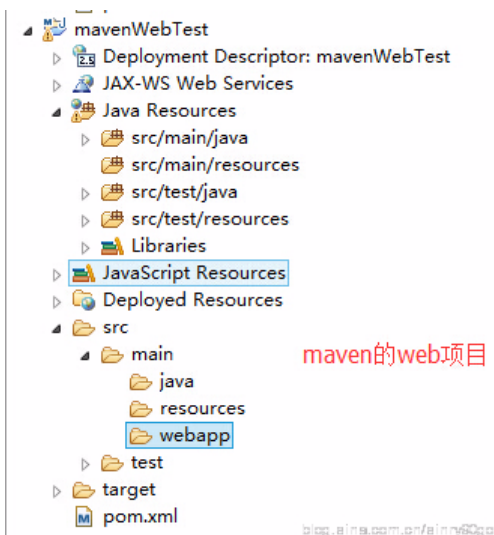
前三步就可以创建一个简单的maven项目，如果我们想创建一个Maven的web项目，把第三步的Packaging的类型改为war，之后点击Finish即可，如图所示：



四、启动tomcat运行web项目

在eclipse中利用maven命令启动tomcat，运行web工程，配置方法如下：

点击【Run】-【Debug Configurations..】，如下图：



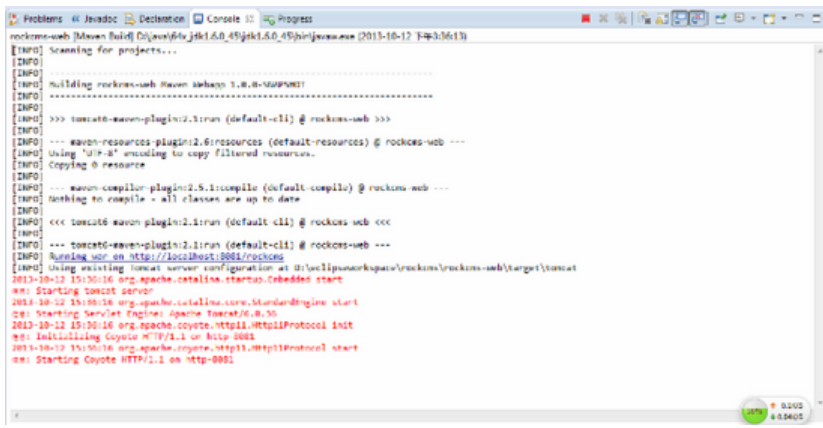
在“Maven Build”处添加一个调试命令，具体如下：

name：调试命令名，可以是项目名或其他

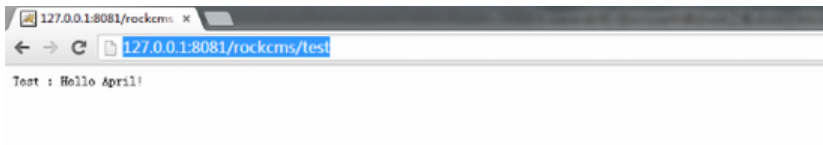
Base direcorey：运行的项目位置，可以通过【Browser Workspace..】,选择当前工作空间里的项目，这里为rockcms-web项目。

Goals：执行的命令，tomcat6:run，这里是简化后的命令名，这也是为什么要配置server的原因。

点击【Debug】按钮，Eclipse的console窗口中会显示启动信息。



第一次启动时，maven会下载tomcat插件，可能需要等待一会儿，后续再启动则不会在下载了，会比较快。启动成功后，访问 “<http://127.0.0.1:8081/rockcms/>”，则会显示我们修改的helloworld界面。



以上内容为在eclipse里建立maven的web工程，并且实现了在eclipse中利用tomcat运行web工程，配置起来步骤比较多，但是环境一旦搭建完成，后续构建、发布等工作量就会大大减少。

来源：http://blog.sina.com.cn/s/blog_7045bf3a0102wzi.html

四.git的安装

1、注册github账号

github官方网站：<https://github.com/>，注册过程：略

2、安装git客户端与tortoisegit

git客户端下载地址：<https://www.git-scm.com/>

tortoisegit下载地址：<https://tortoisegit.org/>

双击下载的安装包，默认安装直到完成。

3、配置git客户端

git客户端需要配置ssh建立与github之间的远程安全连接，安装完git客户端之后有如下快捷方式

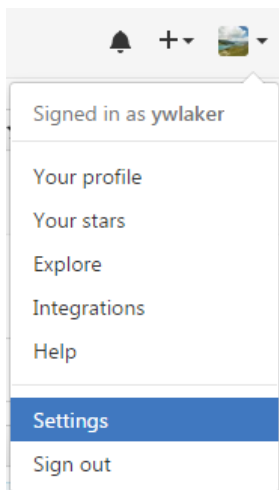


打开git bash，输入以下内容，邮箱为github注册邮箱

```
1 | ssh-keygen -t rsa -C "382243123@qq.com"
```


回车，输入“yes”，再回车，将在“C:\Users\zhuyili\.ssh\id_rsa.pub”文件生成ssh密钥，复制所有内容

登录github，打开“Settings”



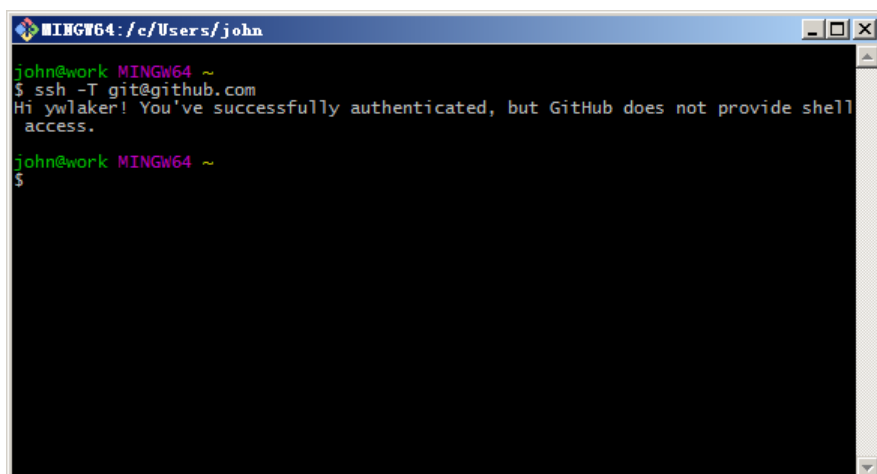
点击左侧“SSH and GPG keys”菜单，然后点击右侧上方“New SSH key”按钮，“Title”随意输入名称，“Key”处粘贴刚才复制的密钥，注意最后不要留空格或换行，保存配置

打开git bash，输入以下内容验证ssh连接

```
1 | ssh -T git@github.com
```

第一次需要输入“yes”然后回车才能看到连接成功的信息，以后就不需要了

如果是第一次的会提示是否continue，输入yes就会看到：You've successfully authenticated, but GitHub does not provide shell access。这就表示已成功连上github。



4、配置全局信息

配置全局的用户名、邮箱信息，作为提交人信息

```
1 git config --global user.name "jiaofu"
2 git config --global user.email "382243123@qq.com"
```

5、创建github仓库

登录github，创建一个工程，名为“test”，勾选“Initialize this repository with a README”，并选择一个license，以“Apache License 2.0”为例

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

Repository name

ywlaker

 /

test

Great repository names are short and memorable. Need inspiration? How about sturdy-giggle.

Description (optional)

test

☒ Public

Anyone can see this repository. You choose who can commit.

☐ Private

You choose who can see and commit to this repository.

☒ Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None

Add a license: Apache License 2.0

Create repository

创建完“test”仓库之后，查看该仓库，只包含两个文件：“README.md”和“LICENSE”

6、创建eclipse工程

在eclipse下创建工程，名为“test”（名称任意），在资源管理器中查看项目

名称	修改日期	类型	大小
.settings	2017/1/23 15:29	文件夹	
src	2017/1/23 15:29	文件夹	
target	2017/1/23 15:29	文件夹	
.classpath	2017/1/23 15:29	CLASSPATH 文件	1 KB
.project	2017/1/23 15:29	PROJECT 文件	1 KB
pom.xml	2017/1/23 15:29	XML 文档	1 KB

在git bash中查看项目（linux下点开头的文件为隐藏文件，所以只能看到三个文件（夹））

```
MINGW64: /d/eclipse/workspace/test
john@work MINGW64 ~
$ cd /d/eclipse/workspace/test/

john@work MINGW64 /d/eclipse/workspace/test
$ ll
total 4
-rw-r--r-- 1 john 197121 743 一月 23 15:29 pom.xml
drwxr-xr-x 1 john 197121 0 一月 23 15:29 src/
drwxr-xr-x 1 john 197121 0 一月 23 15:29 target/

john@work MINGW64 /d/eclipse/workspace/test
$ |
```

7、同步工程与github仓库

github的license与readme文件要同步到eclipse工程，eclipse工程的代码要同步到github仓库，并且eclipse自动生成的那些“.settings”、“”.project”、“”.classpath”还有“target”不需要同步到github仓库，怎么处理？

7.1、git初始化eclipse工程目录

在资源管理器中打开“test”工程根目录，右键“Git Create repository here...”



或者，git bash输入命令

```
1 | cd /d/eclipse/workspace/test/
2 | git init
```

创建新文件夹，打开，然后执行 git init以创建新的 git 仓库

```
MINGW64: /d/eclipse/workspace/test
john@work MINGW64 ~
$ cd /d/eclipse/workspace/test/

john@work MINGW64 /d/eclipse/workspace/test
$ ll
total 4
-rw-r--r-- 1 john 197121 743 一月 23 15:29 pom.xml
drwxr-xr-x 1 john 197121 0 一月 23 15:29 src/
drwxr-xr-x 1 john 197121 0 一月 23 15:29 target/

john@work MINGW64 /d/eclipse/workspace/test
$ git init
Initialized empty Git repository in D:/eclipse/workspace/test/.git/

john@work MINGW64 /d/eclipse/workspace/test (master)
$
```

7.2、同步github仓库文件到eclipse工程目录

git bash输入命令

```
1 git remote add origin git@github.com:ywlaker/test.git
2 git pull origin master
```

后面的ywlaker和test

表示你再github的用户名和刚才新建的仓库，加完之后进入.git，打开config，这里会多出一个remote "origin"内容，这就是刚才添加的远程地址，也可以直接修改config来配置远程地址。

7.3、添加.gitignore

将eclipse自动生成的与代码无关的文件排除在同步列表之外，git bash切换到工程根目录，输入

```
1 vi .gitignore
```

然后按下键盘的“i”键，输入

```
1 target/
2 .settings
3 .classpath
4 .project
```

按下“esc”键，再输入“:wq”保存退出，此时“test”工程根目录多了一个“.gitignore”文件

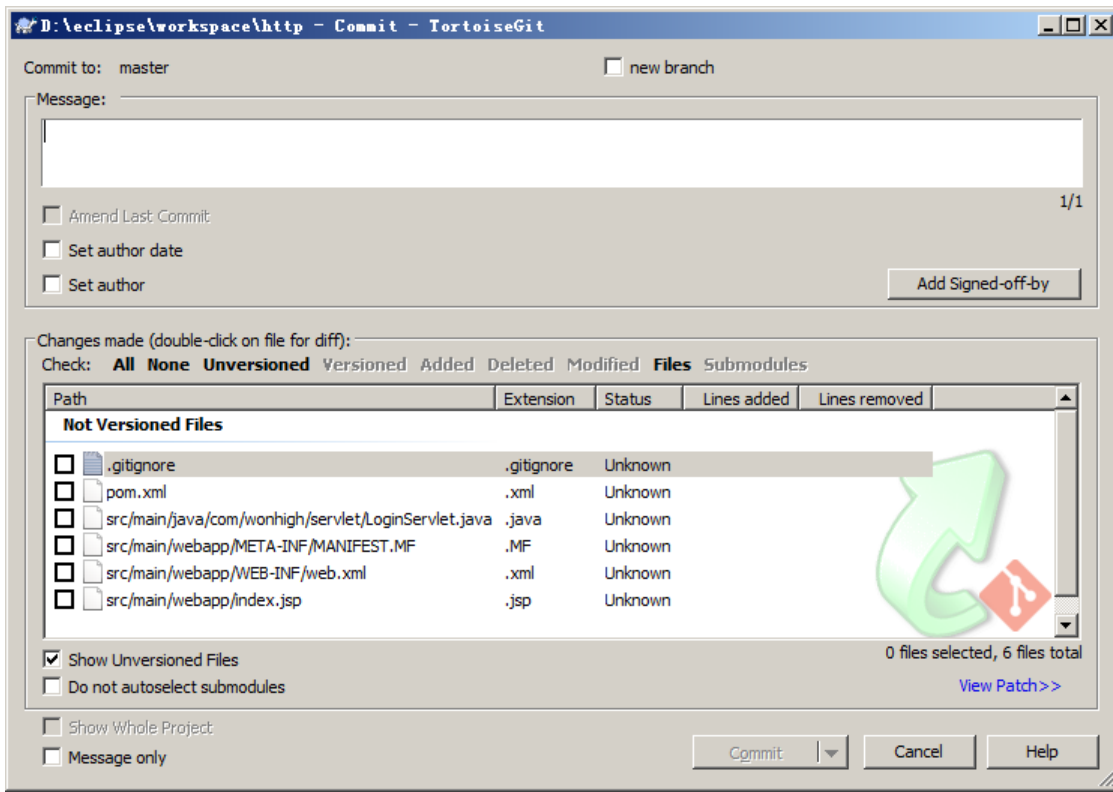
因为windows下无法新建以点号开头的文件，所以只能用上面这种方式。

7.4、同步eclipse工程到github仓库

在资源管理器中进入“test”工程根目录，右键“Git Commit -> 'master'”



出现和tortoisesvn一样的提交界面，确定内容后提交即可



这一步是将文件提交到本地git缓存，用命令行也可以，git bash切换至工程更目录，输入

```
1 git add ./*
2 git commit -m "comment"
```

这些是同步的是本地到 暂存区；第二个是 暂存区（Index），它像个缓存区域，临时保存你的改动

最后，将本地缓存同步到github仓库

```
1 git push -u origin master
```

参考来源: <http://www.cnblogs.com/ywlaker/p/6344334.html>

-----git:jiaofu