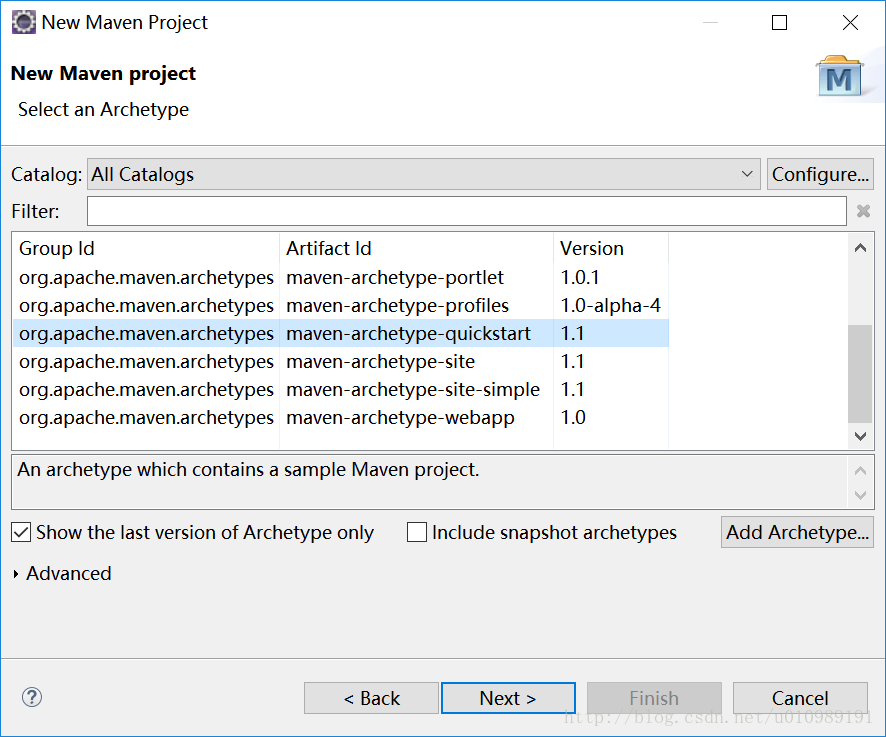
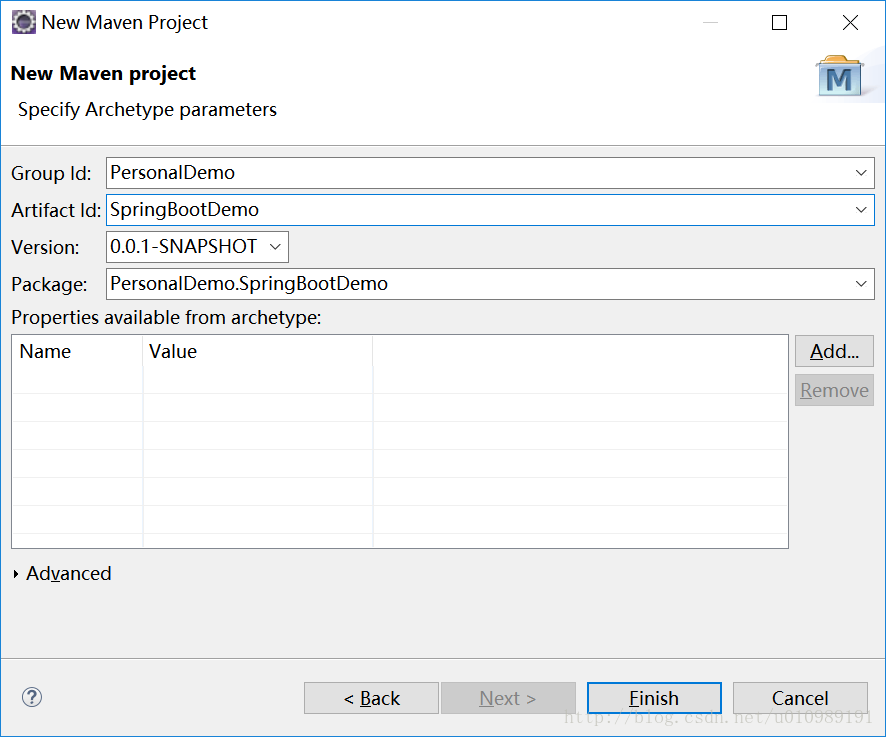
<http://blog.csdn.net/u010989191/article/details/71597860>

## 前期准备

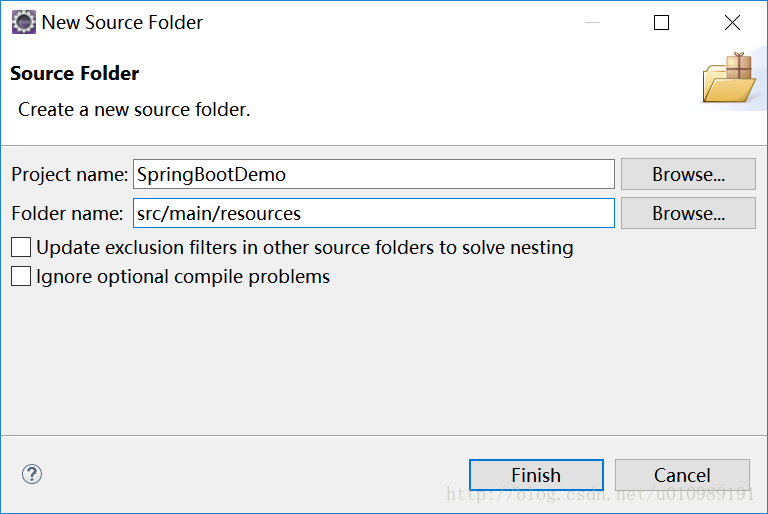
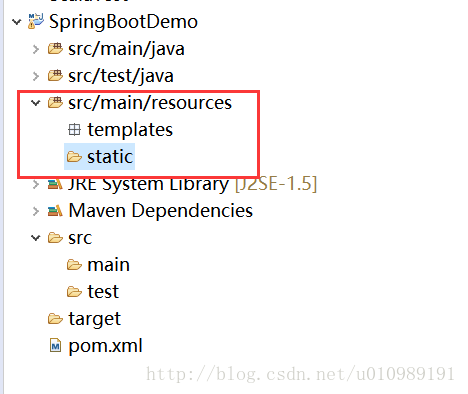
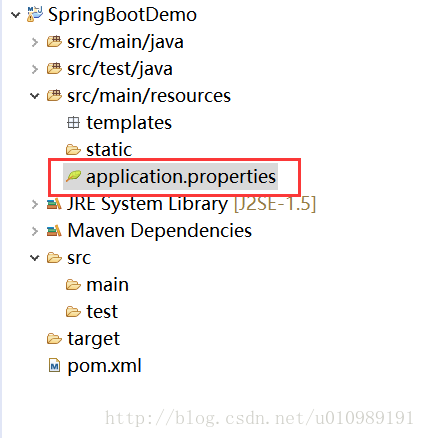
安装maven：下载->安装->配置环境变量。下载链接:<http://maven.apache.org/download.cgi>

安装eclipse：eclipse-jee-neon-3-win32-x86\_64 下载链接:<https://www.eclipse.org/downloads/>

## 创建maven项目

1.eclipse: file->new->maven project   
2.选择创建的版本。   
  
3.输入Group Id 和Artifact Id。   
Group Id代表的是公司组织名称 Artifact Id 表示的当前项目的名称   
  
4.点击Finish即可创建一个maven项目。

## 构建成Spring Boot的项目目录

1.在src/main/java源文件夹上右键，选择new->Source Folder。新建一个src/main/resources源代码文件夹。如下图所示：   
   
2.在src/main/resources源代码文件夹上右键，选择new->folder。新建以下几个文件夹，static和templates。   
   
3.在src/main/resources文件夹下创建application.properties文件。   
   
4.在pom.xml中加入依赖配置参数。

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>PersonalDemo</groupId>

<artifactId>SpringBootDemo</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>jar</packaging>

<name>SpringBootDemo</name>

<url>http://maven.apache.org</url>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

</properties>

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>1.5.3.RELEASE</version>

<relativePath />

</parent>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

5.创建启动类。SpringBootApplication.java

package PersonalDemo.SpringBootDemo;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.SpringBootConfiguration;

@SpringBootConfiguration

public class SpringBootApplication {

public static void main(String[] args) {

SpringApplication.run(SpringBootApplication.class, args);

}

}

6.创建测试TestController

package PersonalDemo.SpringBootDemo.controller;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

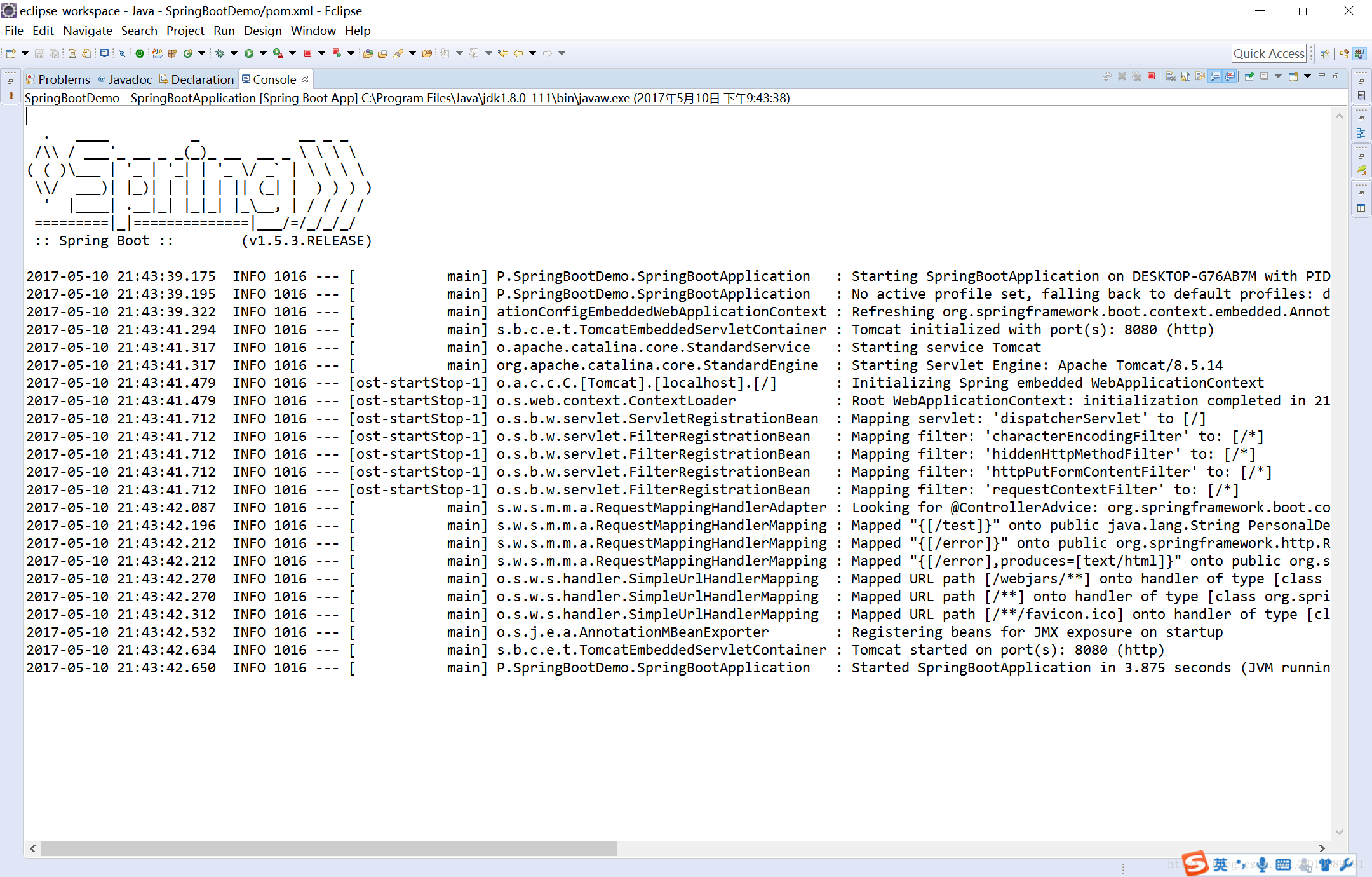
public class TestController {

@RequestMapping("test")

public String test(){

return "test";

}

}7.右键SpringBootApplication类，选择run as->Spring boot app。启动成功界面。   
  
8.测试请求/test。   


## 总结

以上即完成了一个Spring boot的简单环境搭建，可以结合自己的项目需求加入不同的服务。

如果需要整合mybatis或者redis等可以参考以下配置:<http://blog.csdn.net/u010989191/article/details/71330676>

版权声明：本文为博主原创文章，未经博主允许不得转载。