公司接下来某个项目打算使用**[微服务](http://lib.csdn.net/base/microservice" \o "微服务知识库" \t "http://blog.csdn.net/tuzongxun/article/details/_blank)**[**架构**](http://lib.csdn.net/base/architecture)，使用springcloud以及它集成的一些相关项目，因此虽然在其他方面的很多技术上还感觉急需提高，却又不得不以工作为重，先放下其他来了解一下这方面的技术。

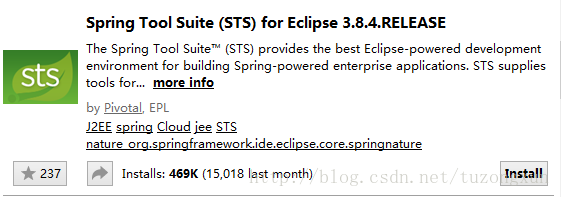
一番了解后发现，[**spring**](http://lib.csdn.net/base/javaee) cloud是建立在spring boot的基础上的，而之前虽然听说过，也随便看了一下spring boot，却没有真正使用，因此还必须先花时间学一下spring boot。

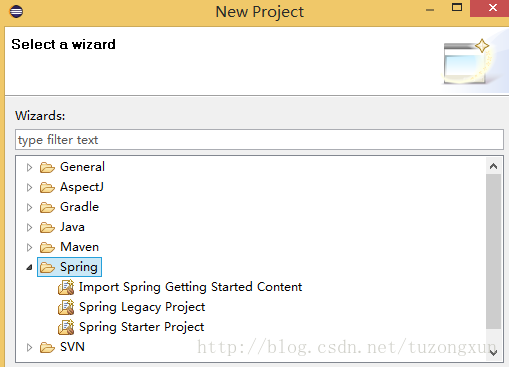
**spring boot的理念是“习惯优于配置”，我个人的理解就是尽量减少开发过程中手动的spring相关的配置文件。同时使用spring boot还有一个优点就是，它可以内嵌很多容器，例如tomcat，使得原本可能需要安装tomcat才能运行的web项目，可以直接以运行jar文件的形式启动运行。**

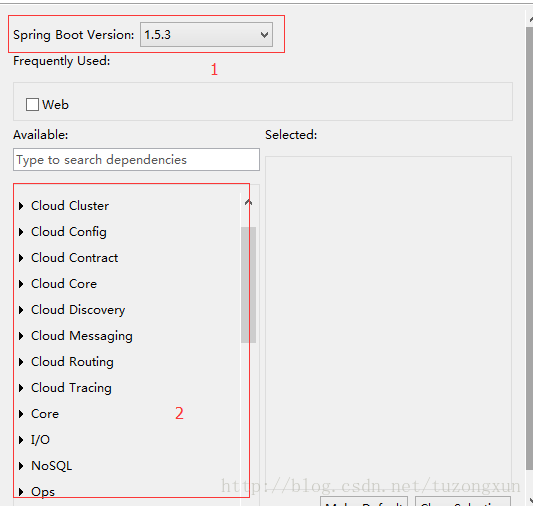
spring boot项目创建有多种方式，鉴于目前工作中使用的是eclipse开发工具，因此整个学习过程中，也都是在eclipse中进行。

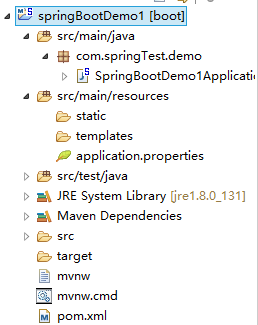
而eclipse中的创建实际上也是可以有两种方式的，一种是在安装了STS插件之后直接创建，另一种是创建简单的maven项目后，修改pom.xml文件，为了提高效率，我这里就安装了STS插件，以第一种方式创建。

这个过程中还有一个小插曲，我原本的eclipse版本是Mars.1 Release (4.5.1)，安装STS的时候安装不成功，说是eclipse版本不匹配，于是安装了新版的eclipse，版本号Neon.3 Release (4.6.3)。但是当我在新版的eclipse中安装好STS后，再来尝试在旧版安装时，居然又一路畅通无阻的成功了。

eclipse中STS安装也有几种方式，我的STS的安装过程是这样的： help –> Eclipse Marketplace –>Popular，然后选择下图中的插件install。   


这个插件安装成功以后，就可以看到在eclipse中new project时会有spring这个选项了（当然了，不知这一个地方有变化），打开之后还会有几个子选项，如图：   


而我快速创建spring boot项目的时候，使用的就是上图中第三个子选项Spring Starter Project。具体步骤是：new –>Project –>Spring Starter Project –>出现的界面中name选项后输入项目名称 –>接下来出现如下图所示界面：   
  
这里我主要是使用了两个地方，第一个就是选择spring boot version，第二个就是在标示2的位置选择要创建的具体spring boot项目，有很多选项可供选择，而我就选了一个web项目。

创建好的web项目基本结构如下图：   
  
创建的时候它会自动生成一个带有main方法的类，这个main方法实际上就是spring boot项目的程序入口，我在里边加入了一个@RestController和这样一段代码：

@RequestMapping("/")

String index(){

return "Hello Spring Boot";

}

之后整个类的代码如下：

package com.springTest.demo;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

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

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

@RestController

@SpringBootApplication

public class SpringBootDemo1Application {

@RequestMapping("/")

String index(){

return "Hello Spring Boot";

}

public static void main(String[] args) {

SpringApplication.run(SpringBootDemo1Application.class, args);

}

}

当安装好STS插件之后，创建一个简单的spring boot的web项目就是这么简单，不需要像传统的spring项目一样还要配置spring.xml等配置文件以及web.xml文件。

但是需要注意的是，我在第一次创建的时候，spring boot相关的jar包下载不下来，因为公司的maven仓库中没有对应版本的，于是自己修改了maven的配置文件，加入了阿里云的maven仓库：

<mirror>

<id>alimaven</id>

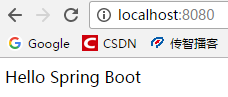
<name>aliyun maven</name>

<url>http://maven.aliyun.com/nexus/content/groups/public/</url>

<mirrorOf>central</mirrorOf>

</mirror>

当重新配置maven仓库，使得程序编译没有问题后，就可以启动项目了，eclipse中使用run as –>Spring Boot App就可以直接运行，不需要像传统web项目那样要加入到tomcat中才行。

启动成功后浏览器访问localhost:8080，会看到页面如下，一个简单的spring boot web项目就成功创建了。   


对于yml等没有提示的

<http://spring.io/tools/sts/all>

手动更新对应自己版本的

