1. 下载源码

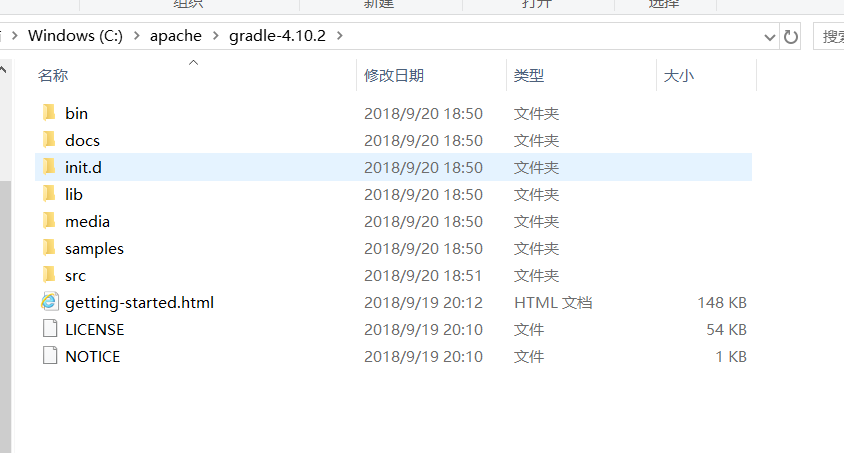
可以在github上下载 ，也可以下载我课堂的源码（有注释）

git clone https://github.com/spring-projects/spring-framework.git

1. 配置gradle 编译环境

Spring源码是通过gradle编译的，所以要先配置gradle编译环境

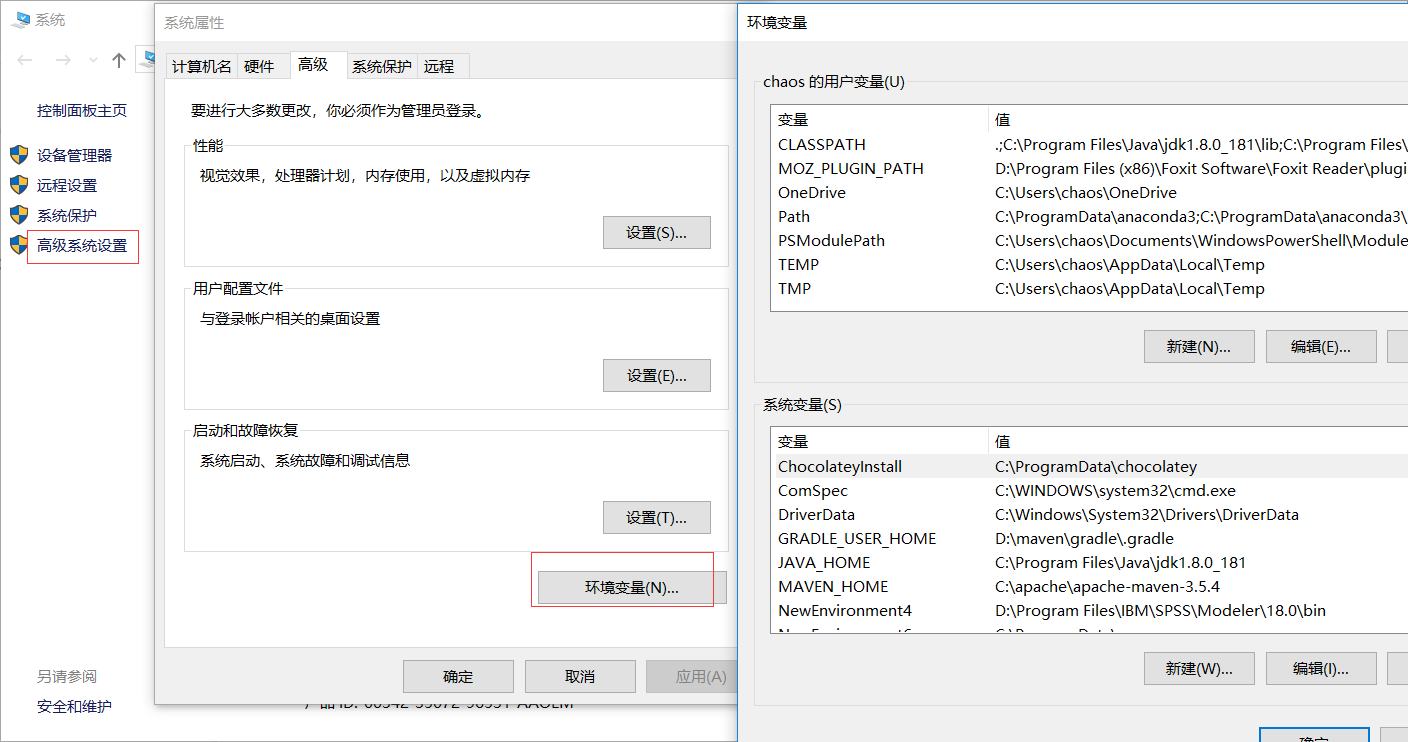
* gradle下载安装 [下载地址](https://gradle.org/releases/) ，我本地环境用的 v4.10.2
* 解压安装，安装路径不要出现空格（建议），如图

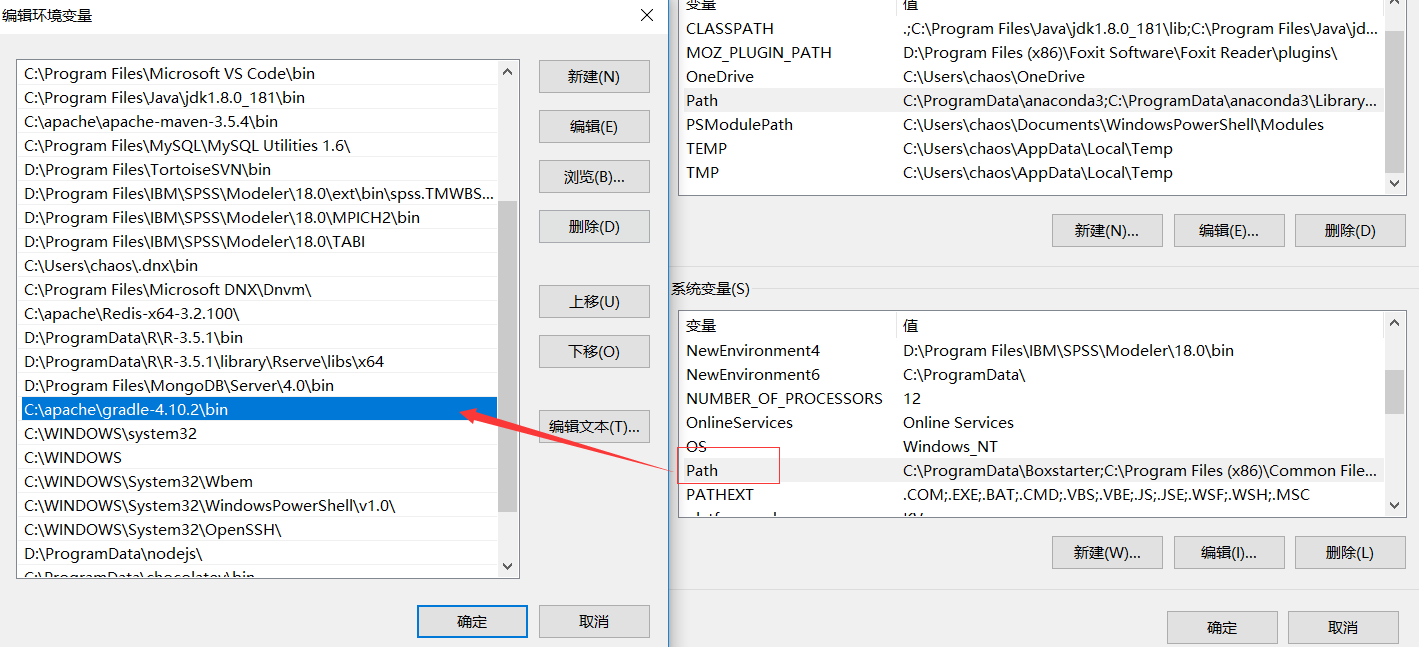


* 配置环境变量

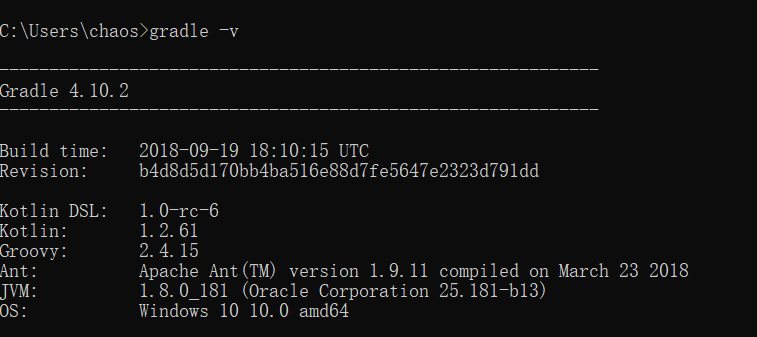
选中系统变量path，然后编辑，打开编辑环境，新建，

输入配置路径C:\apache\gradle-4.10.2\bin





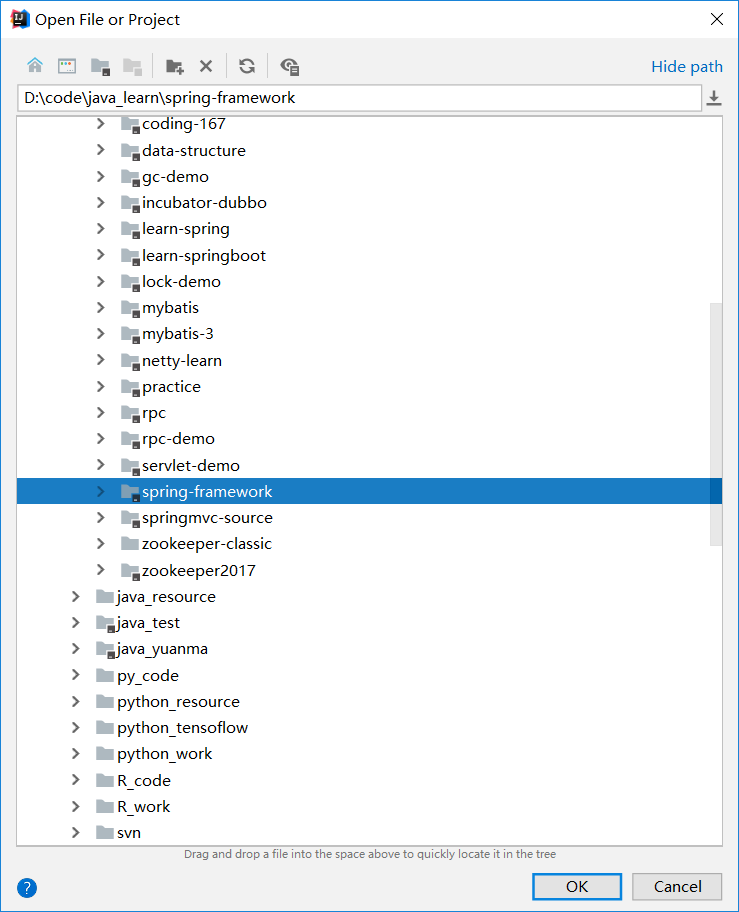
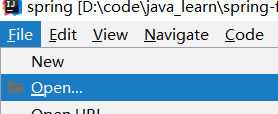
* 验证环境安装是否正常，如下图，表示正常



1. 导入idea,配置idea环境

* 导入spring-framwork到idea

打开idea, 选择File---Open，选择解压的spring-framwork，导入到idea



* 配置idea中gradle环境

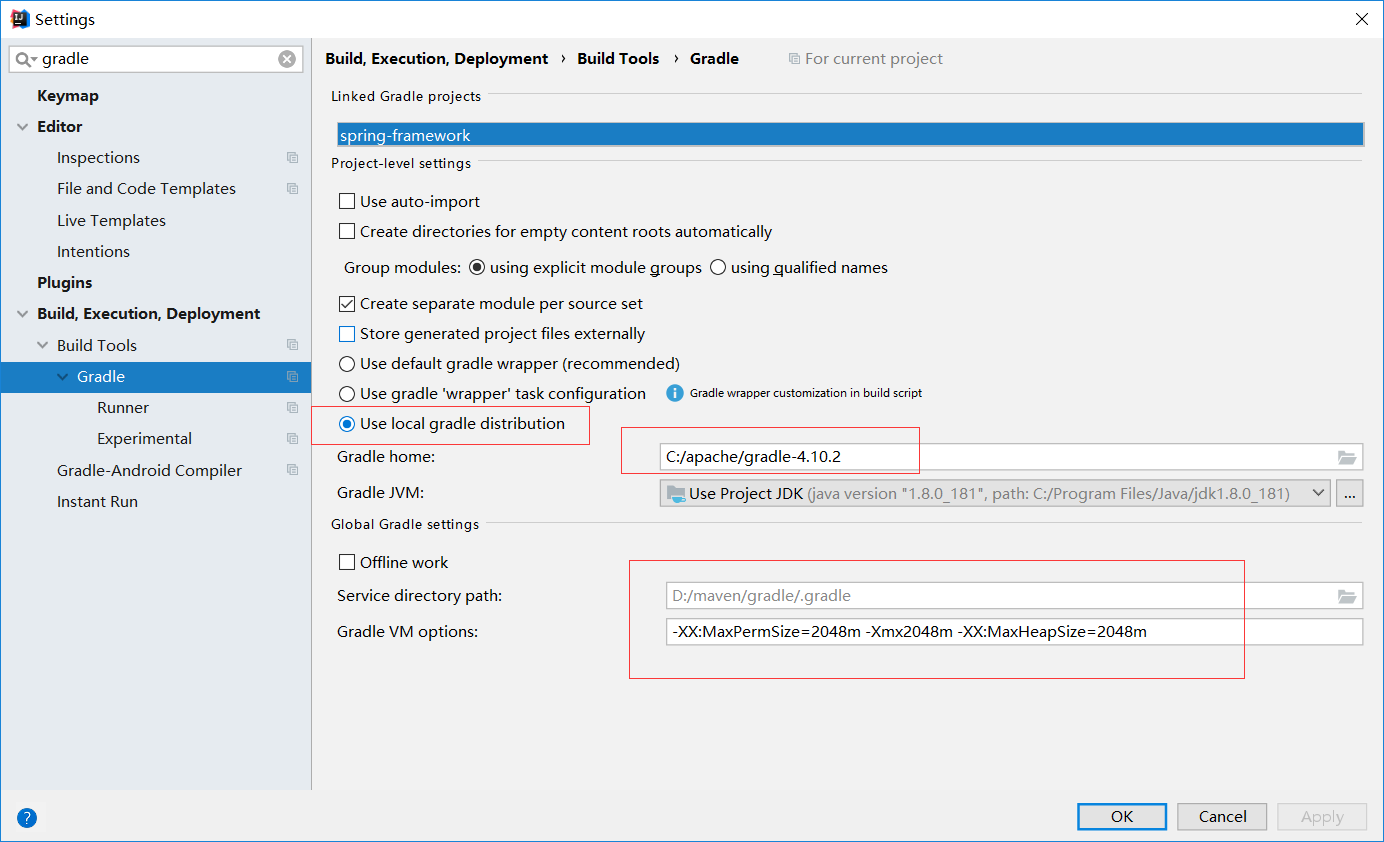
File---Settings ，搜索gradle, 配置gradle

指定gradle home: C:/apache/gradle-4.10.2 安装路径

Service directory path指定了gradle仓库： D:/maven/gradle/.gradle 自定义

vm参数可不配： -XX:MaxPermSize=2048m -Xmx2048m -XX:MaxHeapSize=2048m

确定，等待gradle加载环境，时间有点久，耐心等待



因为gradle构建时会下载国外的源，很慢，可以换阿里的源

可以在 USER\_HOME/.gradle/文件夹下添加 init.gradle文件来配置，我是配置在D:/maven/gradle/.gradle目录下，对应idea中的 Service directory path

init.gradle文件内容如下：

allprojects{

repositories {

def REPOSITORY\_URL = 'http://maven.aliyun.com/nexus/content/groups/public/'

all { ArtifactRepository repo ->

if(repo instanceof MavenArtifactRepository){

def url = repo.url.toString()

if (url.startsWith('https://repo1.maven.org/maven2') || url.startsWith('https://jcenter.bintray.com/')) {

project.logger.lifecycle "Repository ${repo.url} replaced by $REPOSITORY\_URL."

remove repo

}

}

}

maven {

url REPOSITORY\_URL

}

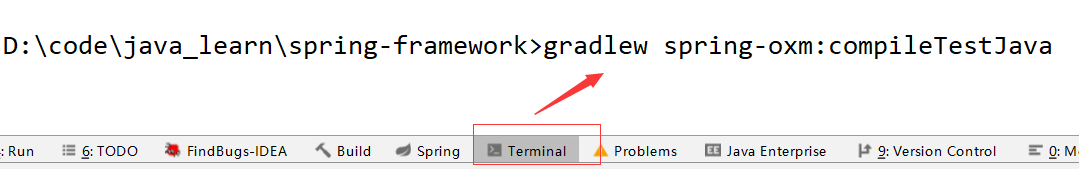
}

}

1. 编译源码

打开Terminal中端,输入 如下命令编译spring源码，等待结果

gradlew spring-oxm:compileTestJava

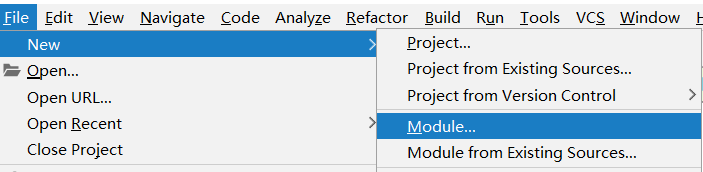


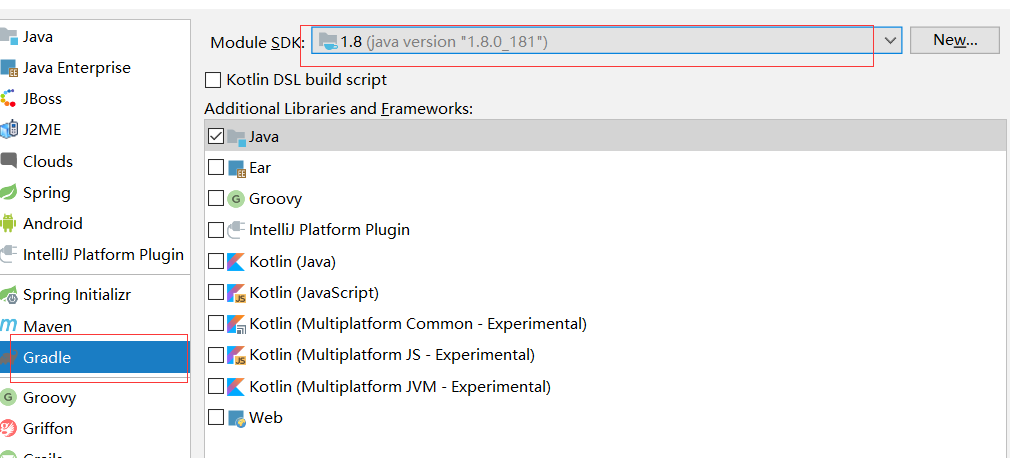
出现 BUILD SUCCESSFUL 表示编译成功

图片

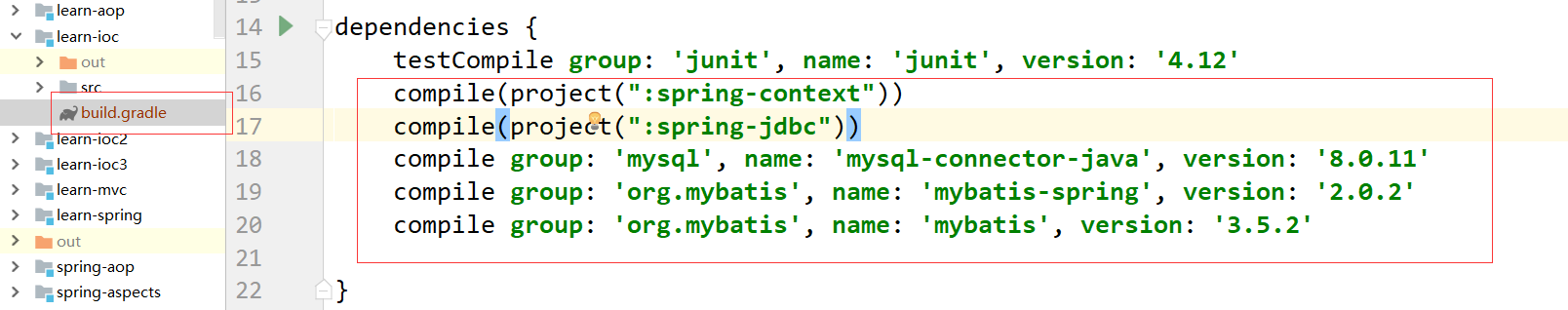
5.新建项目测试

新建一个module, 选择gradle项目





创建完成后打开build.gradle， 引入依赖jar，格式如下图



编写测试类，测试

