

一、Java 安装

1.1、什么是 Java

Java 是一种广泛使用的面向对象的编程语言，由 Sun Microsystems 公司于1995年推出。它的设计目标是实现一次编写，到处运行（Write Once, Run Anywhere, WORA），意味着一次编写的Java代码可以在支持Java的任何平台上运行，而无需重新编写。

Java 语言的主要特点包括：

跨平台性：Java 程序可以在任何支持 Java 的平台上运行，只要有对应的 Java 虚拟机（JVM）。

面向对象：Java 是一种面向对象的编程语言，支持封装、继承和多态等面向对象的特性。

健壮性：Java 具有强类型检查和异常处理机制，有助于编写健壮的程序。

安全性：Java 在设计时考虑了安全性，提供了安全性管理和垃圾回收机制。

多线程：Java 内置支持多线程编程，使得开发多任务的程序更加方便。

分布式计算：Java 支持远程方法调用（RMI），使得分布式计算变得更加容易。

动态性：Java 具有动态扩展的能力，可以在运行时加载新的类。

Java 分为不同的版本和规范，其中 Java Standard Edition（Java SE）和 Java Enterprise Edition（Java EE）是两个主要的版本。

Java SE（Standard Edition）：也称为 J2SE（Java 2 Platform, Standard Edition），是面向桌面和小型服务器应用程序的标准版 Java。它包含了用于开发和部署 Java 应用程序的核心类库、Java 虚拟机（JVM）和其他工具。Java SE 主要用于独立的应用程序和小型服务器端应用程序的开发。

Java EE（Enterprise Edition）：也称为 J2EE（Java 2 Platform, Enterprise Edition），是面向大型企业级应用程序的 Java 版本。Java EE 扩展了 Java SE，提供了一系列的 API 和规范，用于开发分布式、多层次、事务性的企业级应用。Java EE 包括诸如 Servlets、JSP（JavaServer Pages）、EJB（Enterprise JavaBeans）等技术，以支持开发复杂的企业级应用。

总体而言，Java SE 适用于一般的应用程序开发，而 Java EE 则适用于大规模、复杂的企业级应用程序开发。

1.2、什么是 Java 代码审计

Java 代码审计，是针对由 Java 代码编写的项目，进行漏洞挖掘发现潜在安全漏洞风险的技术。

Java 代码审计是一个统称，泛指针对 Java 语言进行代码审计的技术。

其中 JavaWeb 代码审计是 Java 代码审计的其中一个分支。JavaWeb 是指由 Java 语言开发的 WEB 项目。也就是我们常说的网站，后端语言是由 Java 开发的。

两者没有本质区别，都是基于 Java 语言进行的代码审计。

1.3、JDK 介绍

JDK（Java Development Kit）是用于开发 Java 应用程序的软件开发工具包。它包含了 Java 编译器（javac）、Java 虚拟机（JVM）、Java 类库（包括 Java 标准库和其他支持文件）以及一些其他的工具和实用程序，使开发者能够创建、编译和运行 Java 应用程序。

JDK的主要组成部分包括：

Java 编译器 (javac)： 用于将Java源代码编译成Java字节码。

Java 虚拟机 (JVM)： 用于执行Java字节码，使得Java程序能够在不同的平台上运行，实现“一次编写，到处运行”的特性。

Java 类库： 包括 Java 标准库和其他支持文件，提供了丰富的类和方法，方便开发者进行Java应用程序的开发。

开发工具和实用程序： JDK 包含一些开发和调试工具，例如调试器、监视器、性能分析工具等，以帮助开发者更轻松地进行开发和调试Java应用程序。

与JDK相对应的是JRE (Java Runtime Environment)，它是Java运行时环境，用于运行已经编译好的Java程序。JRE包括Java虚拟机(JVM)和Java类库，但不包括Java编译器等开发工具。因此，如果你只是想运行Java应用程序而不是进行开发，通常只需要安装JRE。

总结一下，JDK包含了JRE，而且还额外提供了用于Java应用程序开发的工具和工具集。如果你只是想运行Java应用程序，那么安装JRE就足够了。如果你需要进行Java应用程序的开发，那么你就需要安装JDK。

1.4、JDK 安装

拓展可自行了解Java的JDK版本，不在这里过多赘述。

简而言之：我们口中说的Java8、JDK8、JDK1.8其实都是同一个东西。

Java8发行版本说明中文版：https://www.java.com/zh-CN/download/help/release_changes.html

Java8发行版本说明英文版：<https://www.oracle.com/java/technologies/javase/8u-relnotes.html>

我们先以JDK 8u202为例进行安装，或者我常用的版本JDK 8u161，没有什么特别的，只不过是安装后就没怎么改了。后续练习如果有特殊版本需求，再做说明。

JDK1.8_211 后续版本全是商用收费的了。

基本上后续练习都是基于JDK 1.8的，具体小版本号，会特殊说明，需特殊关注。

官方下载地址，<https://www.oracle.com/java/technologies/javase/javase8u211-later-archive-downloads.html>。

使劲往下滑，选择对应版本下载。

JDK版本会对一些漏洞验证有影响，需要到具体的JDK版本我会再进行说明。

1.5、JDK 版本切换

在实际学习调试中，会经常遇见在各JDK版本切换的情况。

下面是我使用的一个切换JDK版本的VBS脚本，大家可以参考。

需要修改自己当前环境下各JAVA版本的绝对路径。

```
%1 mshta vbscript:CreateObject("Shell.Application").ShellExecute("cmd.exe","/c
%~s0 ::", "", "runas", 1)(window.close)&&exit cd /d "%~dp0"
@echo off
```

```

echo 当前Java版本
java -version

:menu
echo
=====
=====
echo 请选择要切换的jdk版本
echo 1: Java jdk 1.8
echo 2: Java jdk 11
echo 3: Java jdk 13
echo 0: 取消
echo
=====
=====
set /p ch=请选择:
if "%ch%"=="1" goto java1.8
if "%ch%"=="2" goto java11
if "%ch%"=="3" goto java13
if "%ch%"=="0" goto exit
goto menu

:java1.8
set JAVA_HOME=C:\Program Files\Java\jdk1.8.0_261
set JAVA_VERSION=1.8
goto exec

:java11
set JAVA_HOME=C:\Program Files\Java\jdk-11.0.13
set JAVA_VERSION=11
goto exec

:java13
set JAVA_HOME=C:\Program Files\Java\jdk-13.0.2
set JAVA_VERSION=13
goto exec

:exec
reg add "HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Development Kit" /v
CurrentVersion /t REG_SZ /f /d "%JAVA_VERSION%"
reg add "HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment" /v
CurrentVersion /t REG_SZ /f /d "%JAVA_VERSION%"
setx "JAVA_HOME" "%JAVA_HOME%" /m

del /f "C:\windows\System32\java.exe"
copy /Y "%JAVA_HOME%\bin\java.exe" "C:\windows\System32\java.exe"
del /f "C:\windows\System32\javaw.exe"
copy /Y "%JAVA_HOME%\bin\javaw.exe" "C:\windows\System32\javaw.exe"
set JAVA32_PATH=C:\Program Files (x86)\Common Files\Oracle\Java\javapath
del /f "%JAVA32_PATH%\java.exe"
mklink "%JAVA32_PATH%\java.exe" "%JAVA_HOME%\bin\java.exe"
del /f "%JAVA32_PATH%\javaw.exe"
mklink "%JAVA32_PATH%\javaw.exe" "%JAVA_HOME%\bin\javaw.exe"
set JAVA_PATH=C:\Program Files\Common Files\Oracle\Java\javapath
del /f "%JAVA_PATH%\java.exe"

```

```

mklink "%JAVA_PATH%\java.exe" "%JAVA_HOME%\bin\java.exe"
del /f "%JAVA_PATH%\javaw.exe"
mklink "%JAVA_PATH%\javaw.exe" "%JAVA_HOME%\bin\javaw.exe"
echo
=====
=====
echo 已切换到Java%JAVA_VERSION%
java -version
pause
goto exit
:exit

```

其他推荐:

```

https://www.mobaijun.com/posts/772710478.html
https://www.cnblogs.com/yuxuefeng/p/16143440.html
https://github.com/SkyBlueEternal/jdk-change

```

二、Maven 安装

2.1、Maven 介绍

Maven 是一个项目管理工具，它包含了一个项目对象模型（Project Object Model），反映在配置中，就是一个 pom.xml 文件。是一组标准集合，一个项目的生命周期、一个依赖管理系统，另外还包括定义在项目生命周期阶段的插件(plugin)以及目标(goal)。

当我们使用 Maven 的使用，通过一个自定义的项目对象模型，pom.xml 来详细描述我们自己的项目。

简单来说，我们开发一个JavaWeb项目是需要加载很多依赖的，使用Maven可以便于管理这些依赖。

Maven 仓库地址：<https://mvnrepository.com/>

pom.xml

POM 是项目对象模型(Project Object Model)的简称,它是 Maven 项目中的文件，使用 XML 表示，名称叫做 pom.xml。该文件用于管理：源代码、配置文件、开发者的信息和角色、问题追踪系统、组织信息、项目授权、项目的url、项目的依赖关系等等。Maven项目中必须包含 pom.xml 文件。

需要导入的依赖应该在 pom.xml 中进行配置与填写。比如导入某些依赖，如下图所示：



```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3     xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
4     <modelVersion>4.0.0</modelVersion>
5     <parent>
6         <groupId>org.springframework.boot</groupId>
7         <artifactId>spring-boot-starter-parent</artifactId>
8         <version>2.5.6</version>
9         <relativePath/> <!-- lookup parent from repository -->
10    </parent>
11    <groupId>com.example</groupId>
12    <artifactId>demo</artifactId>
13    <version>0.0.1-SNAPSHOT</version>
14    <packaging>war</packaging>
15    <name>demo</name>
16    <description>Demo project for Spring Boot</description>
17    <properties>
18        <java.version>1.8</java.version>
19    </properties>
20    <dependencies>
21        <dependency>
22            <groupId>org.springframework.boot</groupId>
23            <artifactId>spring-boot-starter-web</artifactId>
24        </dependency>
25    </dependencies>
26
27    <dependency>
28        <groupId>com.alibaba</groupId>
29        <artifactId>fastjson</artifactId>
30        <version>1.2.56</version>
31    </dependency>

```

`project` - `project` 是 pom.xml 中描述符的根。

`modelVersion` - `modelVersion` 指定 pom.xml 符合哪个版本的描述符。maven 2 和 3 只能为 4.0.0。

`parent` - maven 支持继承功能。子 POM 可以使用 `parent` 指定父 POM，然后继承其配置。

`dependencies` - 在 `dependencies` 中进行依赖配置

`groupId` - 团体、组织的标识符。团体标识的约定是，它以创建这个项目的组织名称的逆向域名 (reverse domain name) 开头。一般对应着 java 的包结构。

`artifactId` - 单独项目的唯一标识符。比如我们的 tomcat、commons 等。不要在 `artifactId` 中包含点号(.)。

`version` - 版本信息。

2.2、Maven 安装

官方下载地址: <https://maven.apache.org/download.cgi>

关于 Maven 环境部署与安装，可参考下面的文章，我安装的版本是 3.8.8，选择下载 Binary zip archive 即可。

<https://www.runoob.com/maven/maven-setup.html>

当然 IDEA 中内置了 Maven。

2.3、Maven 加速配置

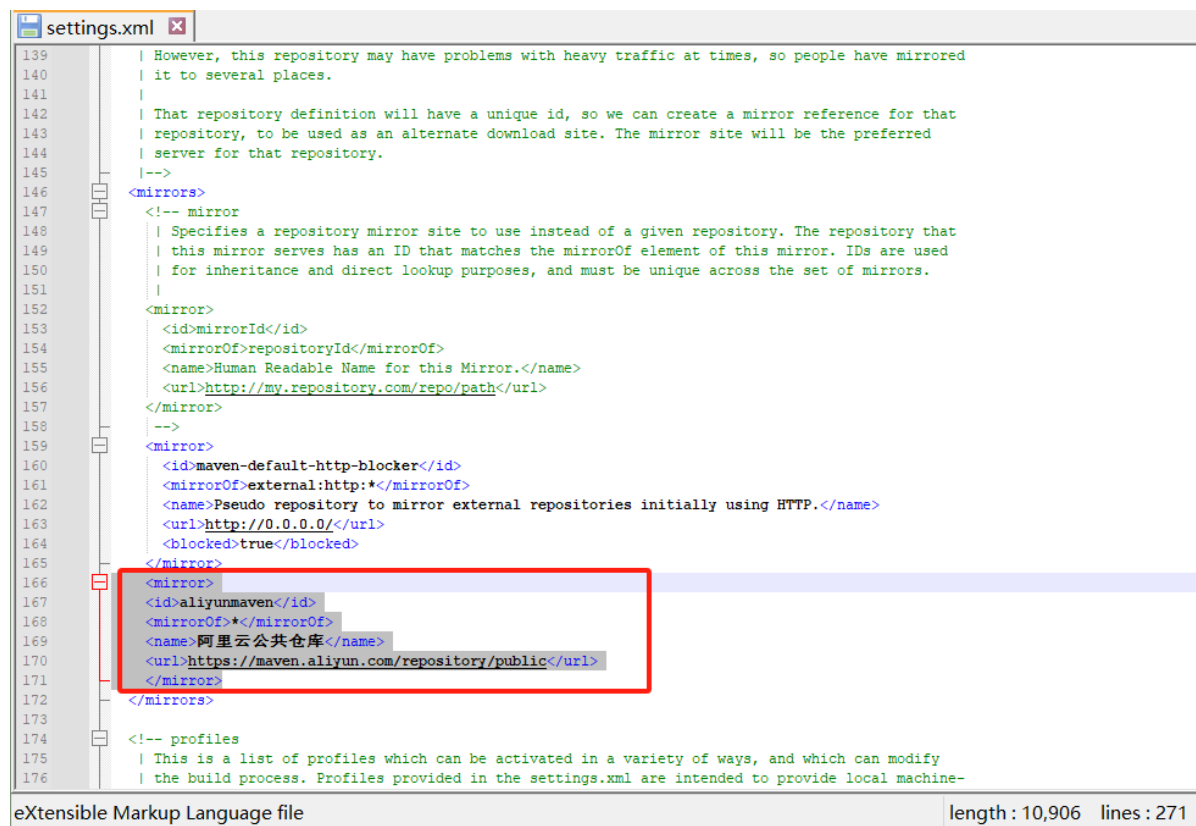
2.3.1、Maven 配置 (推荐)：

参考地址：<https://developer.aliyun.com/mvn/guide>

下载成功 Maven 后，进入 Maven 的 conf 目录下，找到 settings.xml，使用任意文件编辑器打开即可。

向下滑，找到 <mirrors> 标签，在该标签内添加以下加速源即可，最终如下图所示：

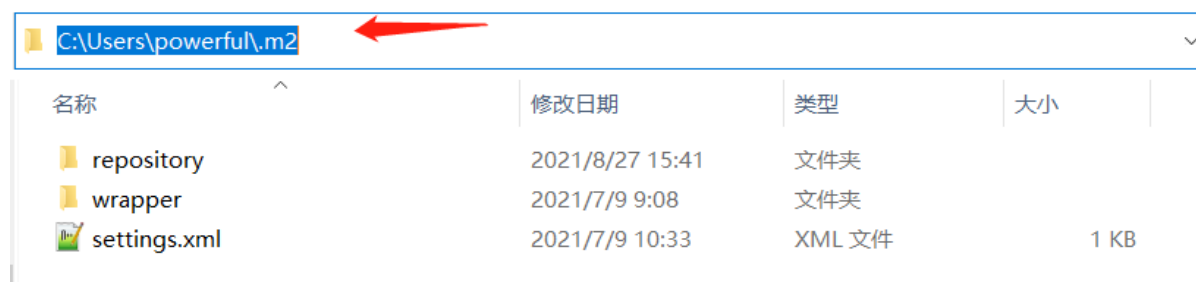
```
<mirror>
<id>aliyunmaven</id>
<mirrorOf>*</mirrorOf>
<name>阿里云公共仓库</name>
<url>https://maven.aliyun.com/repository/public</url>
</mirror>
```



2.3.2、本地配置：

配置国内源下载一些依赖组件会非常快，但会有极个别情况，有些组件使用国内源无法下载，则需要再更改配置，大家留有印象就好。

①、访问 c:\Users\当前用户\.m2 目录，当前用户文件夹需要根据当前用户来定，如下图所示：



②、打开 settings.xml 文件，复制粘贴以下内容：

参考链接: <https://developer.aliyun.com/mvn/guide>

fcmit.cc

```
<?xml version="1.0" encoding="UTF-8"?>
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
http://maven.apache.org/xsd/settings-1.0.0.xsd">
  <pluginGroups></pluginGroups>
  <proxies></proxies>
  <localRepository>C:\Users\当前用户\.m2\repository</localRepository>
  <servers>
  </servers>

  <mirrors>
    <mirror>
      <id>aliyunmaven</id>
      <mirrorOf>*</mirrorOf>
      <name>阿里云公共仓库</name>
      <url>https://maven.aliyun.com/repository/public</url>
    </mirror>
  </mirrors>
</settings>
```

注意代码中的 当前用户 该位置路径应与你当前用户一致。

三、Mysql 安装

3.1、Mysql 介绍

MySQL 是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，属于 Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统关系型数据库管理系统(696511)之一，在 WEB 应用方面，MySQL 是最好的 RDBMS(Relational Database Management System，关系数据库管理系统) 应用软件之一。

MySQL 是一种关系型数据库管理系统，关系数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言。MySQL 软件采用了双授权政策，分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型和大型网站的开发都选择 MySQL 作为网站数据库。

3.2、Mysql 安装

我采用的是 PHPStudy 的 Mysql，他提供各个版本的 Mysql，还便于一键切换。

官方下载地址：

<https://www.xp.cn/download.html>

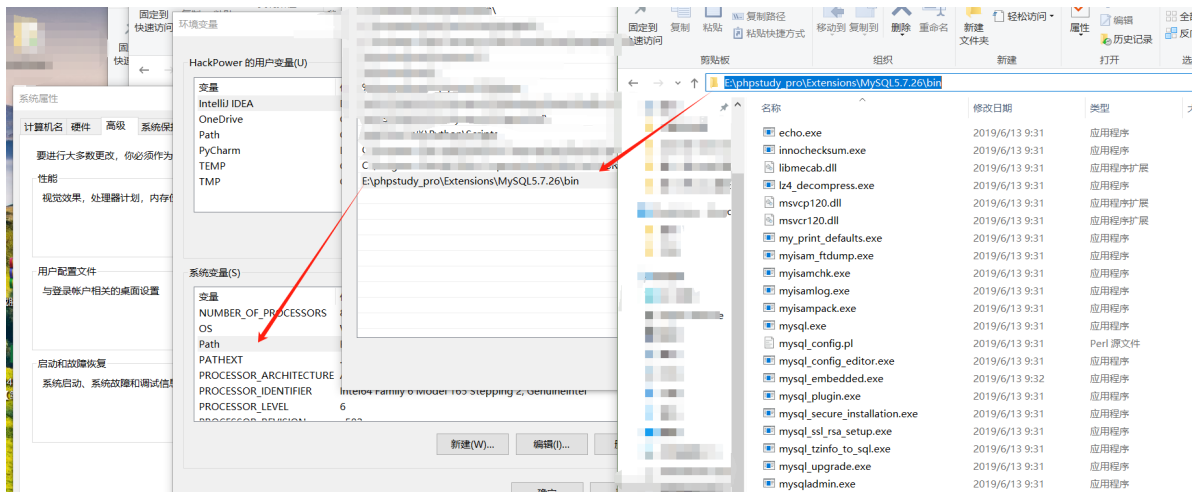
安装成功后，在首页下找到 Mysql，点击启动即可。如下图所示：



下面我们需要将 Mysql 添加至环境变量，便于调用。

第一步：找到 PHPStudy 安装目录，进入 Extensions 目录，找到对应的 Mysql 目录后，在进入 bin 目录，复制整个路径。

第二步：打开环境变量，将 Mysql 绝对路径添加至系统变量中，如下图所示。



第三步：打开 CMD 命令行，输入 `mysql -u root -p` 然后敲击回车键，输入密码，即可看到通过命令行链接到了 Mysql 数据库中，如下图所示：


```

C:\Users\power>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.26 MySQL Community Server (GPL)

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

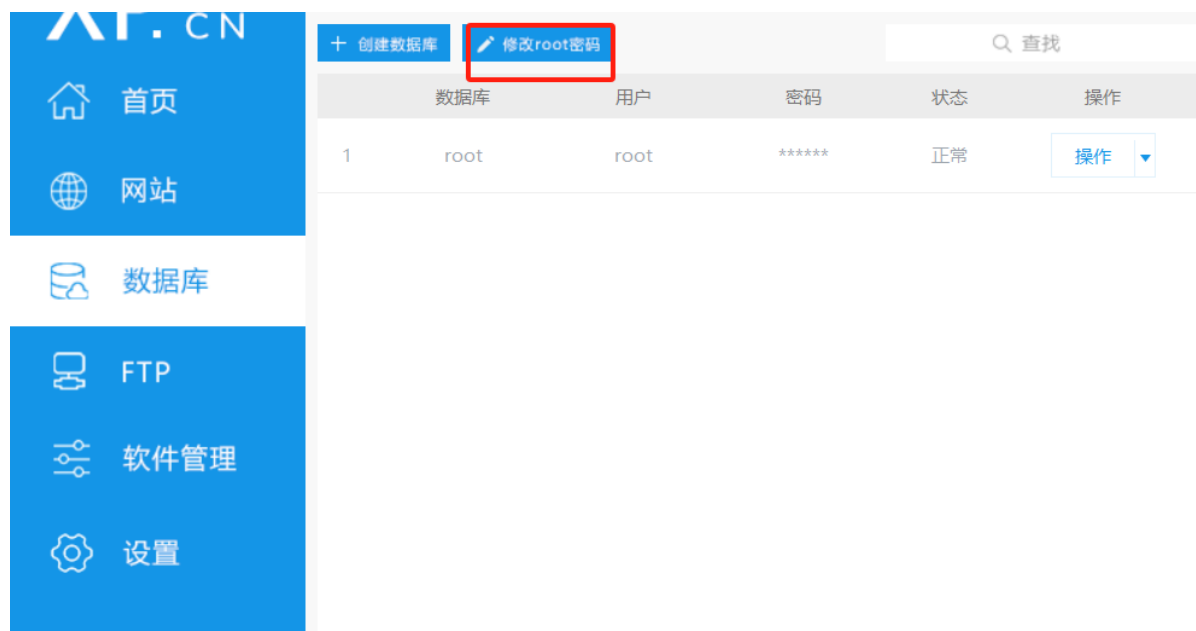
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| challenges      |
| mysql           |
| performance_schema |
| rbac            |
| security        |
| sys             |
| xenforo         |
+-----+
8 rows in set (0.00 sec)

```

一般账号密码默认为 root/root，可点击左侧数据库栏后在密码处修改，如下图所示：



四、Tomcat 安装

1、Tomcat 介绍

Apache Tomcat（通常简称为Tomcat）是一个开源的 Java Servlet 容器，由 Apache 软件基金会开发。它实现了Java Servlet和JavaServer Pages（JSP）规范，并提供了一个运行这些Web应用程序的环境。Tomcat是一个轻量级的、可扩展的、开源的服务器，被广泛用于部署 Java Web 应用程序。

以下是一些Apache Tomcat的关键特点和概念：

Servlet容器： Tomcat是一个Java Servlet容器，负责管理和执行Java Servlet和JSP。

开源和免费： Tomcat是开源的，可以免费使用。这使得它成为许多Java开发者首选的Servlet容器。

轻量级：Tomcat相对较轻，易于安装和配置。它专注于Servlet和JSP的支持，不包含其他不必要的功能，这有助于提高性能和减小内存占用。

可扩展性：Tomcat是可扩展的，允许用户通过添加插件和扩展来增强其功能。

支持Java Servlet和JSP：Tomcat实现了Java Servlet和JSP规范，使得开发者能够创建动态的、基于Java的Web应用程序。

连接器和协议支持：Tomcat支持多种连接器和协议，包括HTTP、AJP（Apache JServ Protocol）等，使得它能够与不同的前端和后端服务器进行集成。

安全性：Tomcat提供了一些安全功能，包括SSL/TLS支持、用户身份验证和访问控制，以确保Web应用程序的安全性。

管理工具：Tomcat提供了Web界面和命令行工具，用于管理和监控Tomcat服务器，包括部署、启动、停止Web应用程序等。

生态系统：Tomcat是一个活跃的开源项目，拥有庞大的社区支持。这意味着有许多插件、工具和第三方扩展可用于增强Tomcat的功能。

2、Tomcat 安装

Tomcat 官方网站：<https://tomcat.apache.org/>

Tomcat 目前有8, 9, 10, 11多个版本，后续都会基于 Tomcat 8 进行讲解，如有特别会特殊说明。

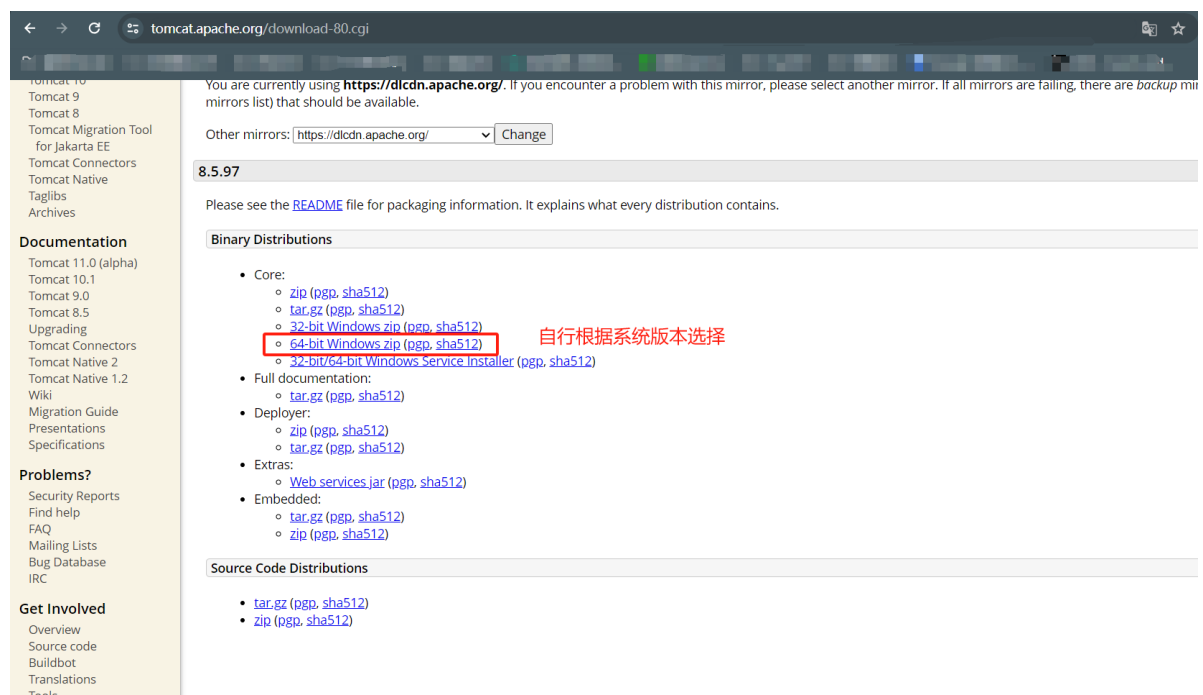
Tomcat8 官方下载地址：<https://tomcat.apache.org/download-80.cgi>。

Apache Tomcat提供两种主要类型的发行版：Binary Distributions（二进制发行版）和Source Code Distributions（源代码发行版）。

Binary Distributions（二进制发行版）：二进制发行版包含已经编译好的可执行文件和相关的库，使得用户可以直接使用 Tomcat 而无需自行编译源代码。

Source Code Distributions（源代码发行版）：源代码发行版包含 Tomcat 的完整源代码。这意味着用户需要自行编译这些源代码以生成可执行的 Tomcat 服务器。

我们需要下载 **Binary Distributions（二进制发行版）**，如下图所示：



成功下载后，会得到一个 zip 压缩包，我们将文件解压到常用的目录中即可。

需要固定使用该目录，因为在 IDEA 中会配置 Tomcat 路径。

五、IDEA 安装

1、IDEA 介绍

IntelliJ IDEA（简称IDEA）是由 JetBrains 公司开发的一款强大的集成开发环境（IDE），用于 Java、Kotlin、Groovy、Scala 等语言的开发。它被广泛认为是一款功能丰富、灵活且高效的IDE，被许多开发者用于各种类型的软件开发，包括 Web 应用、桌面应用、移动应用等。

智能代码编辑：IDEA 提供了强大的代码编辑功能，包括自动代码补全、实时错误检测、代码导航等。它能够理解上下文并提供智能建议，提高开发效率。

内置工具和集成：IDEA 集成了许多常用的开发工具，如版本控制系统（Git、SVN等）、构建工具（Maven、Gradle等）、数据库工具等，使得开发者可以在一个环境中完成多个任务。

强大的调试器：IDEA 提供了先进的调试功能，包括断点调试、变量监视、表达式求值等，帮助开发者更轻松地诊断和修复代码问题。

自动化测试支持：IDEA 支持各种自动化测试框架，如JUnit、TestNG等，以及集成了代码覆盖工具，帮助开发者编写和运行单元测试。

Web开发支持：对于 Web 开发，IDEA 提供了对 HTML、CSS、JavaScript、Java EE、Spring 等框架的全面支持。它还支持常见的前端技术，如 Angular、React、Vue 等。

Kotlin 支持：IDEA 是 Kotlin 的官方 IDE，提供了深度集成和对 Kotlin 的全面支持。Kotlin 是一种现代的、表达力强的编程语言，可以与 Java 互操作，并逐渐成为Android 开发的首选语言。

插件生态系统：IDEA 具有丰富的插件生态系统，允许开发者通过安装插件来扩展其功能。社区和第三方插件提供了各种工具和支持，满足不同开发需求。

智能重构：IDEA 提供了多种智能的代码重构工具，使得开发者可以轻松地重构代码结构、变量名等，而不影响程序的功能和稳定性。

2、IDEA 安装

对于 Java 代码审计的调试运行工具，我强烈推荐 IDEA，当然后面学习与调试都会基于此工具。

IDEA 全称 IntelliJ IDEA，是 Java 语言开发的集成环境。

IDEA社区版的话，是没有 WEB 和企业开发，大大的限制了我们的使用。

我们需要使用 Ultimate 版本，也就是企业版，但这个版本是收费的。

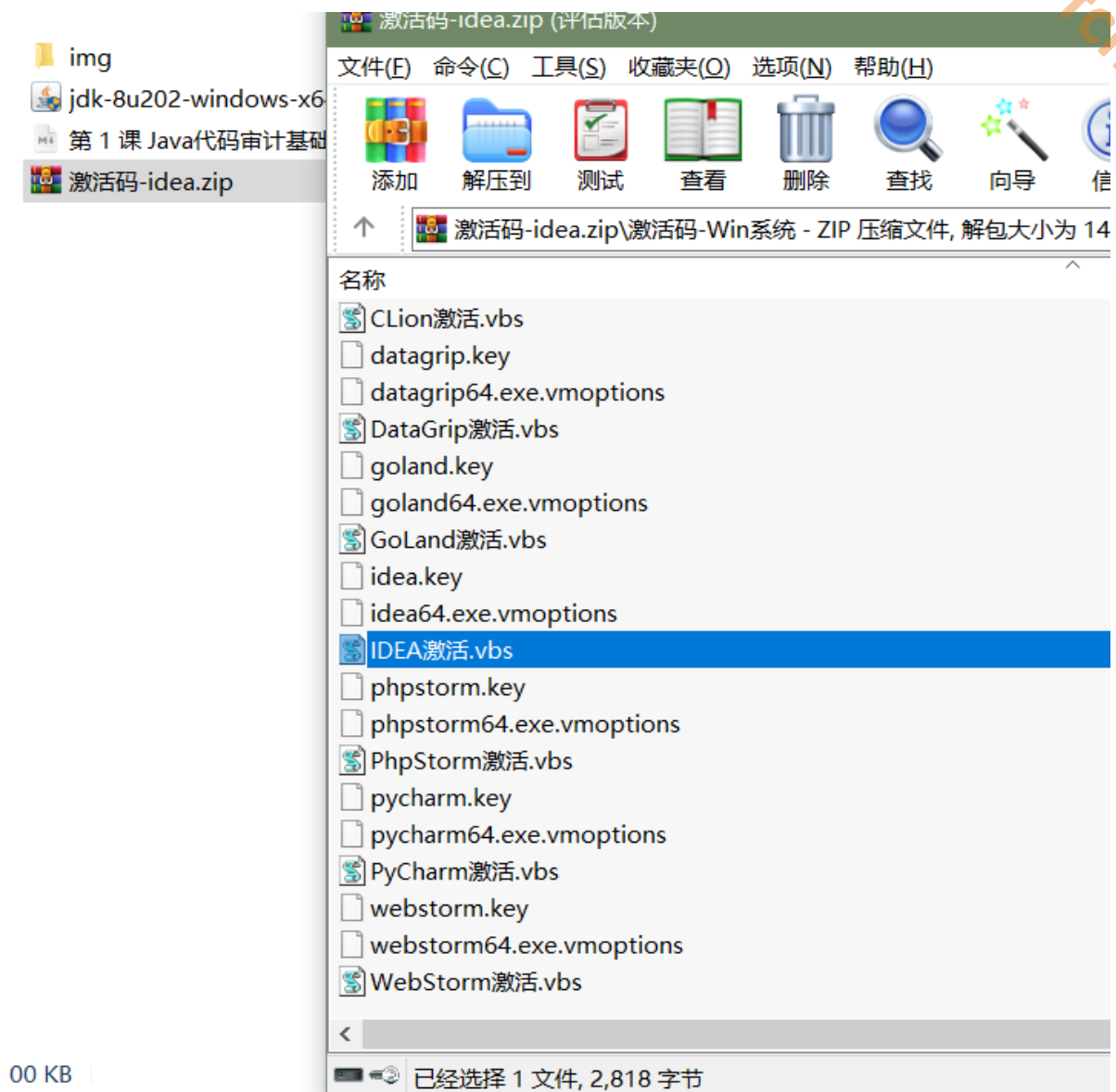
初次使用可免费使用30天。

那我们可能需要动点歪脑筋了。

课件会使用两个版本的 IDEA，一个比较新的是 IDEA-2023.1.2 版本，另一个就是 IDEA-2021 版本了，目的是便于有些课程的练习，所以会使用不同的版本，我们现在应先专注代码，减少配置。

这两个版本都大差不差，对于代码审计影响目前还不是很大。

然后，使用下方小工具，找到 IDEA激活，双击即可。如下图所示：



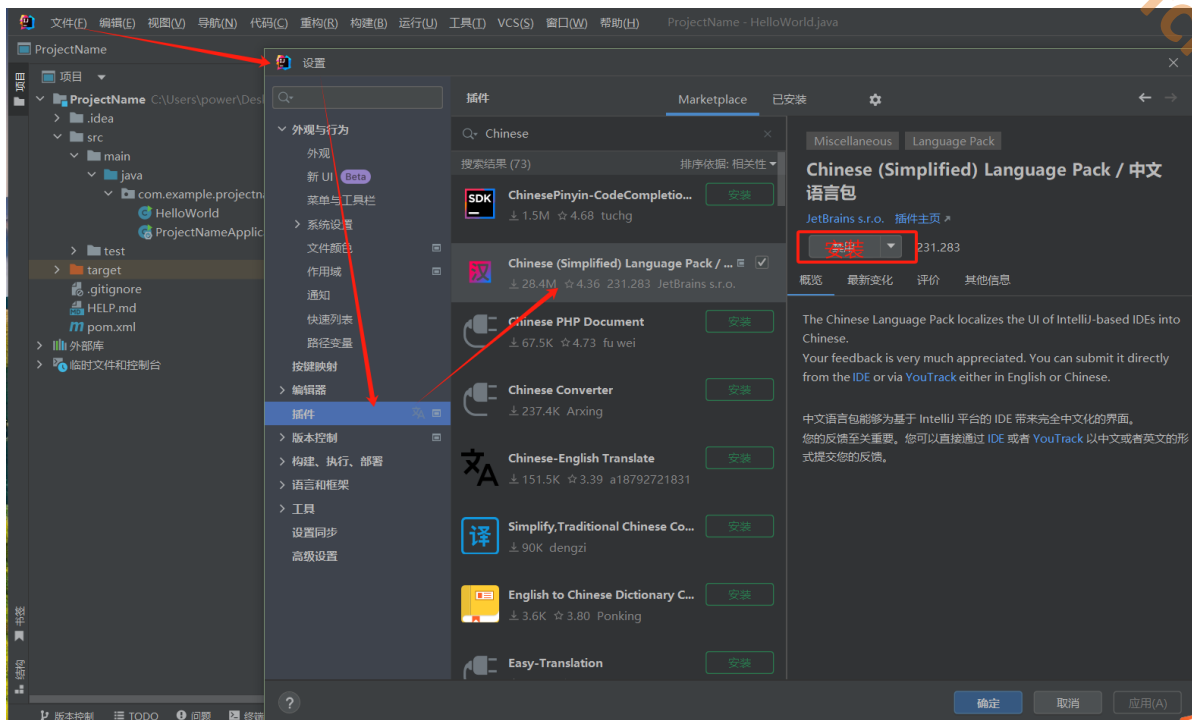
3、IDEA 基础使用教程

3.1、设置中文

作为母语是中文的我们来说，使用英文版的软件，难免会不适应。前几年还需要各种汉化版本，现在直接在插件库里安装下载中文包，翻译比较准确，这样用起来就得心应手多了。

设置中文步骤很简单。

点击左上角文件 - 设置 - 插件 - Marketplace，搜索 chinese 即可，大家一开始都是英文版，可以根据下图自行比对位置，进行下载安装。



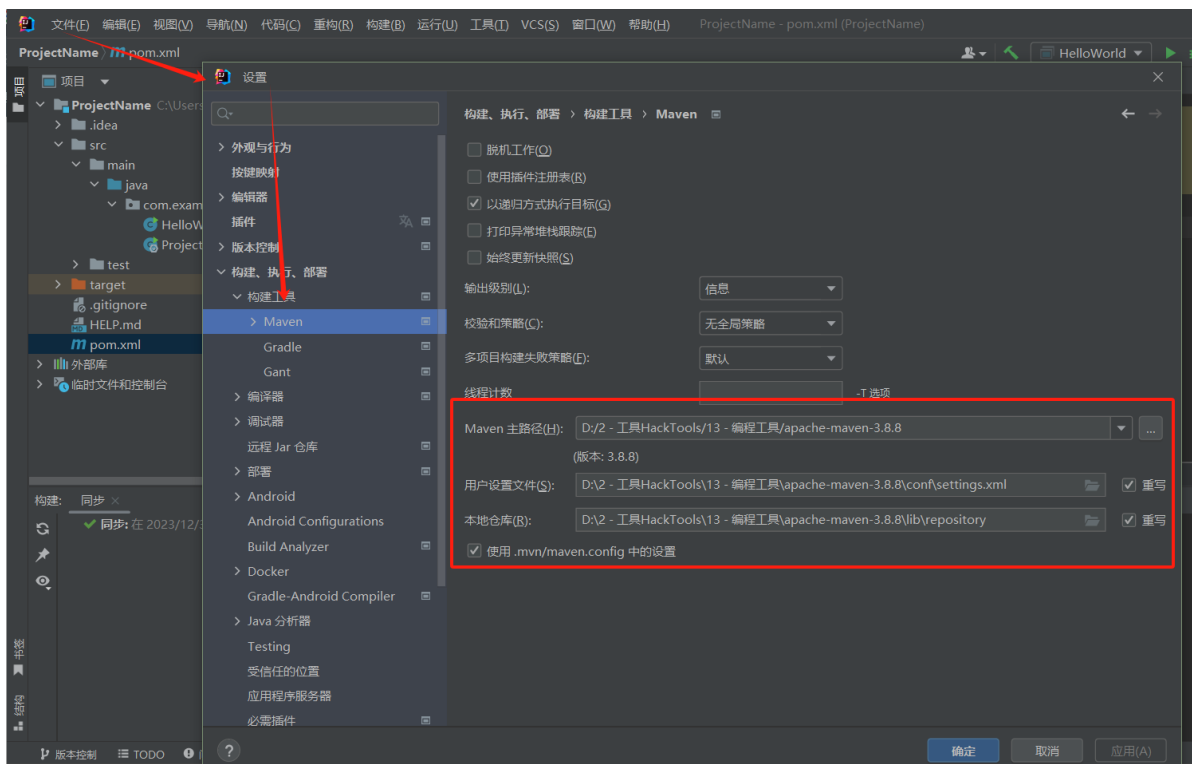
3.2、Maven 配置

3.2.1、修改 Maven 中央仓库

在 IDEA 中配置我们前面下载的 Maven，如果熟练也可以使用默认的。

如果仅是修改设置中 Maven 中央仓库则是临时的，我们需要在新项目设置中设置成永久的。具体配置如下。

点击左上角文件，找到 新项目设置 - 设置 - 构建、执行、部署 - 构建工具 - Maven，点击即可，我们设置咱自己下载的 Maven，以及配置文件，还有本地仓库。默认的本地仓库是在 C 盘（也可以设置），这样会极大压缩我们 C 盘空间，不妨设置个专门的本地仓库。



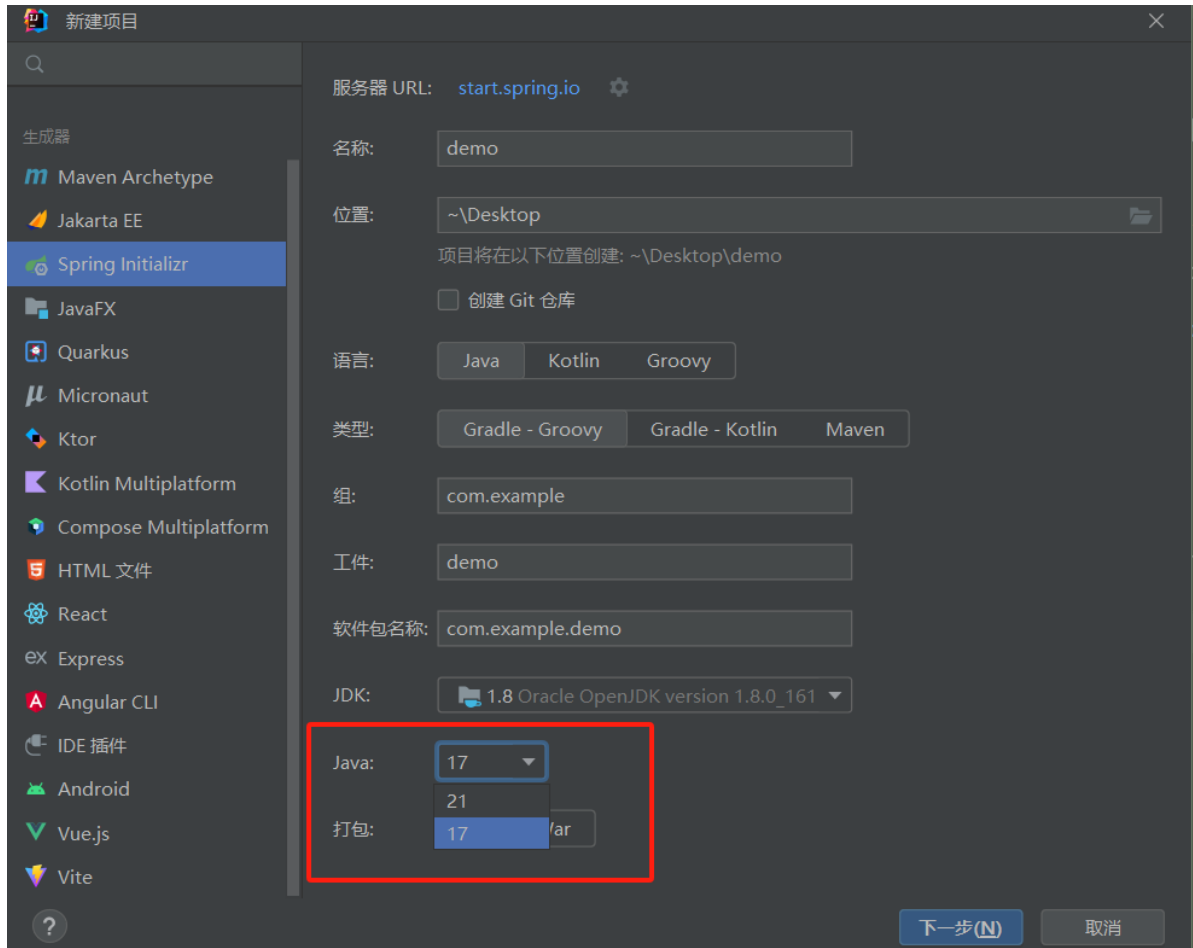
最后点击应用即可。

3.3、Tomcat 配置

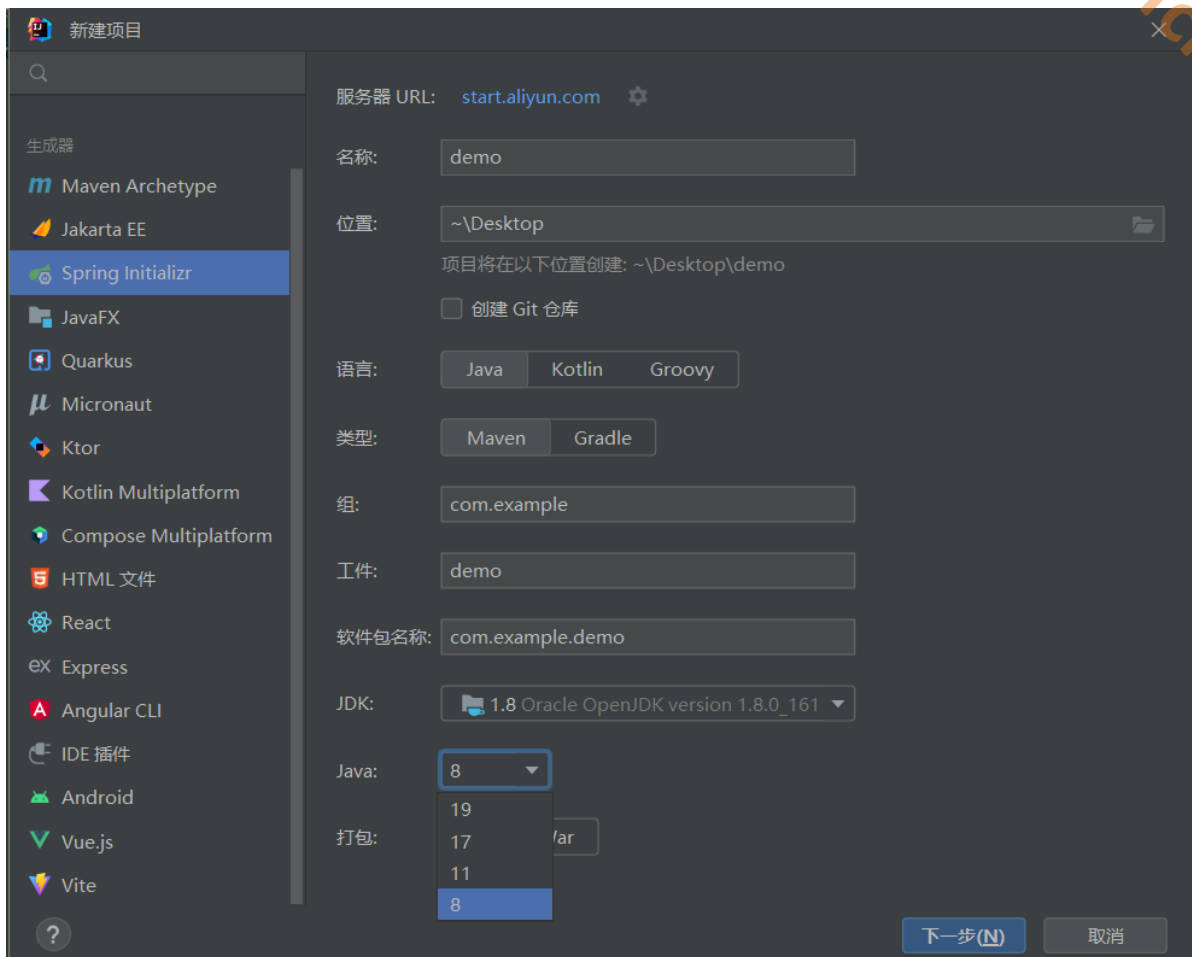
这个配置再具体遇见 Spring 项目会进一步讲解。

3.4、新建项目修改服务器 URL

IDEA 抛弃了 Java 8，所以现在使用 IDEA 创建 Spring Boot 项目时，不能选择 Java 8 了。就像下面这个样子。



解决方法很简单，将服务器 URL 处改为 <https://start.aliyun.com> 即可，这是阿里云提供的脚手架镜像。



3.5、Maven 加载各种报错

IDEA 在加载 Maven 配置的相关依赖时，会出现各种报错。

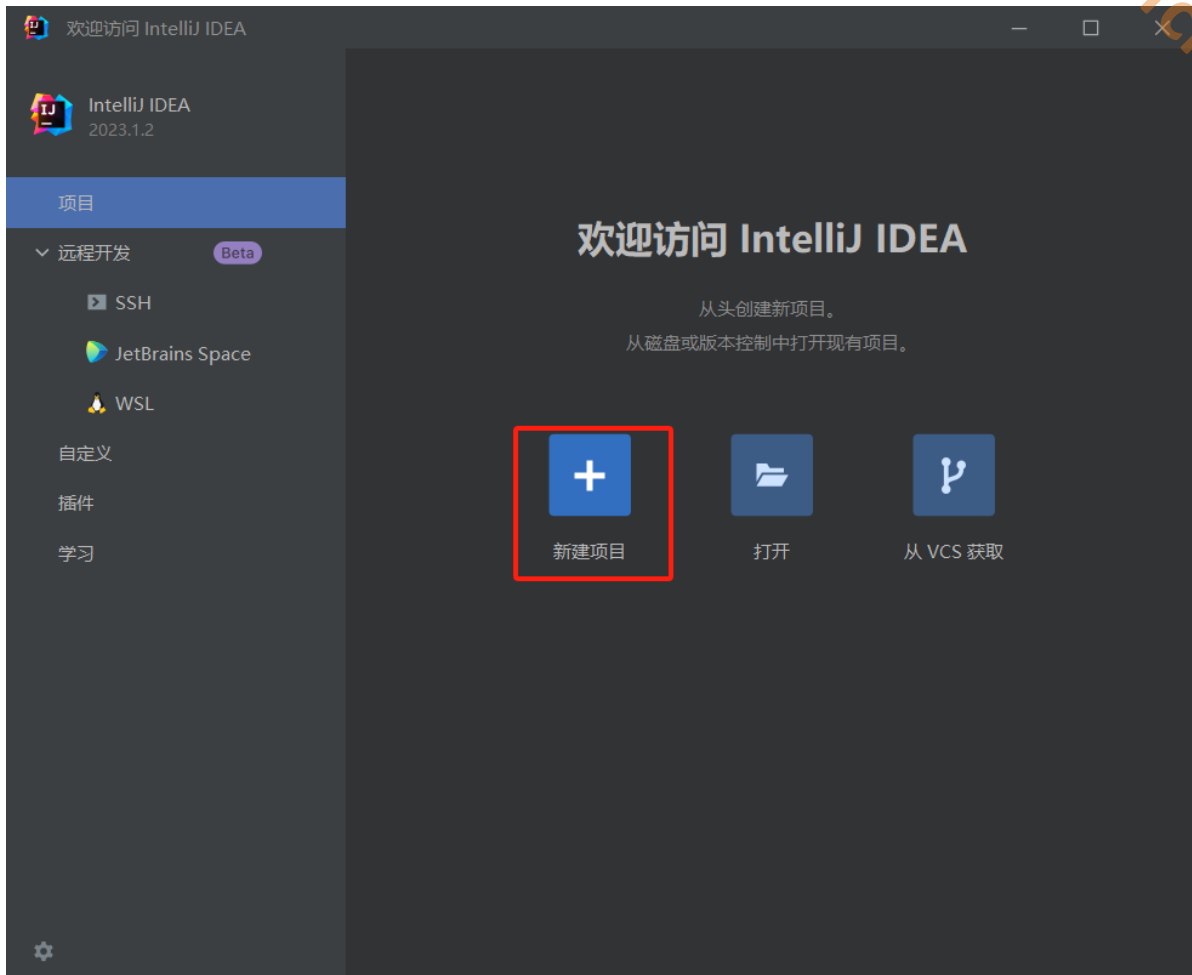
我们可以尝试上述配置为国内源。

我们可以尝试在 文件 中选择清除缓存，并且重启。

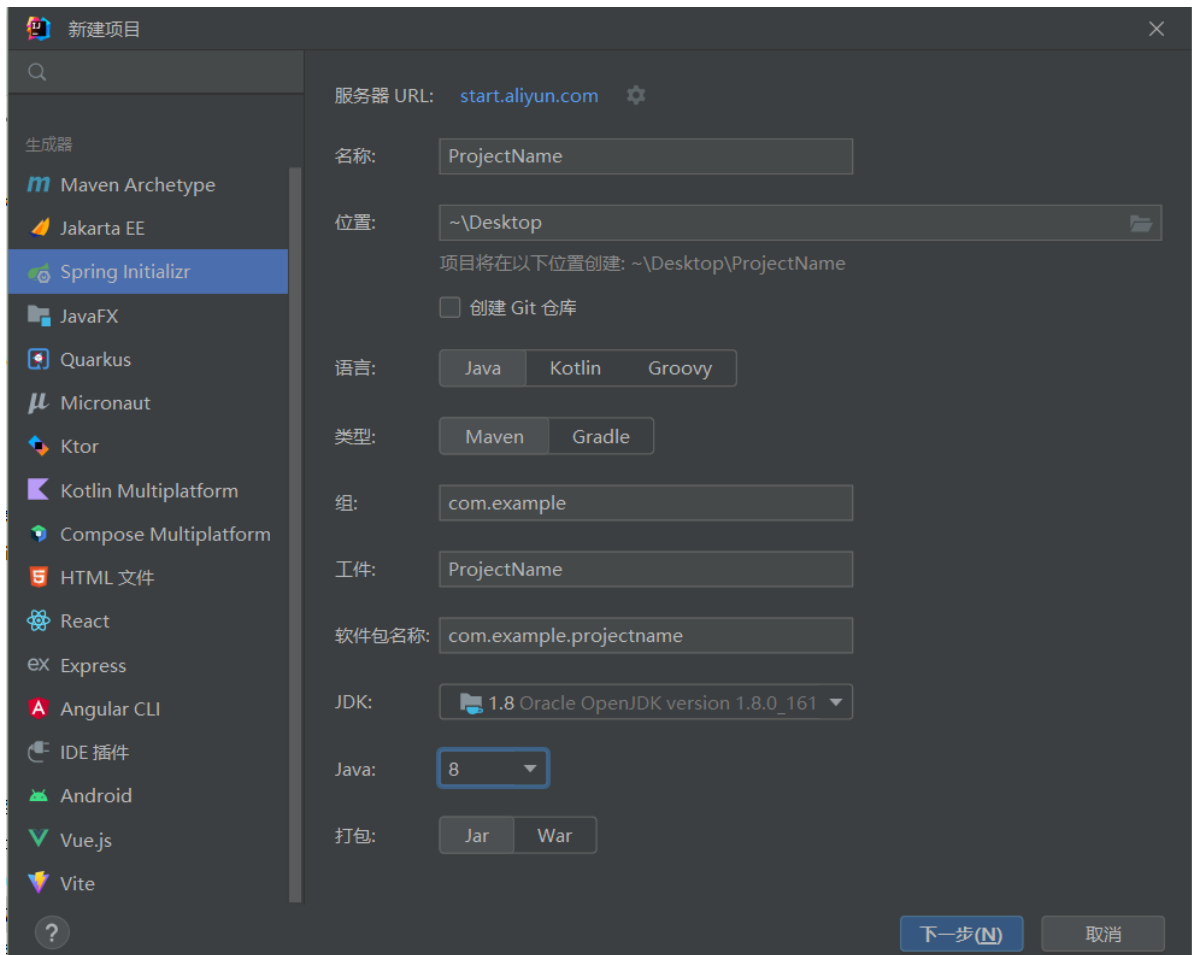
来解决相关问题。

4、第一个 IDEA 项目

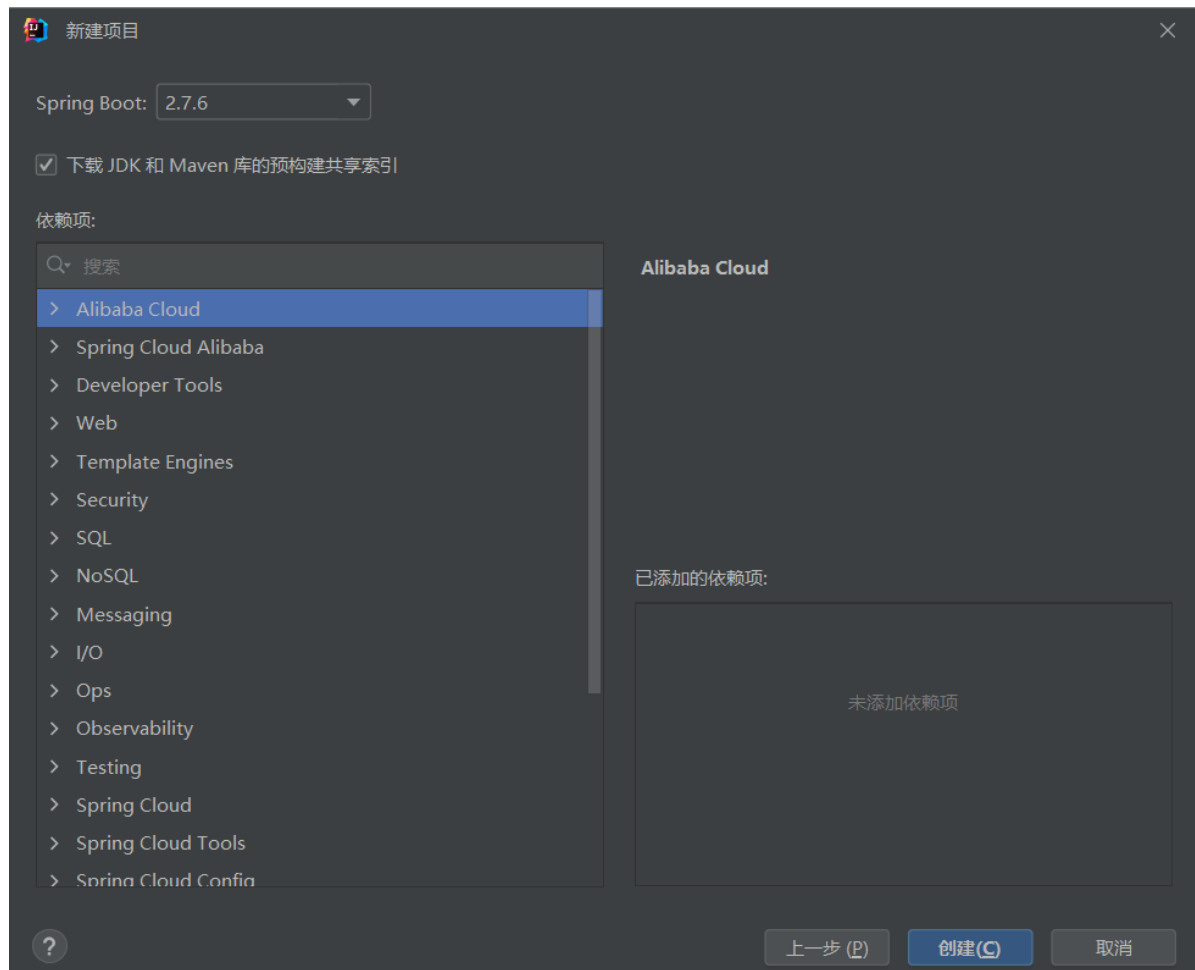
第一步：打开 IDEA，点击新建项目。



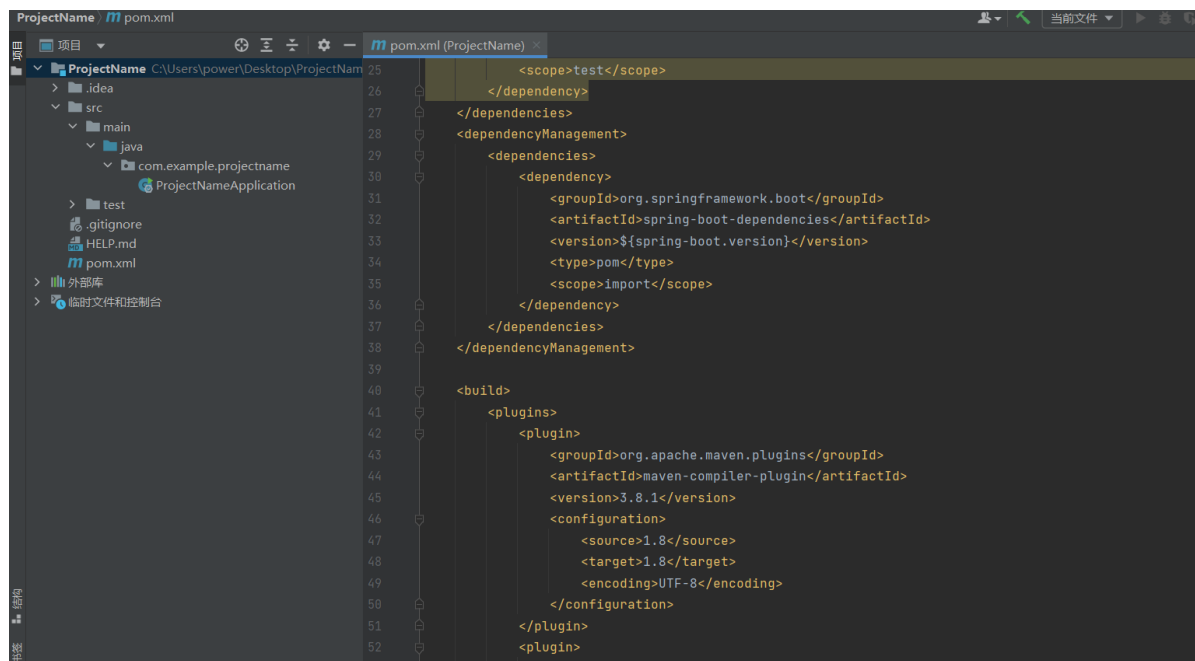
第二步：选择 Spring Initializr 创建 SpringBoot 项目，由于最新版 IDEA 弃除了 JDK 1.8，所以我们需要在服务器 URL 处设置为阿里云的地址。另外注意一下各个选项的选择，如下图所示：



第三步：选择所需的依赖以及 Springboot 项目版本，现在我们默认不选择即可。后续练习会特殊说明，如下图所示：



第四步：最后点击创建即可，如果我们选择了依赖会在后台进行下载。稍等片刻，如果项目编译完成，有如下几个特征。其一是代码文件被编译成了 Class 文件。其二是 Pom.xml 文件中没有任何报错。如下图所示：



上面我们是以 SpringBoot 架构进行的项目创建，实际上对于平时练习的小项目，也可以使用这套架构。

我们只需将代码编写在 com/example/projectname 路径下即可。

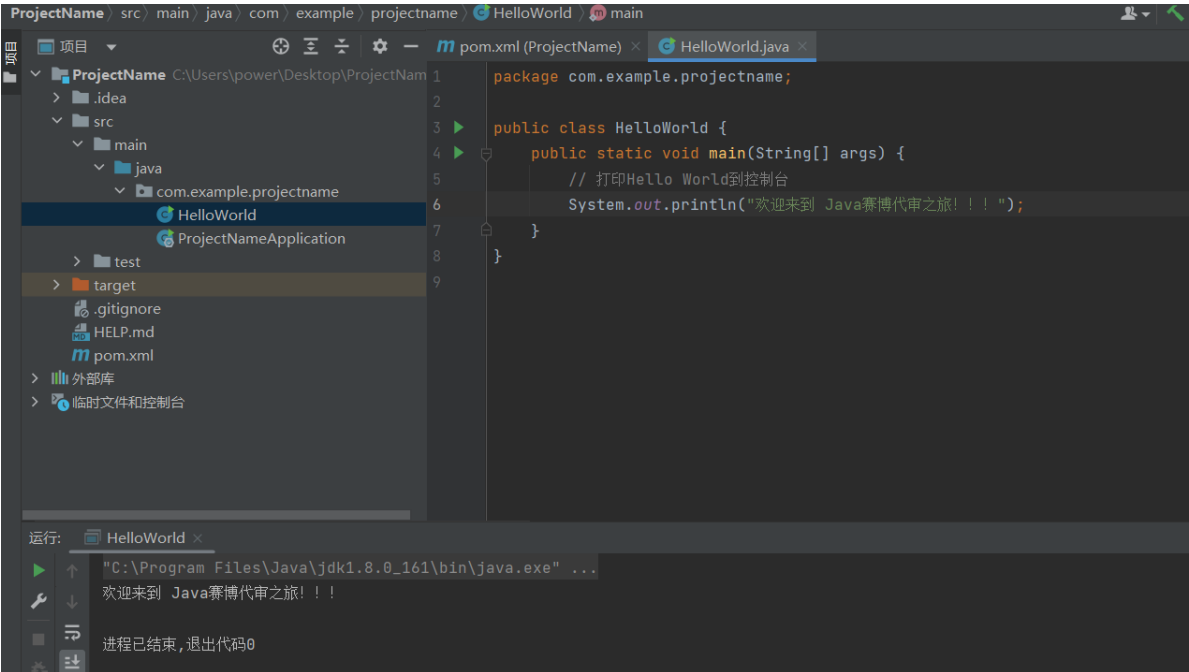
右键该目录，选择 新建-Java Class 即可，如下图所示：



随便起个名字。

在 Java 中，`main` 函数是程序的入口点，它是 Java 应用程序执行的起始点

我们创建个 `main` 函数，随便编写点什么。



至此，Java 代码审计基础环境搭建完毕了。

后续如果还有其他需求环境，会特殊说明。