

本章将进入正式的项目开发。首先介绍如何搭建 Spring Boot 项目的基础环境,其中主要包括 JDK 的安装和配置、项目开发工具的安装与体验,以及 Maven 的安装和配置。建议读者的开发环境与本书安装的环境等版本保持一致,以避免因版本不兼容等问题而导致各种错误的发生。这样能够确保在开发过程中获得更加稳定和一致的工作环境。

3.1 JDK 的安装和配置

在 2.1.3 节中提到了本项目所使用的 Spring Boot 版本为 3.1.3,要求最低使用 JDK 17 的环境,因此,需要选择安装和配置 JDK 17 版本。

3.1.1 JDK 的概念

JDK(Java Development Kit)是 Java 语言的软件开发工具包,提供了 Java 程序的编译器、虚拟机、调试器及其他辅助工具。它被用于开发和运行 Java 应用程序和 Applet。作为 Java 平台的核心组件,JDK 在 Java 语言体系中扮演着重要角色。主要版本包括 Java SE(标准版)、Java EE(企业版)和 Java ME(微型版),分别针对桌面应用程序、Web 应用程序和移动应用程序的开发。

3.1.2 下载 JDK

首先访问 Oracle 官方网站 <https://www.oracle.com/>,然后进行登录。如果没有账号,则需要自行注册一个 Oracle 账号,登录界面如图 3-1 所示。

选择 Resources→Java Downloads 选项,如图 3-2 所示。

跳转页面之后,页面上会出现相关版本的 JDK 安装包供下载,选择 JDK 17→Windows→x64 Installer 选项,如图 3-3 所示。根据计算机系统的配置,选择后缀为 .exe 的安装包下载,如果是 64 位的系统,则需要下载对应的 x64 Installer 的 JDK 版本;如果是 32 位的系统,则需要下载对应的 x86 Installer 的 JDK 版本。



图 3-1 Oracle 登录界面

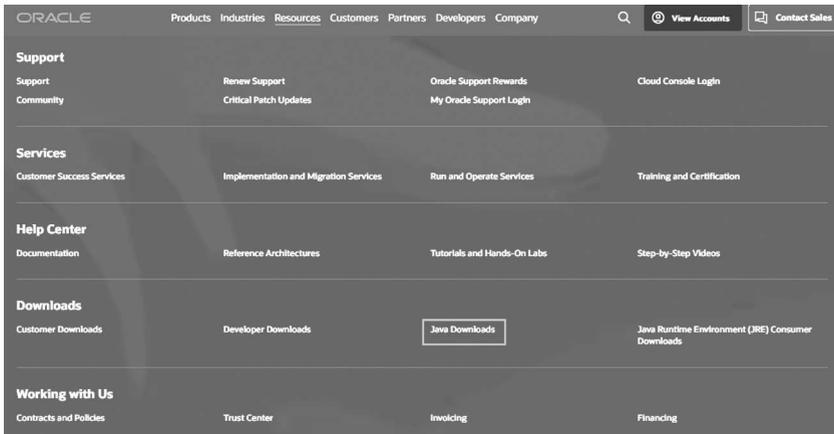


图 3-2 选择 Resources→Java Downloads 选项

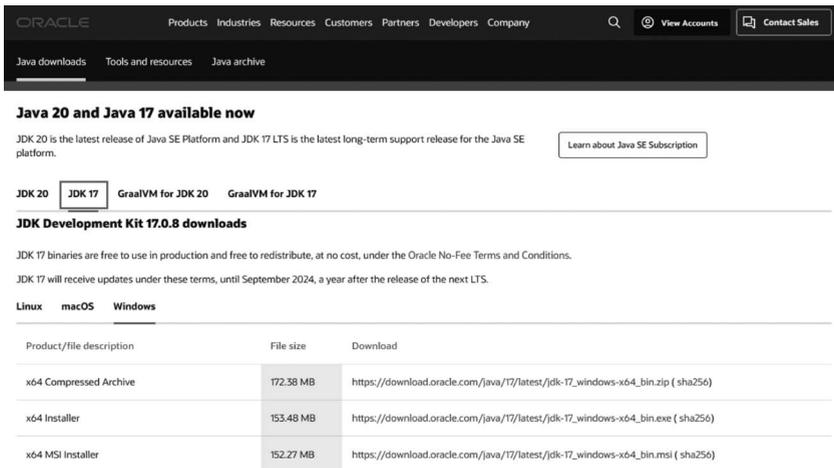


图 3-3 选择 JDK 17 安装包

单击对应版本的 JDK 文件, 直接下载即可(本书写作时 JDK 17 版本还在维护, 所以可以直接下载到 2024 年 9 月)。如果弹出以下界面, 在登录的状态下, 勾选“同意许可协议”之后就可以正常下载了(这里笔者使用 JDK 8 作为演示), 如图 3-4 所示。

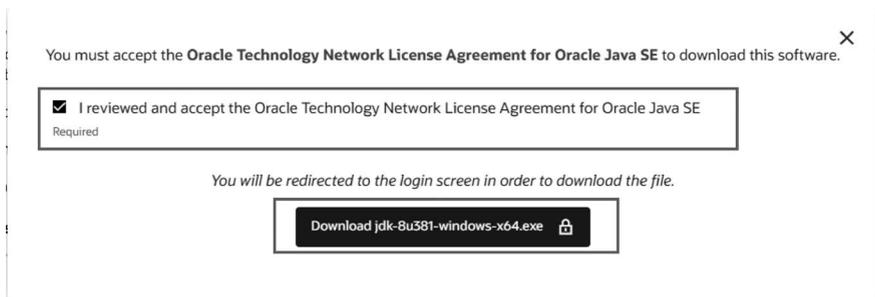


图 3-4 JDK 下载界面

3.1.3 安装 JDK

JDK 下载完成后, 双击该安装文件, 然后根据安装向导进行安装。根据页面向导的提示, 单击“下一步”按钮, 如图 3-5 所示。



图 3-5 JDK 安装向导

选择 JDK 安装的目标文件夹, 安装的目录可以进行修改, 或者保持默认路径 C:\Program Files\Java\jdk-17, 笔者直接将其安装到默认路径下。

注意: 如果选择自定义安装路径, 则安装路径的文件夹名不要包含文字和空格。

然后单击“下一步”按钮, 等待安装完成, 如图 3-6 所示。

提示安装成功后, 单击“关闭”按钮, 这时 JDK 已经安装完成, 如图 3-7 所示。

打开安装 JDK 的地址目录, 查看是否有安装信息相关文件夹。例如, 笔者选择安装在默认的路径, 所以在 C:\Program Files\Java\jdk-17 目录下就可以看到 JDK 安装的相关文件夹了, 如图 3-8 所示。



图 3-6 选择 JDK 安装的目标文件夹



图 3-7 JDK 安装完成

此电脑 > 本地磁盘 (C:) > Program Files > Java > jdk-17

名称	修改日期	类型	大小
bin	2023/9/6 23:08	文件夹	
conf	2023/9/6 23:08	文件夹	
include	2023/9/6 23:08	文件夹	
jmods	2023/9/6 23:08	文件夹	
legal	2023/9/6 23:08	文件夹	
lib	2023/9/6 23:08	文件夹	
LICENSE	2023/9/6 23:08	文件	7 KB
README	2023/9/6 23:08	文件	1 KB
release	2023/9/6 23:08	文件	2 KB

图 3-8 JDK 安装成功后生成的目录

3.1.4 配置环境变量

安装完 JDK 为什么还要配置环境变量呢？这样做主要是为了确保系统能够准确地定位和正确地使用 JDK。当在命令行或其他开发工具中执行与 Java 相关的命令时，系统需要知道 JDK 的安装路径，以便找到相应的可执行文件。通过配置环境变量，向系统提供 JDK 的安装路径信息，从而确保系统能够正确地执行与 Java 相关的命令。

如果找不到，则可以在计算机左下角的任务栏中找到“搜索”图标，并在搜索栏输入“系统环境变量”就会出现对应的搜索结果，如图 3-9 所示。

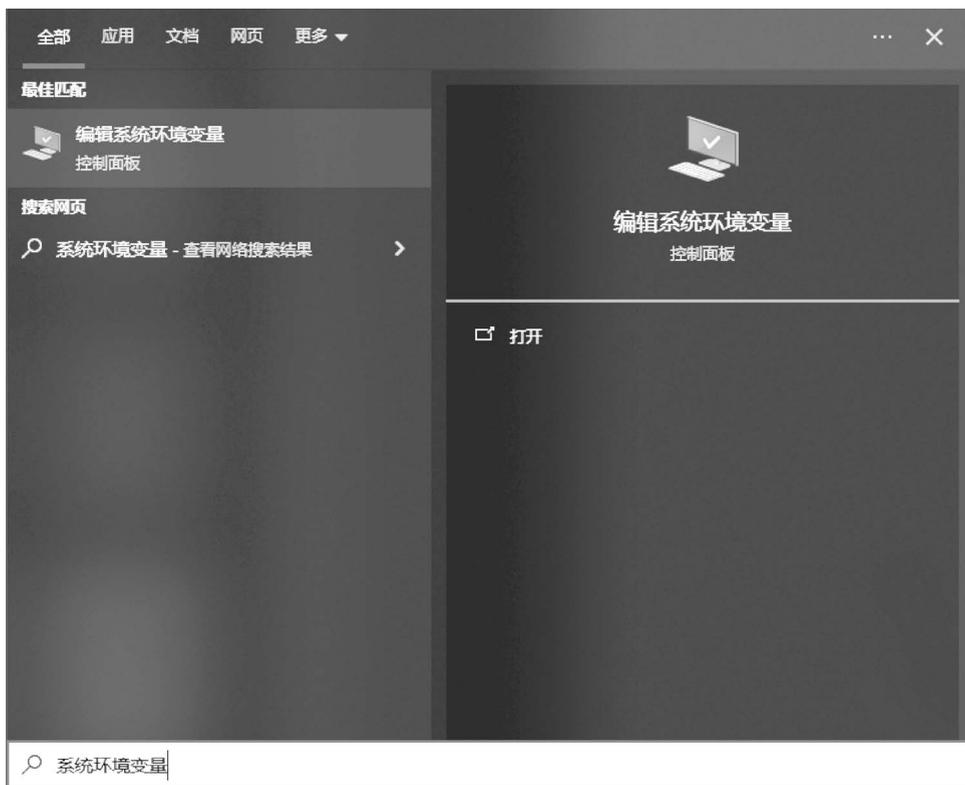


图 3-9 搜索“系统环境变量”

打开“编辑系统环境变量”窗口后，单击“环境变量”按钮，如图 3-10 所示。

打开后会看到共有上下两栏，第一栏是用户变量；第二栏是系统变量，这里要做的就是系统变量的下方新建一个系统变量。变量名输入 JAVA_HOME(这里名字全部大写)。变量值输入 JDK 安装的路径。具体内容如图 3-11 所示。

添加完成后，再次新建一个系统变量，变量名为 CLASSPATH，变量值为.；%JAVA_HOME%\lib，然后单击“确定”按钮添加完成，具体内容如图 3-12 所示。

注意：是英文格式下的点. 分号；百分号% JAVA_HOME 百分号% 反斜杠\ lib。



图 3-10 打开“系统属性”对话框

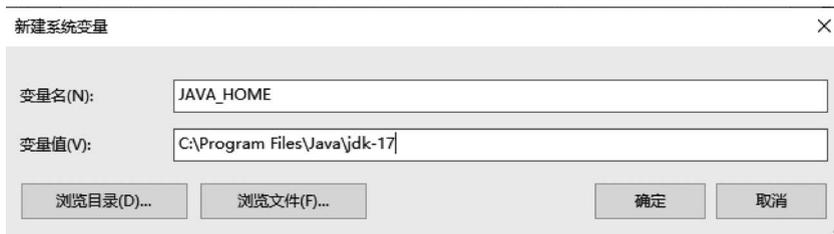


图 3-11 新建系统变量

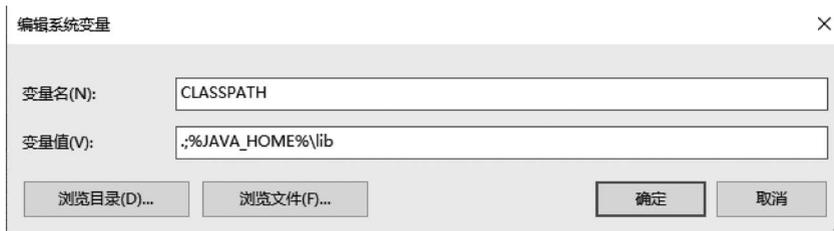


图 3-12 编辑系统变量

在系统变量中找到 Path 变量,选中 Path,单击“编辑”按钮,然后在窗口的右侧单击“新建”按钮,输入 %JAVA_HOME%\bin,最后单击“确定”按钮即可添加成功,如图 3-13 所示。

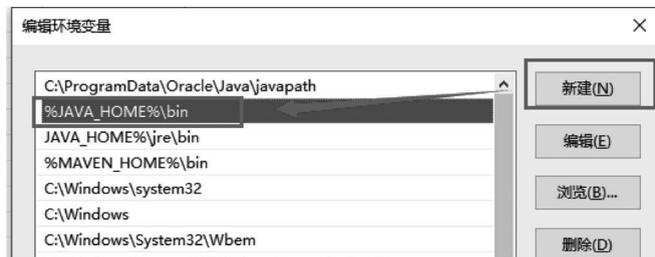


图 3-13 编辑环境变量

至此,JDK 环境变量已经配置完成。接下来测试 JDK 环境是否配置成功。按 Win+R 快捷键输入 cmd 命令,按 Enter 键,此时会弹出命令提示符窗口,然后输入如下命令:

```
java -version
```

如果环境配置正确,则在命令提示符窗口中会输出 JDK 的版本信息,如图 3-14 所示;如果执行命令后报错,则应先检查一下环境变量配置中的路径和 Path 中添加的变量是否有问题,然后去分析其他的错误原因。

```
C:\Users\Administrator>java -version
java version "17.0.8" 2023-07-18 LTS
Java(TM) SE Runtime Environment (build 17.0.8+9-LTS-211)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.8+9-LTS-211, mixed mode, sharing)
```

图 3-14 JDK 版本信息

3.1.5 JDK 和 JRE 有什么区别

JDK 和 JRE 两个有什么区别?这也是在面试时面试官会经常问到的基础题目。先来看一下 JDK 和 JRE 的定义。

(1) JDK(Java Development Kit): JDK 是用于 Java 应用程序开发的工具包。它包含了 Java 编译器(javac)、Java 虚拟机、调试器和其他开发工具,还包括了用于开发 Java 应用所需的各种类库、头文件和示例代码。JDK 适用于开发者,提供了创建、编译和调试 Java 程序的工具。

(2) JRE(Java Runtime Environment): JRE 是用于运行 Java 应用程序的环境。它包含了 Java 虚拟机和 Java 类库,用于执行 Java 程序。JRE 适用于用户端,用户可以使用 JRE 来运行 Java 应用,而不需要进行开发工作。

简而言之,JDK 是用于开发 Java 应用程序的工具包,而 JRE 是用于运行 Java 应用程序的运行环境。

3.2 IntelliJ IDEA 开发工具的安装

目前,Java 开发者主要使用的主流开发工具是 IntelliJ IDEA。此外,还有两款 Java 开发工具,分别是 Eclipse 和 MyEclipse,这两款在高校或一些初学者中使用比较多。本项目

选择使用企业主流的开发工具 IDEA,所有涉及的 Java 开发编码均采用 IDEA 开发工具。

IDEA 可以被形容为一款现代智能化的开发工具,而 Eclipse 则有些过时。IDEA 拥有强大的静态代码分析功能,能够检测代码错误、潜在问题和代码规范性问题,并提供相应的修复建议。这一特性旨在提升 Java 开发人员的工作效率和代码质量,因此,它成为许多 Java 开发者首选的 IDE 之一。

3.2.1 下载 IntelliJ IDEA

本书中的项目使用 JDK 17 的版本,则要求 IDEA 最低是 2022.1 及以上的版本,之前的 IDEA 版本不支持使用 JDK 17,所以本书使用的 IDEA 是 Ultimate 2023.1.2 的版本。

官方下载网址 <https://www.jetbrains.com/idea/>,单击 Download 按钮,下载 IntelliJ IDEA,如图 3-15 所示。

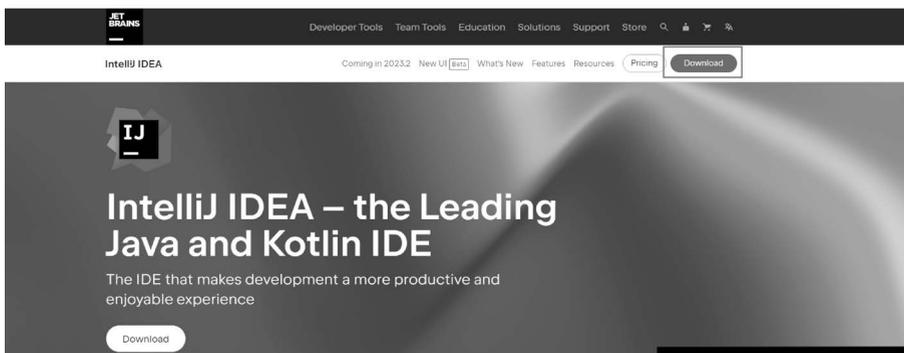
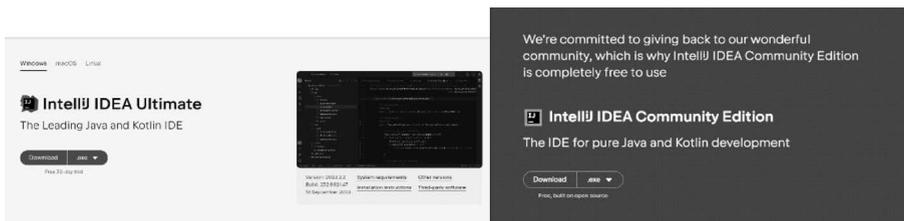


图 3-15 IntelliJ IDEA 官方首页

IDEA 官方提供了两个下载版本,一个是 IDEA 收费的 Ultimate 版本,但可以免费试用 30 天,如图 3-16(a)所示;另一个是免费的社区 Community 版本,如图 3-16(b)所示。



(a) Ultimate版本下载界面

(b) Community版本下载界面

图 3-16 不同版本下载界面

那么这两个版本有什么区别?该如何选择?

(1) IntelliJ IDEA Ultimate 版包含了全部功能,并提供了更多高级的功能和工具,如 Spring、Hibernate、Web 和企业开发等方面的全面支持,而 IntelliJ IDEA Community 版则是免费的开源版本,功能相对较少,主要关注于核心的 Java 开发功能。

(2) IntelliJ IDEA Ultimate 版支持所有插件,Community 版则只支持一部分插件。

综上所述,本项目使用 IntelliJ IDEA Ultimate 版本来编写项目代码。由于官方提供了 30 天的免费试用期,对于完成本书的项目开发基本上够用了。

3.2.2 IntelliJ IDEA 的安装

下载完成后,双击运行 .exe 安装文件,然后单击 Next 按钮,根据提供的安装导航开始安装,如图 3-17 所示。



图 3-17 IntelliJ IDEA 开始安装页面

设置 IDEA 的安装路径,默认安装在 C:\Program Files\JetBrains\IntelliJ IDEA 2023.1.2 的目录下,笔者将默认安装地址改为自定义的 D:\Software\IntelliJ IDEA 2023.1.2\目录下,然后单击 Next 按钮,如图 3-18 所示。

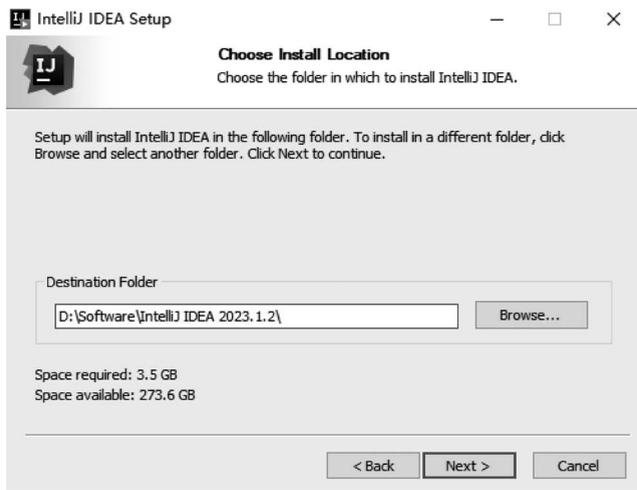


图 3-18 设置 IDEA 安装路径

勾选 IDEA 需要安装的配置项,IntelliJ IDEA 选项表示是否添加桌面图标; Add "bin" folder to the PATH 选项表示是否添加到系统环境变量; Add "Open Folder as Project" 选项表示打开文件夹作为项目; Create Associations 选项表示默认打开类型。勾选完单击 Next 按钮,如图 3-19 所示。

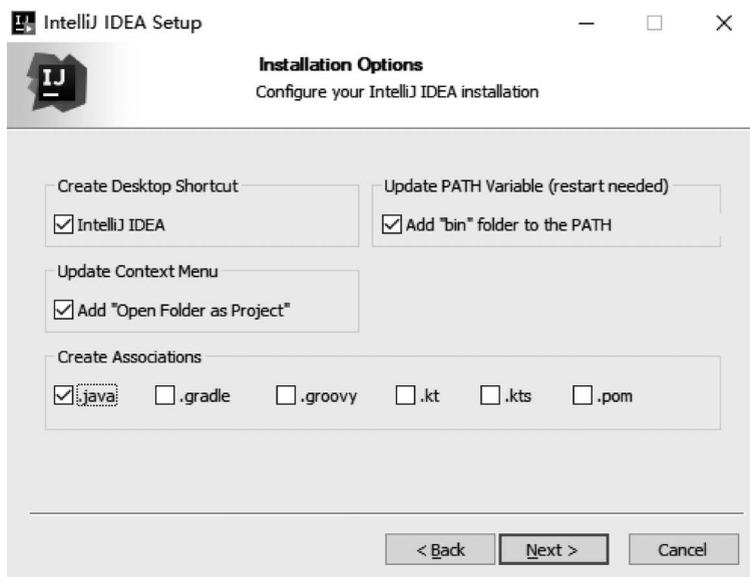


图 3-19 勾选 IDEA 安装配置项

最后,单击 Install 按钮进行安装,等待安装完成即可,如图 3-20 所示。



图 3-20 安装 IDEA

3.3 Maven 的安装与配置

Apache Maven(简称 Maven)是一个用于软件项目构建和管理的工具。Maven 通过采用标准的目录结构,使不同开发工具中的项目结构能够保持一致。它提供了一系列命令,如清理、编译、测试、安装、打包和发布等,使项目构建变得更加便捷。本书的后端项目也是选择了 Maven 作为项目依赖管理的工具。

选择 Maven 主要有以下优点。

- (1) 自动构建项目,包括清理、编译、测试、安装、打包、发布等。
- (2) JAR 包依赖管理会自动下载 JAR 及其依赖的 JAR 包。
- (3) 在多种开发工具中也能实现项目结构的统一。

3.3.1 下载 Maven

打开 Maven 官方网站 <https://maven.apache.org/>,单击 Download 按钮,如图 3-21 所示。

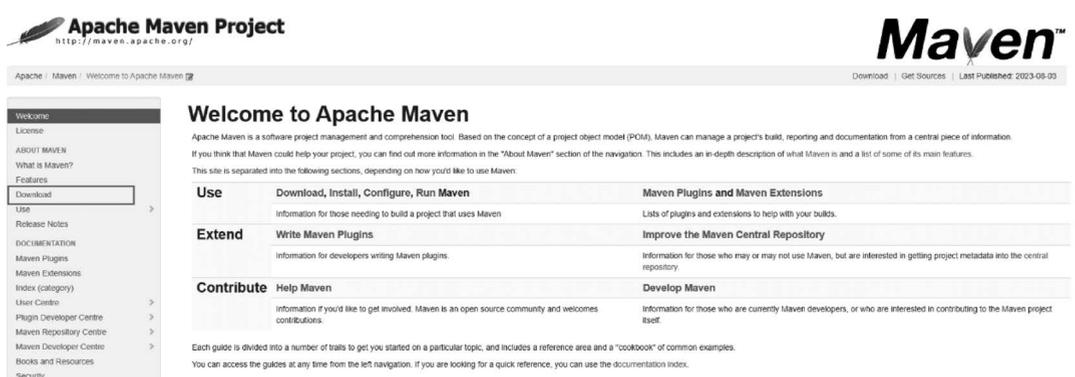


图 3-21 Maven 官方首页

目前 Maven 的最新版本是 3.9.4,因 Spring Boot 使用的版本是 3.0 以上的,所以笔者在本书中使用的 Maven 的版本为 3.6.3,可以选择历史的版本下载,如图 3-22 所示。

Other Releases

It is strongly recommended to use the latest release version of Apache Maven to take advantage of newest features and bug fixes.

If you still want to use an old version, you can find more information in the [Maven Releases History](#) and can download files from the [Maven 3 archives](#) for versions 3.0.4+ and legacy archives for earlier releases.

图 3-22 选择下载历史版本

查找到该版本,选择 binaries 目录下的 apache-maven-3.6.3-bin.zip 下载完成即可,如图 3-23 所示。

Index of /dist/maven/maven-3/3.6.3/binaries

Name	Last modified	Size	Description
Parent Directory	-	-	-
apache-maven-3.6.3-bin.tar.gz	2019-11-19 21:50	9.1M	
apache-maven-3.6.3-bin.tar.gz.asc	2019-11-19 21:50	235	
apache-maven-3.6.3-bin.tar.gz.sha512	2019-11-19 21:50	128	
apache-maven-3.6.3-bin.zip	2019-11-19 21:50	9.2M	
apache-maven-3.6.3-bin.zip.asc	2019-11-19 21:50	235	
apache-maven-3.6.3-bin.zip.sha512	2019-11-19 21:50	128	

图 3-23 选择 Maven 安装包

3.3.2 安装配置 Maven

下载完成后无须安装,直接对下载的压缩包进行解压,然后将文件存放到硬盘中即可,例如笔者放在了 D:\apache-maven-3.6.3 目录下,如图 3-24 所示。

名称	修改日期	类型	大小
bin	2023/9/19 15:10	文件夹	
boot	2023/9/19 15:10	文件夹	
conf	2023/9/19 15:10	文件夹	
lib	2023/9/19 15:10	文件夹	
LICENSE	2019/11/7 12:32	文件	18 KB
NOTICE	2019/11/7 12:32	文件	6 KB
README.txt	2019/11/7 12:32	文本文档	3 KB

图 3-24 Maven 文件目录

接下来配置 Maven 的环境变量,这里需要注意的是,配置 Maven 环境变量之前要确保 JDK 环境的配置没有问题。和之前配置 JDK 环境变量基本一致。先创建一个系统变量,变量名为 MAVEN_HOME(这里的字母全部大写),变量值为 Maven 存放的路径 D:\apache-maven-3.6.3。填写完成后,单击“确定”按钮,保存系统变量,如图 3-25 所示。

编辑系统变量

变量名(N): MAVEN_HOME

变量值(V): D:\apache-maven-3.6.3

浏览目录(D)... 浏览文件(F)... 确定 取消

图 3-25 Maven 编辑系统变量

在系统变量中选中 Path,然后单击“编辑”按钮,新建一个 Maven 的变量,配置 Maven 的 bin 目录,添加的配置如下:

```
% MAVEN_HOME % \bin
```

配置完成后,打开命令提示符窗口,输入 `mvn -v` 命令,查看 Maven 版本信息。如果配置正确,则会出现版本、安装地址等信息;如果没有显示图 3-26 所示的信息,则首先需要检查配置的环境变量是否有问题,其次查看下载的 Maven 包是否完整,如图 3-26 所示。

```
C:\Users\admin>mvn -v
Apache Maven 3.6.3 (cecedd343002696d0abb50b32b541b8a6ba2883f)
Maven home: D:\xyh\maven\apache-maven-3.6.3\bin\..
Java version: 17.0.8, vendor: Oracle Corporation, runtime: C:\Program Files\Java\jdk-17
Default locale: zh_CN, platform encoding: GBK
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"
```

图 3-26 Maven 安装验证

3.3.3 Maven 的相关配置

在使用 Maven 下载项目依赖文件时,首先它会检查本地仓库是否已经存在所需的依赖包,如果没有,则会尝试从中央仓库获取。然而,中央仓库通常位于国外服务器,导致下载速度比较慢,甚至可能导致下载失败,接下来就解决这个问题。

1. 配置本地仓库

Maven 默认的仓库下载地址是在 C 盘中,但一般不推荐使用 C 盘存放本地仓库,所以在其他硬盘中创建一个文件夹用来当作 Maven 的本地仓库文件。例如,笔者将仓库的默认地址改为 `D:\maven\maven_repository`。

本地仓库其实起到了一个缓存的作用,它的默认地址是 `C:\Users\用户名.m2`。现在要修改成自定义的仓库文件,进入 Maven 的安装目录,在 `conf` 文件夹中打开 `settings.xml` 配置文件,在文件中找到 `localRepository` 标签,`localRepository` 节点是用于配置本地仓库,将创建的仓库地址添加到配置文件中,代码如下:

```
<localRepository>D:\maven\maven_repository</localRepository>
```

2. 配置中央仓库

为了解决下载依赖慢的问题,要对 Maven 配置进行修改,将默认的中央仓库换成阿里云的中央仓库或者华为云的中央仓库,需要修改 Maven 在配置文件中的 `mirrors` 标签来配置镜像仓库。

本书以阿里云镜像仓库为例,打开 Maven 的 `settings.xml` 配置文件,添加阿里云仓库镜像的配置,需要添加在 `<mirrors></mirrors>` 标签中,`mirrors` 可以配置多个子节点,但是它只会使用其中的一个节点生效,即在默认情况下,如果配置多个 `mirror`,则只有第 1 个生效,代码如下:

```
<!-- 阿里云仓库 -->
<mirror>
  <id>nexus - aliyun</id>
  <mirrorOf>central</mirrorOf>
  <name>Nexus aliyun</name>
  <url>http://maven.aliyun.com/nexus/content/groups/public</url>
</mirror>
```

3. 配置 JDK 版本

如果要在 Maven 中设置 JDK 环境,则需要在 settings.xml 配置文件中的 profiles 标签中添加代码配置,代码如下:

```
<!-- java 版本 -->
<profile>
  <id>jdk-17</id>
  <activation>
    <activeByDefault>true</activeByDefault>
    <jdk>17</jdk>
  </activation>
  <properties>
    <maven.compiler.source>17</maven.compiler.source>
    <maven.compiler.target>17</maven.compiler.target>
    <maven.compiler.compilerVersion>17</maven.compiler.compilerVersion>
  </properties>
</profile>
```

配置完成后,打开命令提示符窗口,输入 mvn help:system 命令,如果第 1 次执行该命令,则在执行命令后会从 Maven 仓库下载一些必要的插件,下载完成后就会显示有关 Maven 系统的信息,如图 3-27 所示。

```
Microsoft Windows [版本 10.0.18363.1500]
(c) 2019 Microsoft Corporation. 保留所有权利。

C:\Users\70423>mvn help:system
[INFO] Scanning for projects...
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.pom (3.9 kB at 2.5 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/22/maven-plugins-22.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/22/maven-plugins-22.pom (13 kB at 15 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/21/maven-parent-21.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/21/maven-parent-21.pom (26 kB at 27 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/apache/10/apache-10.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/apache/10/apache-10.pom (15 kB at 30 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.5/maven-clean-plugin-2.5.jar (25 kB at 24 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-install-plugin/2.4/maven-install-plugin-2.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-install-plugin/2.4/maven-install-plugin-2.4.pom (6.4 kB at 15 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/23/maven-plugins-23.pom
```

图 3-27 Maven 相关信息

到此,Maven 安装和配置就结束了,接下来还需要完成 MySQL 数据库的安装与配置及 Navicat 工具的安装。

3.4 MySQL 的安装与配置

本书中的项目使用的数据库是 MySQL,MySQL 是目前最流行的关系数据库管理系统,在 Web 应用方面 MySQL 是最好的关系数据库管理系统应用软件之一。项目使用的是

MySQL 8 以上的版本,本项目使用的是 MySQL 8.0.34 版本。

3.4.1 下载 MySQL

打开 MySQL 官方网站 <https://dev.mysql.com/downloads/mysql/>, 单击 General Availability(GA)Releases 按钮,在 Select Version 中选择下载 MySQL 的版本;并在 Select Operating System 中选择下载的操作系统,然后单击 Go to Download Page 按钮,跳转到下载页面,如图 3-28 所示。

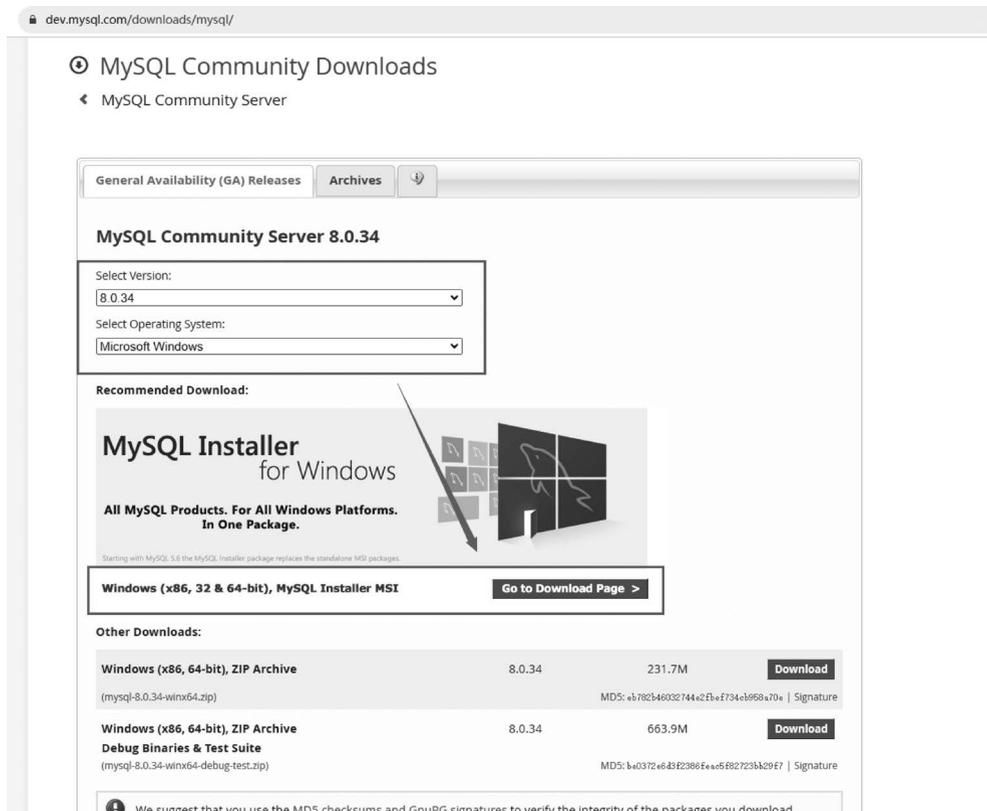


图 3-28 MySQL 选择下载版本

然后选择 mysql-installer-community-8.0.34.0.msi 安装包,单击 Download 按钮进行下载,如图 3-29 所示。

单击 No thanks, just start my download. 协议后,开始下载并安装包,如图 3-30 所示。

下载完成后,双击下载的安装包,在安装首页勾选 Custom 选项,即修改成自定义安装,然后单击 Next 按钮,进行下一步操作,如图 3-31 所示。

选择要安装的产品,将左侧选择框中的树结构展开,单击 MySQL Server 8.0.34 -X64,然后单击中间向右的箭头,将其添加到右边待安装区,选择完成后,单击 Next 按钮,如图 3-32 所示。

MySQL Community Downloads

MySQL Installer

General Availability (GA) Releases Archives

MySQL Installer 8.0.34

Note: MySQL 8.0 is the final series with MySQL Installer. As of MySQL 8.1, use a MySQL product's MSI or Zip archive for installation. MySQL Server 8.1 and higher also bundle MySQL Configurator, a tool that helps configure MySQL Server.

Select Version:
8.0.34

Select Operating System:
Microsoft Windows

Windows (x86, 32-bit), MSI Installer (mysql-installer-web-community-8.0.34.0.msi)	8.0.34	2.4M	Download
Windows (x86, 32-bit), MSI Installer (mysql-installer-community-8.0.34.0.msi)	8.0.34	331.3M	Download

We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.

图 3-29 选择下载并安装包

dev.mysql.com/downloads/file/?id=520407

MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

[Login »](#)
using my Oracle Web account

[Sign Up »](#)
for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can sign up for a free account by clicking the Sign Up link and following the instructions.

[No thanks, just start my download.](#)

ORACLE © 2023 Oracle

[Privacy / Do Not Sell My Info](#) | [Terms of Use](#) | [Trademark Policy](#) | [Cookie 喜好设置](#)

图 3-30 下载 MySQL 安装包

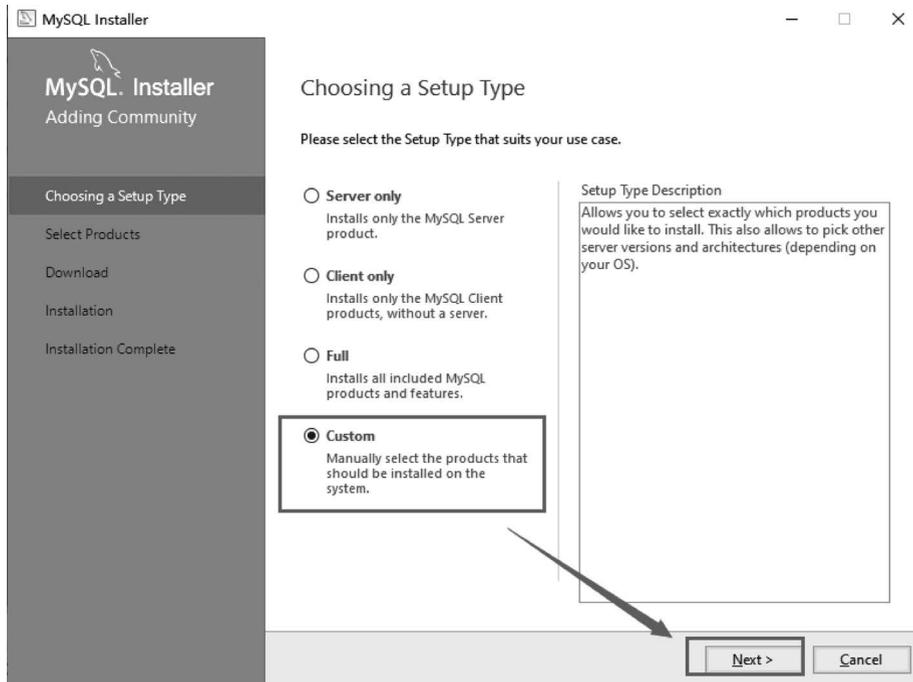


图 3-31 选择 MySQL 安装方式

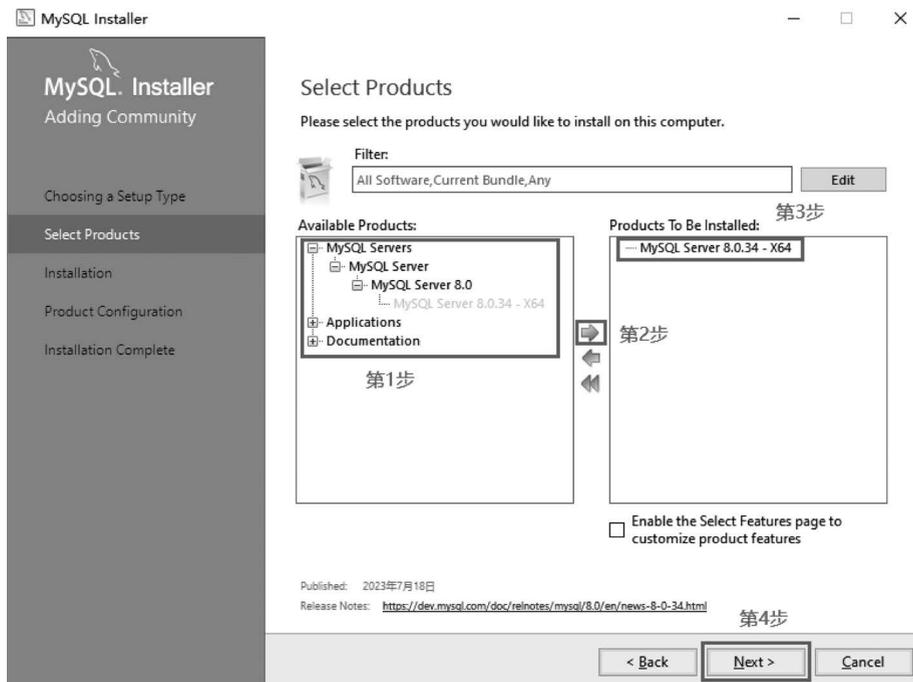


图 3-32 选择要安装的 MySQL

接下来,选择 MySQL 安装目录,安装的路径不要有中文名称出现。例如,笔者将 MySQL 的安装路径修改为 D:\Software\MySQL8.0.34,选择完成后,单击 Next 按钮,如图 3-33 所示。

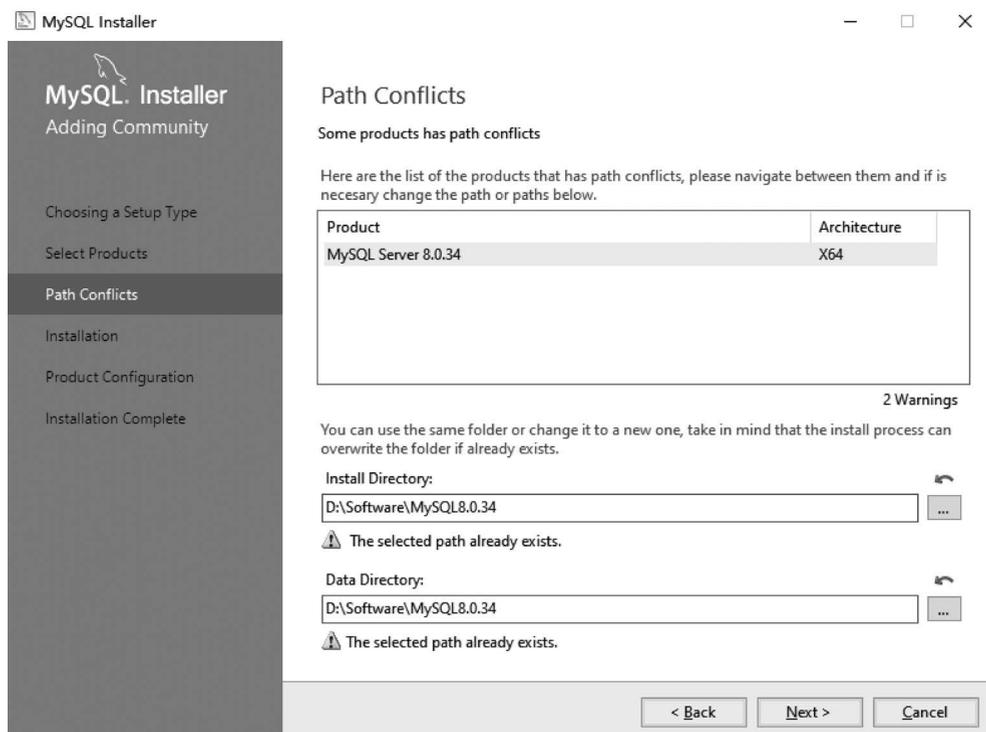


图 3-33 选择安装路径

接下来根据 MySQL 安装向导,依次单击 Next 按钮或 Execute 按钮安装相关环境。执行到配置 MySQL 端口的界面时,Port 默认为 3306 端口,其余的配置默认不变,单击 Next 按钮,如图 3-34 所示。

接下来的步骤依次单击 Next 按钮往下执行,直到提示安装完成即可。

3.4.2 配置 MySQL

安装完成后,打开系统环境变量,在系统变量的 Path 中添加安装 MySQL 的路径,这个路径要配置到 MySQL 路径下的 bin 目录。如果安装时选择的是默认安装路径,则目录为 C:\Program Files\MySQL\MySQL Server 8.0,添加完成后,单击“确定”按钮,保存成功,如图 3-35 所示。

3.4.3 验证配置

打开命令提示符窗口,执行的命令如下:

```
mysql -u root -p
```

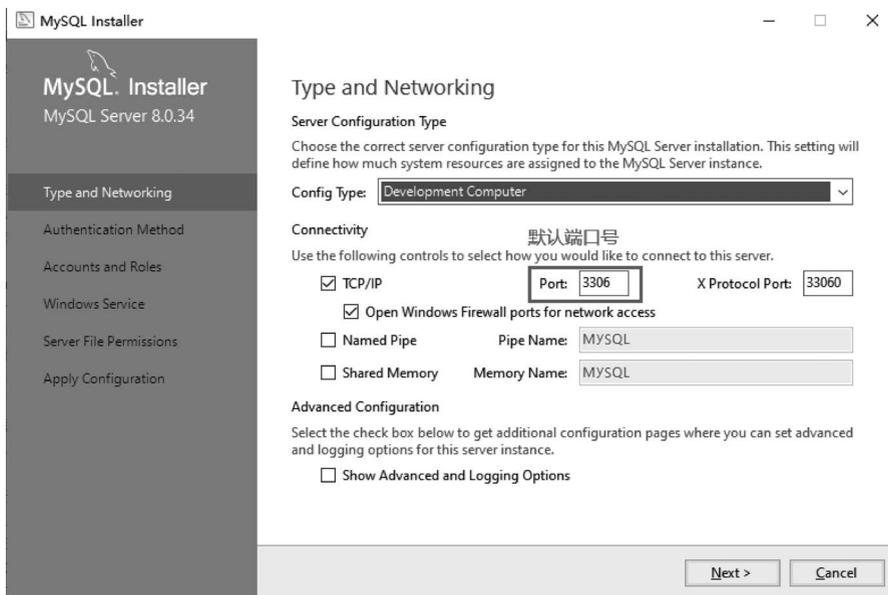


图 3-34 设置 MySQL 端口号

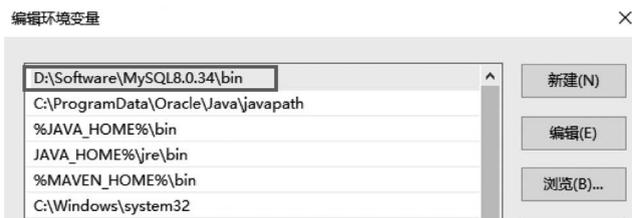


图 3-35 编辑环境变量

执行该命令后,显示要输入 MySQL 密码,该密码是安装数据库时设置的密码,输入密码后按 Enter 键执行。如果出现图中的 Welcome to the MySQL monitor. 及数据库版本等信息就说明已经配置成功,如图 3-36 所示。

```
C:\WINDOWS\system32\cmd.exe - mysql -u root -p
Microsoft Windows [版本 10.0.19045.3208]
(c) Microsoft Corporation。保留所有权利。

C:\Users\admin>mysql -u root -p
Enter password: ***** 安装MySQL时设置的密码
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.34 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

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>
```

图 3-36 执行命令后进入 MySQL

3.5 MySQL 可视化工具安装

MySQL 已成功安装并运行,但每次操作数据库都需通过命令行方式进入 MySQL,然后使用命令进行数据操作。对技术人员而言,这种方式过于烦琐,因此,需要安装一款连接数据库的可视化工具,以便轻松进行数据库操作。

在丰富的可视化工具中,可供选择的主要有 DataGrip、DBeaver、Navicat for MySQL (简称: Navicat)及 MySQL 官方的 MySQL Workbench,本项目选择了流行且广泛应用的 Navicat。Navicat 界面友好、功能强大,可用于多种数据库管理任务。它能通过直观的图形界面连接数据库、执行查询、管理数据,并支持数据库设计等任务,大幅减少了单调的命令行输入。

3.5.1 下载 Navicat for MySQL

打开 Navicat for MySQL 官网 <https://www.navicat.com.cn/products>,可以直接下载目前最新版本 Navicat Premium 16,它可以从单一应用程序中同时连接 MySQL、Redis、MariaDB、MongoDB、SQL Server、Oracle、PostgreSQL 和 SQLite 等,功能比较全面。本书选择 Navicat Premium 16 版本来连接数据库。

因为 Navicat 是收费的工具,所以优先选择免费试用,然后根据计算机的配置进行选择下载并安装包。下载完成后直接安装,如图 3-37 所示。

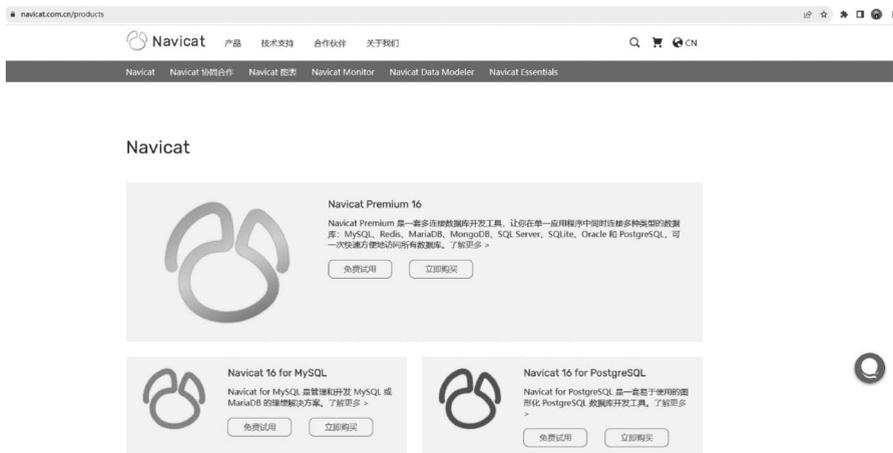


图 3-37 Navicat 官网下载界面

3.5.2 连接 MySQL

Navicat 安装完成后,打开 Navicat 工具,单击左上角的“连接”图标,选择 MySQL 选项,如图 3-38 所示。



图 3-38 新建数据库连接

选择完成后,需要填写 MySQL 连接信息,这里的密码和端口号都是安装 MySQL 时配置的。填写完信息后,单击左下角的“测试连接”按钮,如果弹出连接成功窗口,则说明填写的信息是正确的,然后单击“确定”按钮,新建连接成功,如图 3-39 所示。



图 3-39 新建连接

本章小结

本章着重介绍了项目开发前所需的环境配置工作。通过本章的介绍,目前已经掌握了以下内容。

(1) JDK 的重要性不言而喻,JDK 作为 Java 开发所必不可少的工具包,为项目提供了必要的核心库和工具,它确保能够编写、编译和运行 Java 代码。

(2) 本书选择 IDEA 作为项目的开发工具,IDEA 提供了丰富的功能和集成开发环境,有助于提高开发效率。

(3) 了解到如何使用 Maven 来管理项目的依赖关系。Maven 能够自动下载并管理所需的库和框架,使项目的依赖管理更加便捷。

(4) 目前已经安装了 MySQL 数据库并配置成功,同时选择了 Navicat 等可视化工具来方便地操作数据库。

接下来,将迈入项目的正式开发阶段。在这一阶段,将能够运用所搭建的环境,开始编写代码、构建应用程序,并逐步实现项目的各项功能和特性。