

Java 开发环境配置

试验目的：

- 1) 掌握 Java 虚拟机的安装与配置；
- 2) 掌握 Eclipse 的基本使用；
- 3) 掌握 MySQL 数据库服务器的配置与使用；
- 4) 掌握程序的导出与运行。

实验步骤：

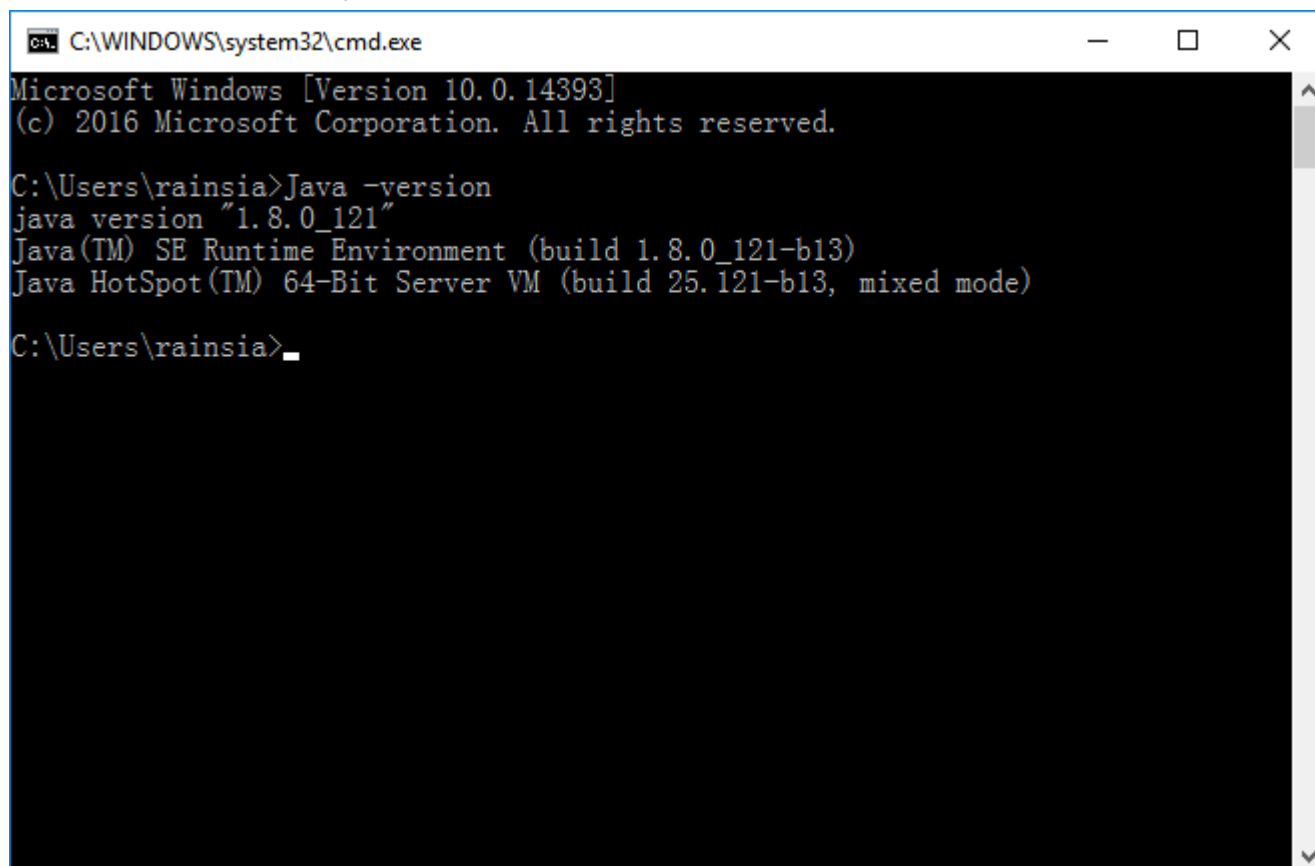
【任务一】：安装 JDK。

步骤：

1. 双击 JDK 安装文件。
2. 按照提示安装 JDK 到 D 盘，并记录下安装路径，例如 d:\Program files\Oracle\JDK\。（注意：安装过程中会出现两次路径选择提示，第一次为 JDK，第二次为 JRE，两个路径不能相同）。
3. 配置环境变量（注意大小写，此外 JAVA_HOME 请使用上面记录的路径，如果 JAVA_HOME 和 CLASSPATH 在原系统中出现过，请先删除原值）：

JAVA_HOME	d:\Program files\Oracle\JDK\
Path	%JAVA_HOME%\bin;%JAVA_HOME%\jre\bin
CLASSPATH	.;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar

4. 打开命令提示符，输入 `java -version` 查看是否输出正确的版本信息，如果成功，则 JRE 配置成功：

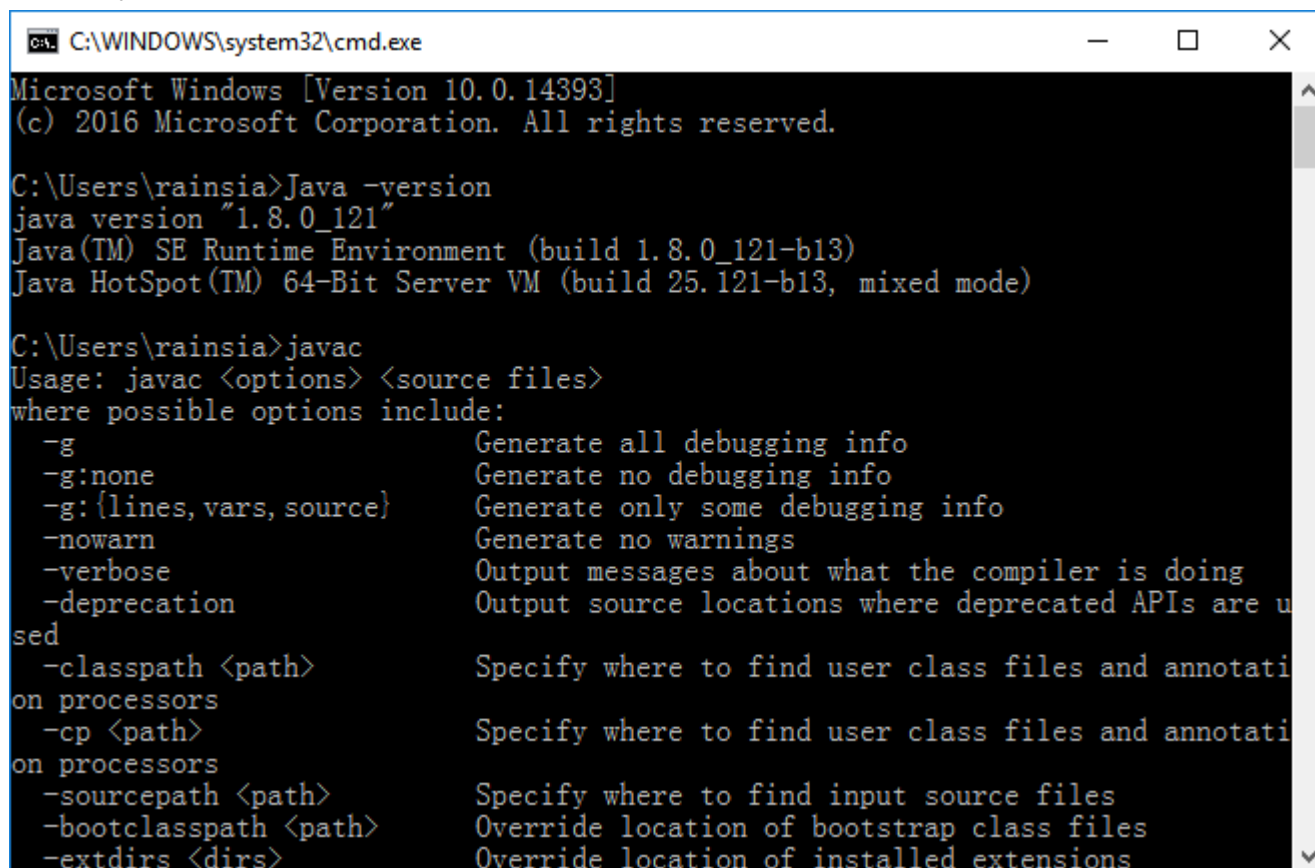


```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\rainsia>Java -version
java version "1.8.0_121"
Java(TM) SE Runtime Environment (build 1.8.0_121-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.121-b13, mixed mode)

C:\Users\rainsia>
```

5. 输入 javac 看是否输出编译信息，若输入，则 JDK 配置成功：



```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

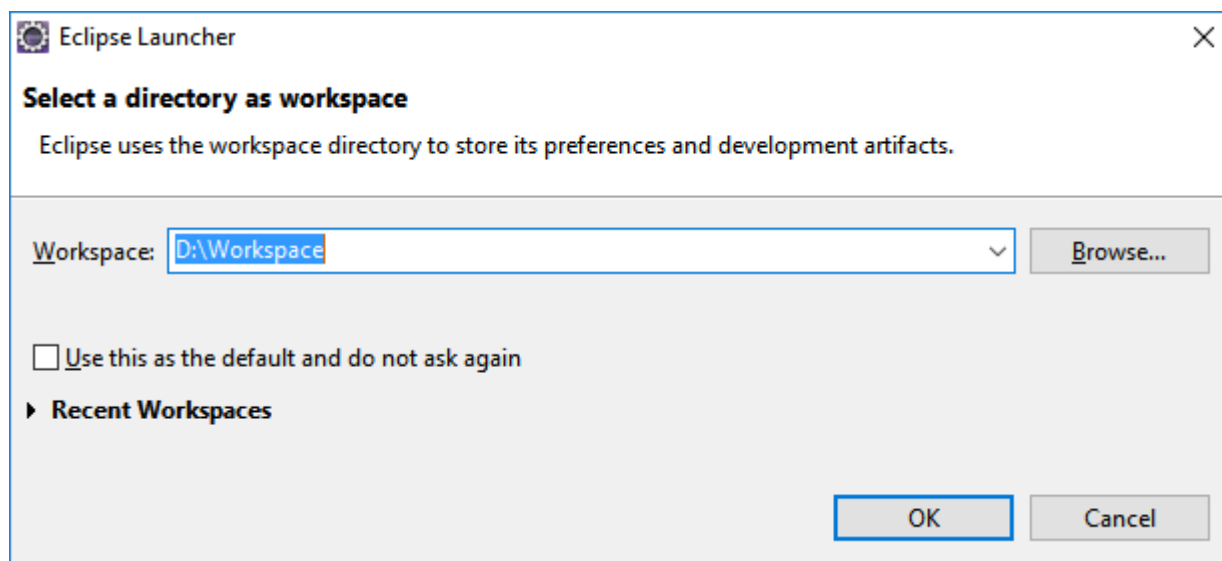
C:\Users\rainsia>Java -version
java version "1.8.0_121"
Java(TM) SE Runtime Environment (build 1.8.0_121-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.121-b13, mixed mode)

C:\Users\rainsia>javac
Usage: javac <options> <source files>
where possible options include:
  -g               Generate all debugging info
  -g:none          Generate no debugging info
  -g:{lines,vars,source}  Generate only some debugging info
  -nowarn          Generate no warnings
  -verbose         Output messages about what the compiler is doing
  -deprecation     Output source locations where deprecated APIs are used
  -classpath <path>  Specify where to find user class files and annotations processors
  -cp <path>        Specify where to find user class files and annotations processors
  -sourcepath <path> Specify where to find input source files
  -bootclasspath <path> Override location of bootstrap class files
  -extdirs <dirs>   Override location of installed extensions
  
```

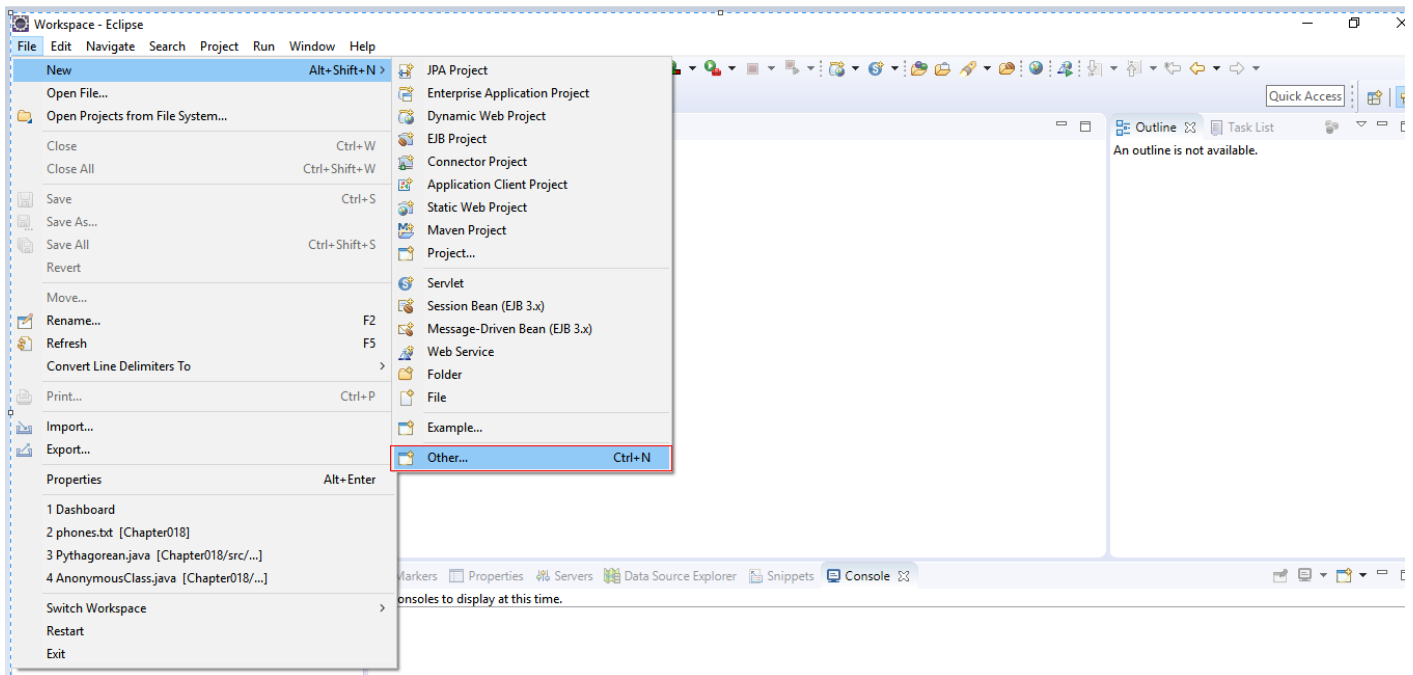
【任务二】：在 Eclipse 中创建 Java Web 工程。

步骤：

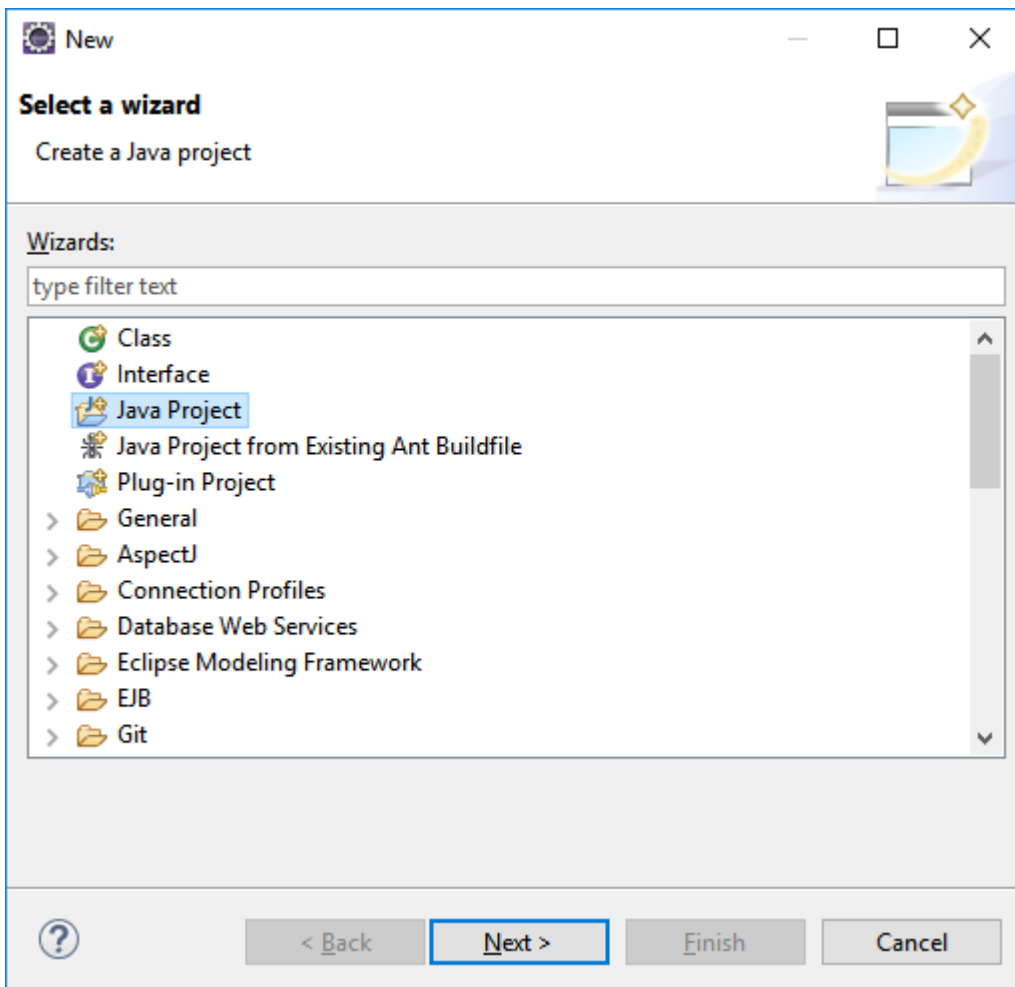
1. 将提供的 Eclipse 文件解压到 D 盘。
2. 启动 Eclipse 文件，将询问 Eclipse 工作区：



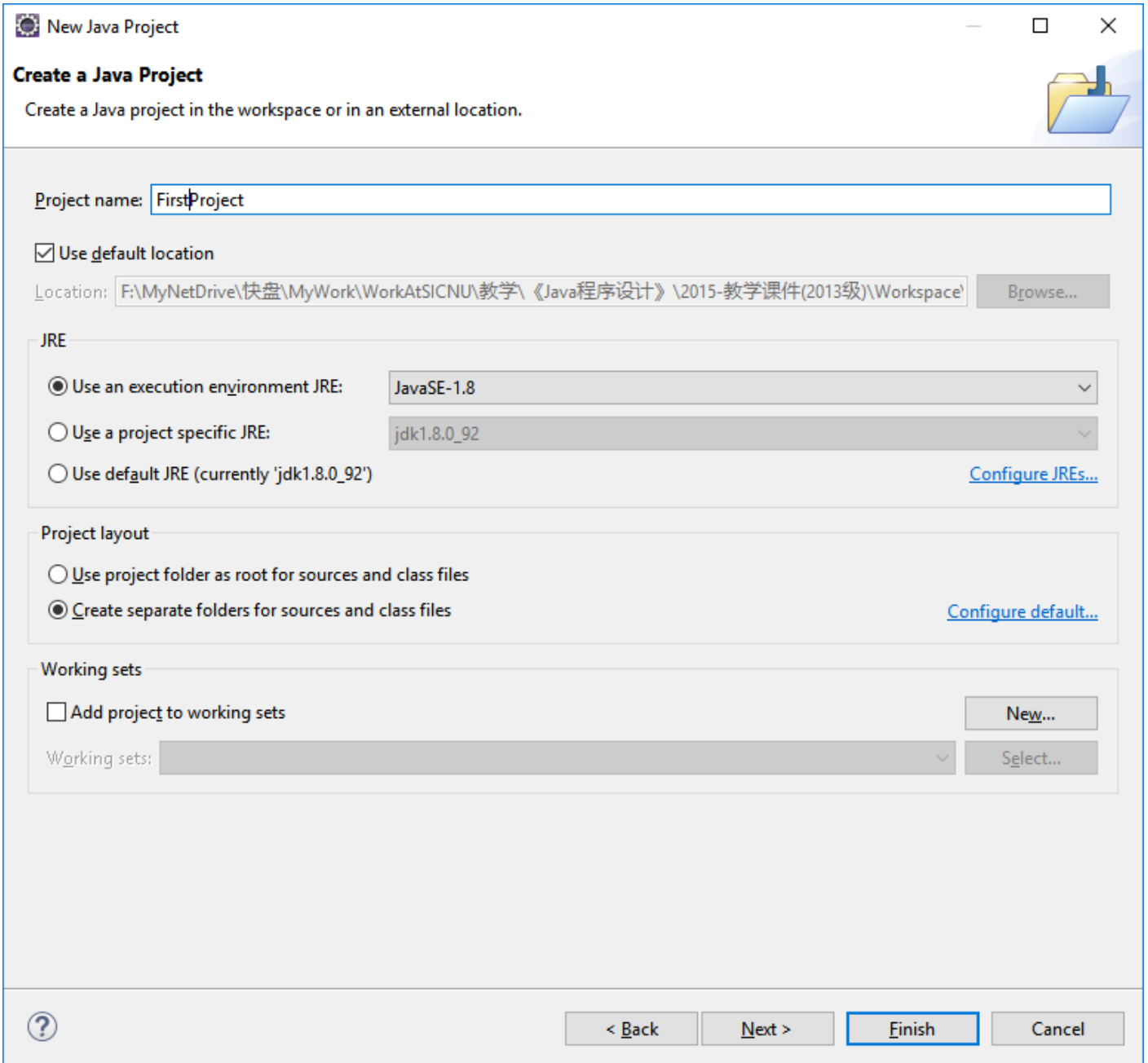
3. 将 Eclipse 工作区设置到 D 盘（或者其他 C 盘之外的盘），例如：D:\Workspace 下。然后点击“OK”按钮。注：工作区是指所有工程存放的目录。
4. 选中 Eclipse 菜单项“New”→“Other”：



5. 在弹出框当中选择 Java Project，点击 Next。



6. 在新建项目界面中 project name 处输入工程名称 FirstProject。并点击 Finish。



New Java Project

Create a Java project in the workspace or in an external location.

Project name:

☒ Use default location

Location:

JRE

☒ Use an execution environment JRE:

☐ Use a project specific JRE:

☐ Use default JRE (currently 'jdk1.8.0_92') [Configure JREs...](#)

Project layout

☐ Use project folder as root for sources and class files

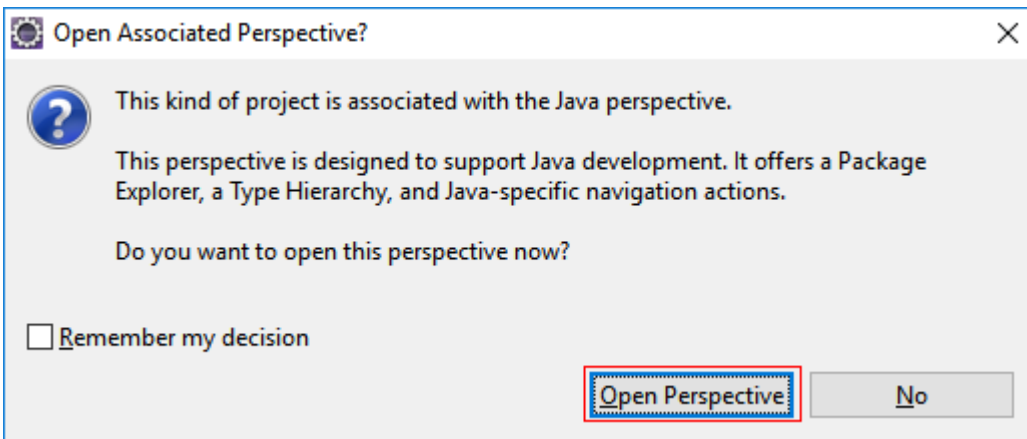
☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

☐ Add project to working sets

Working sets:

7. 若弹出对话框让选择是否进入 Java 工程视图，则回答是进入 Java 工程视图。



Open Associated Perspective?

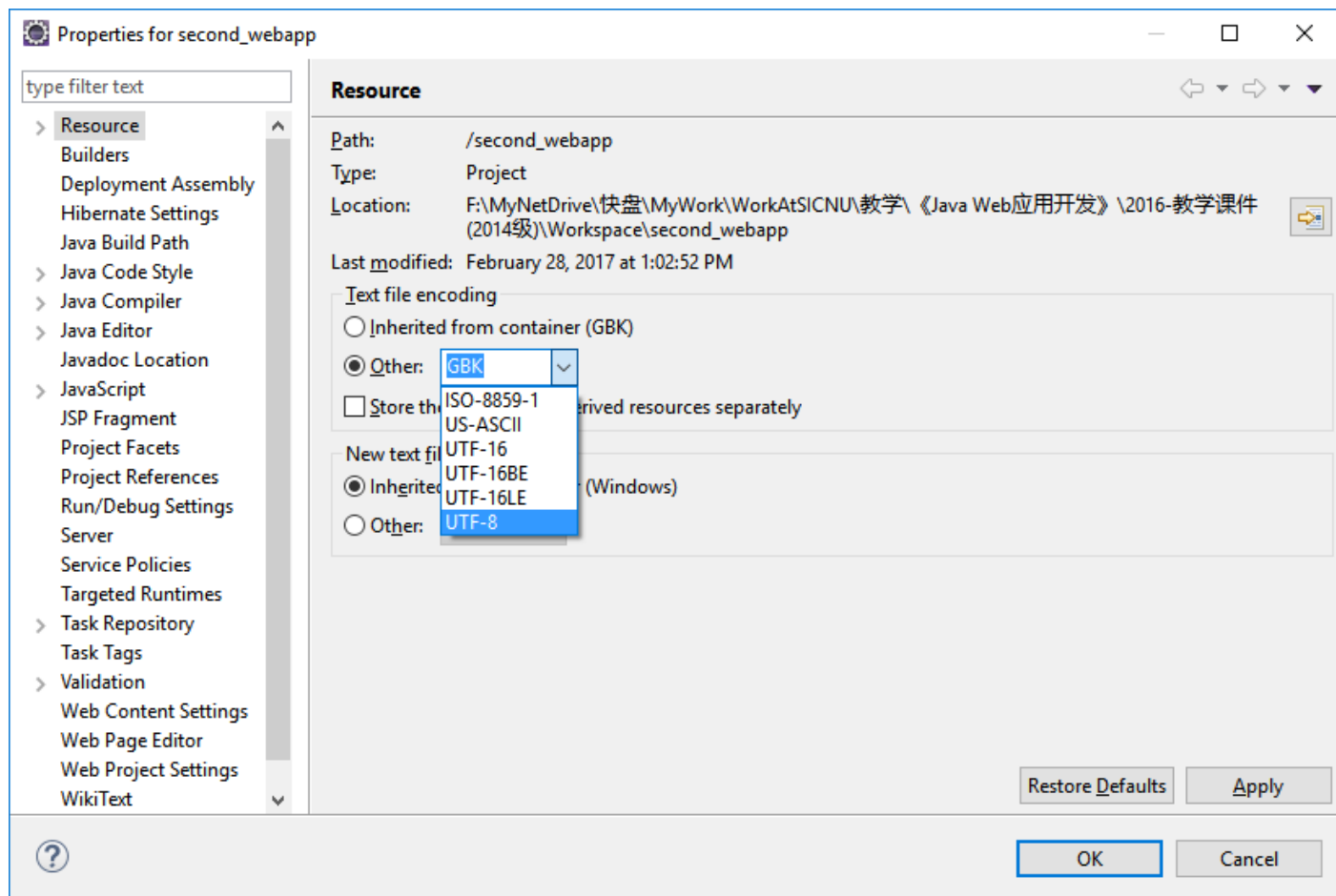
This kind of project is associated with the Java perspective.

This perspective is designed to support Java development. It offers a Package Explorer, a Type Hierarchy, and Java-specific navigation actions.

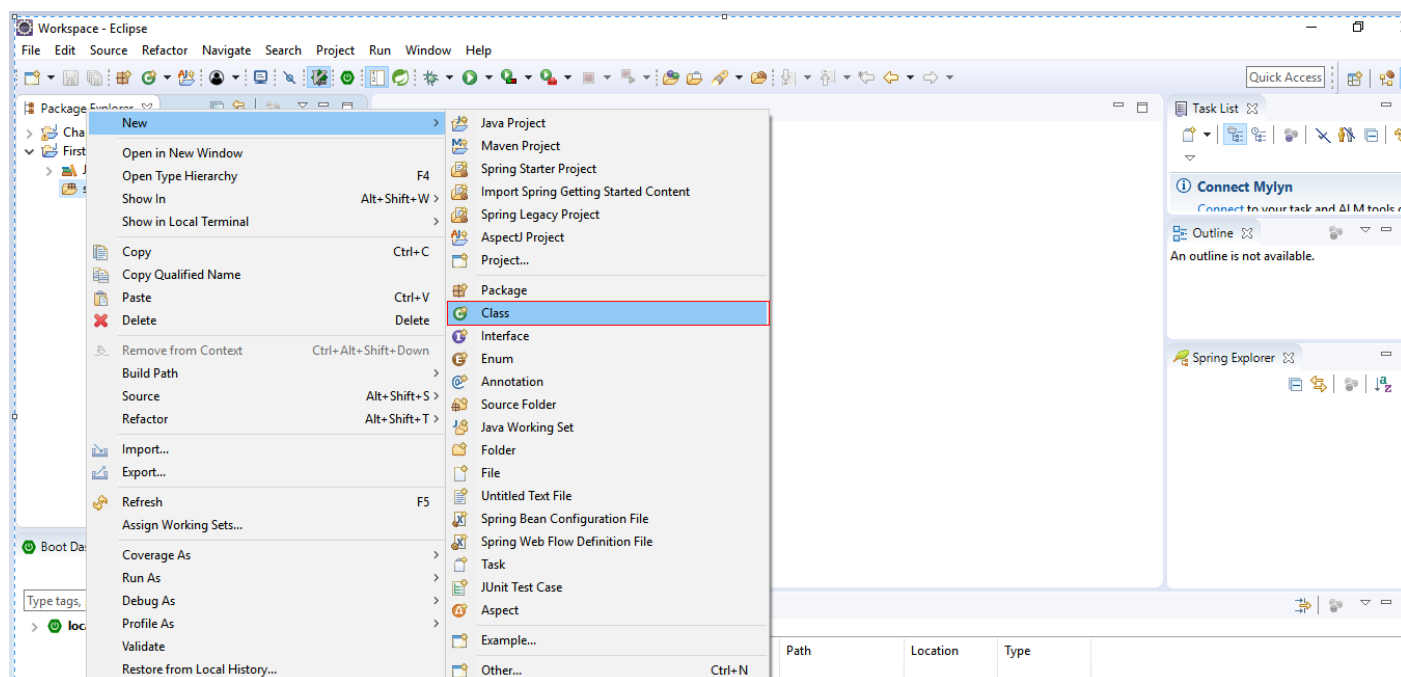
Do you want to open this perspective now?

☐ Remember my decision

8. 在工程上点击右键，并选中“Properties”菜单项。弹出工程属性对话框。
9. 若看到工程默认编码为 GBK，则在对话框中将工程的编码方式改为 UTF-8，目的是避免以后中文出现乱码（以后作业或者考试，**均必须使用 UTF-8 作为工程编码，切记！！**）：



10. 展开工程，在 src 目录上点击鼠标右键，选择 new->Class:



11. 然后输入类的名字，并点击 Finish:

New Java Class

Java Class

⚠ The use of the default package is discouraged.

Source folder: FirstProject/src Browse...

Package: (default) Browse...

☐ Enclosing type: Browse...

Name: HelloWorld

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static

Superclass: java.lang.Object Browse...

Interfaces: Add... Remove

Which method stubs would you like to create?

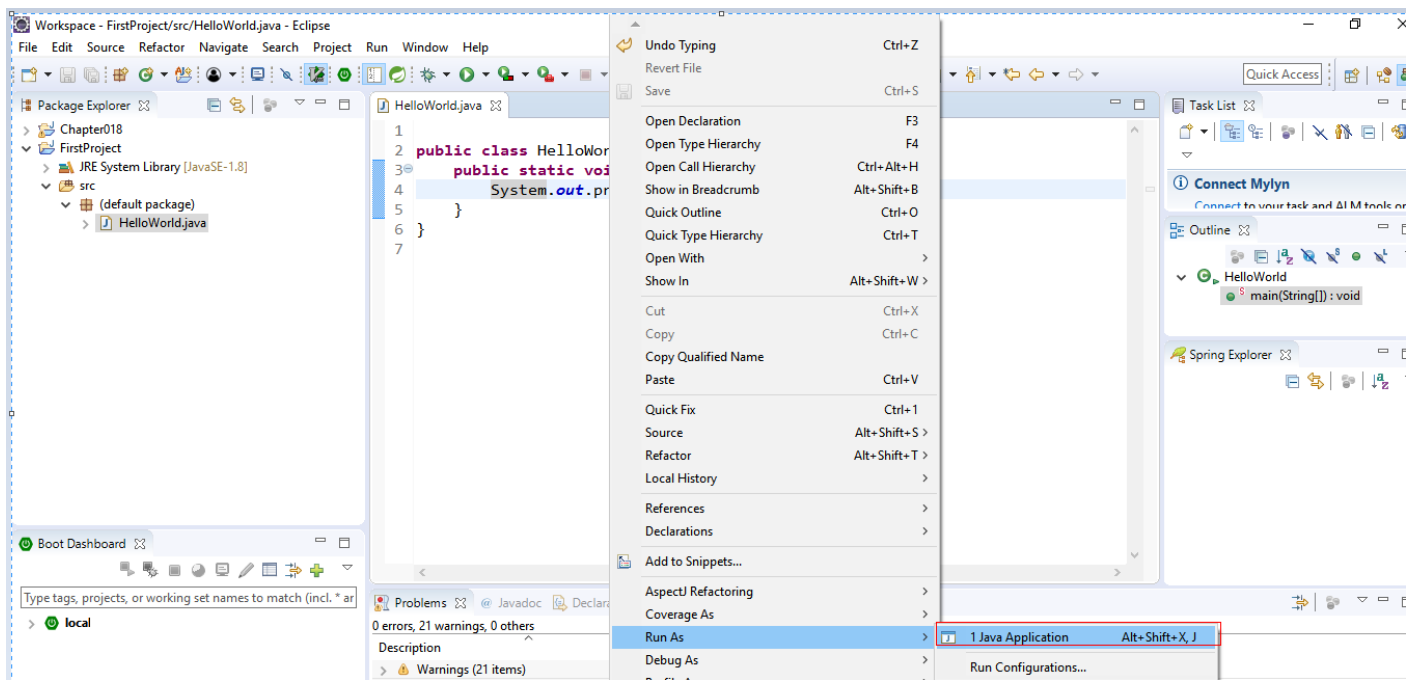
☐ public static void main(String[] args)
☐ Constructors from superclass
☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))
☐ Generate comments

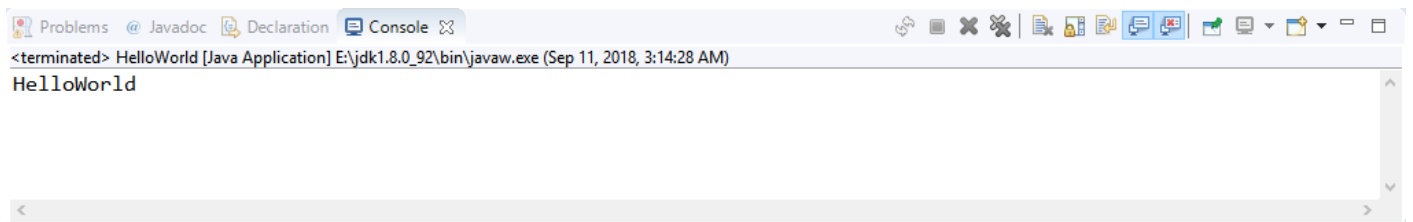
Finish Cancel

12. 此时，程序自动生成类的框架，完成 HelloWorld 程序。

13. 在程序编辑区点击右键，选择 Run As->Java Application。



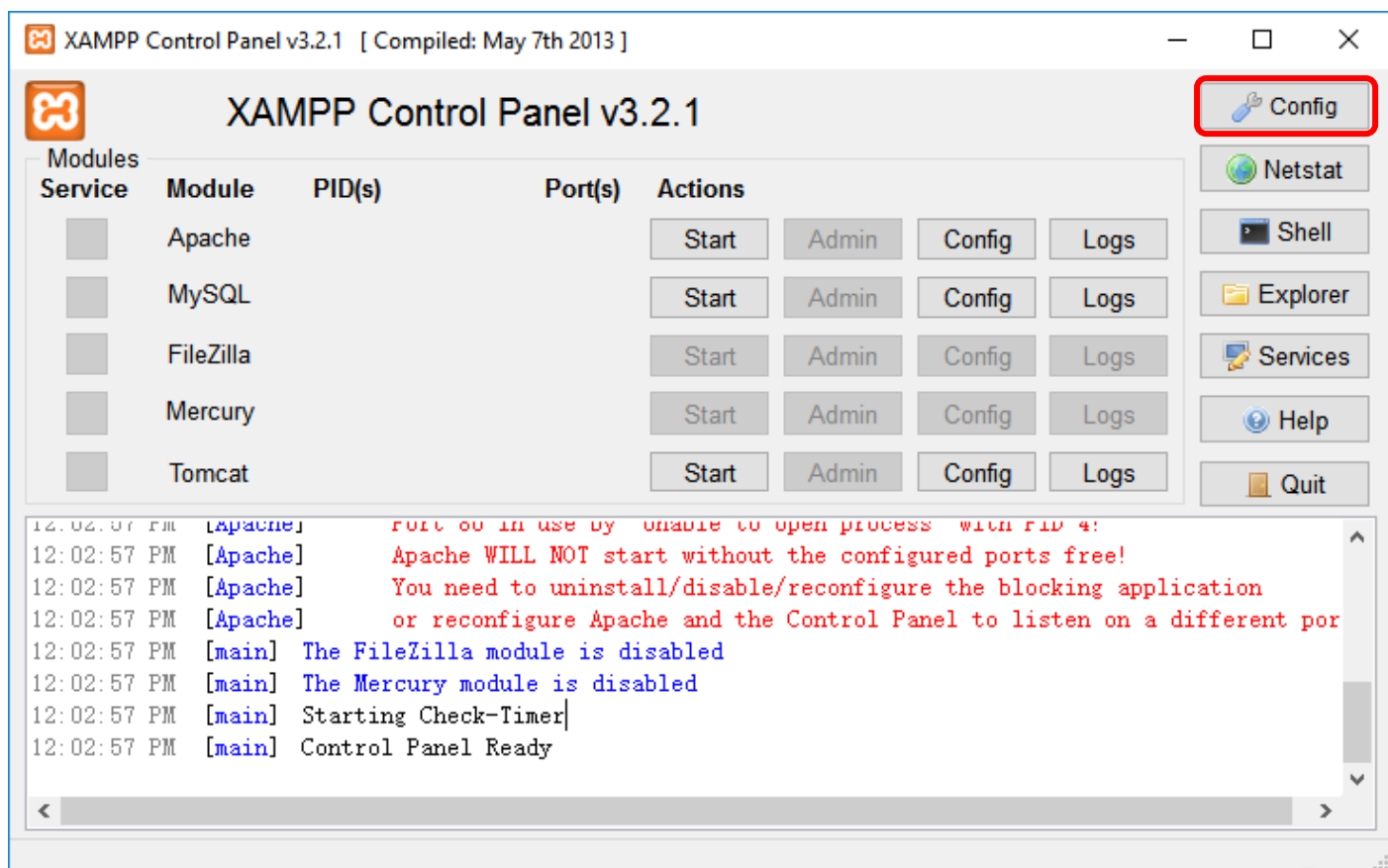
14. 最终应该能看到最后运行结果。



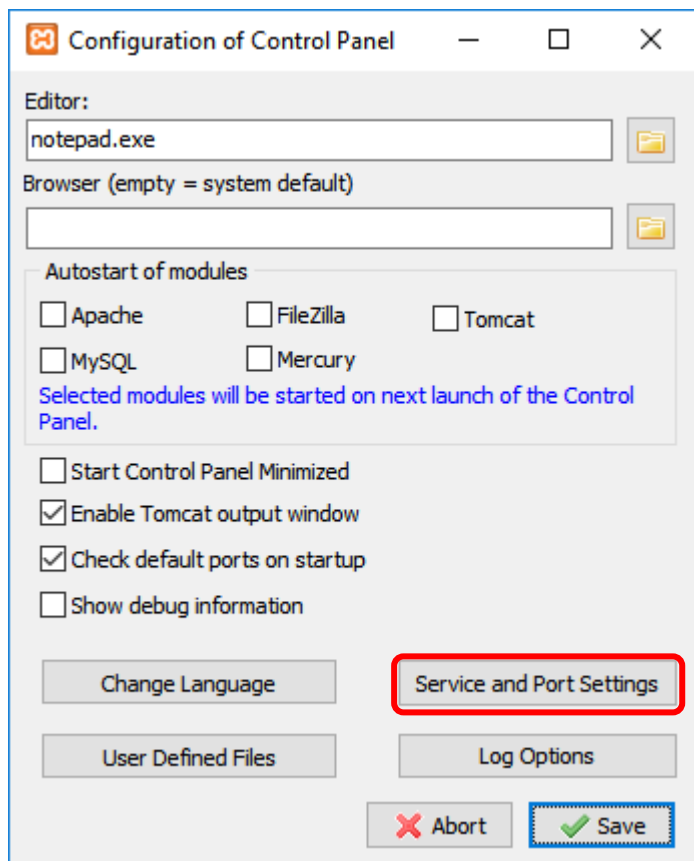
【任务三】：安装配置 MySQL 服务器。

步骤：

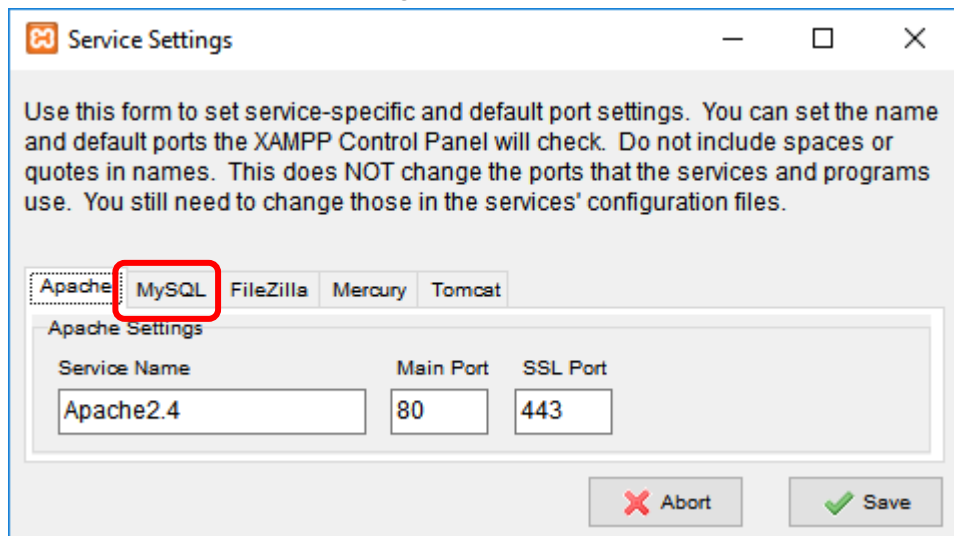
1. 将提供的 xampp 工具解压到 D 盘根目录下。（注意 xampp 一定要解压到根目录下才有效。所谓根目录是指 D:\xampp\目录下不能再嵌套 xampp 目录，而是直接在 D:\xampp\目录下能够找到 mysql 目录。切记！！）
2. xampp 是一套 web 开发套件，里面包含了 mysql、apache、tomcat 等常用组件。这里我们主要利用其中的 mysql 组件来完成数据库的开发。
3. 双击 xampp 目录下的 xampp-control.exe 文件运行，并显示如下界面：



4. 为了避免和本机上已经安装的 MySQL 服务器冲突，我们需要更改 xampp 的服务名称：单机界面上的 Config 按钮，显示如下界面：

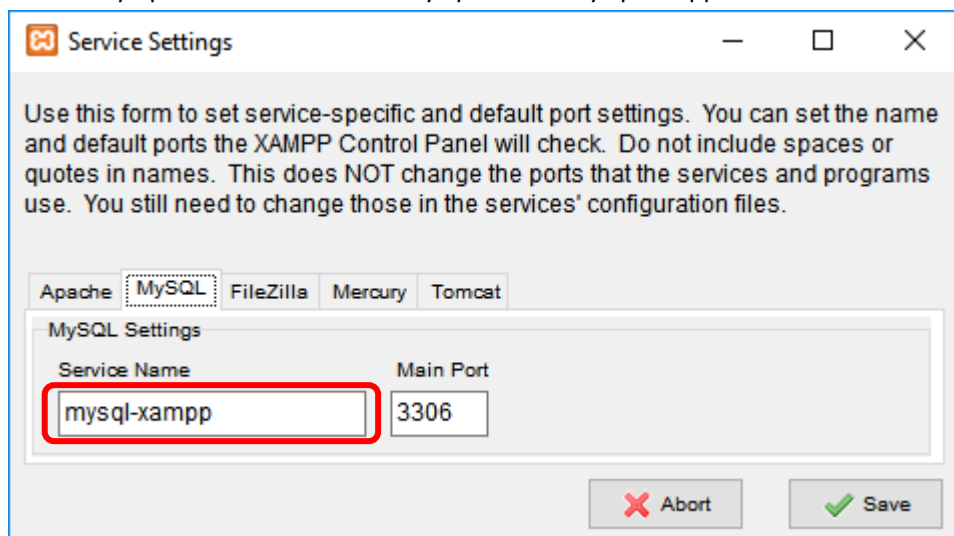


5. 点击“Service and Port Settings”按钮，将出现如下界面：



6. 点击 MySQL 标签页，配置 MySQL。

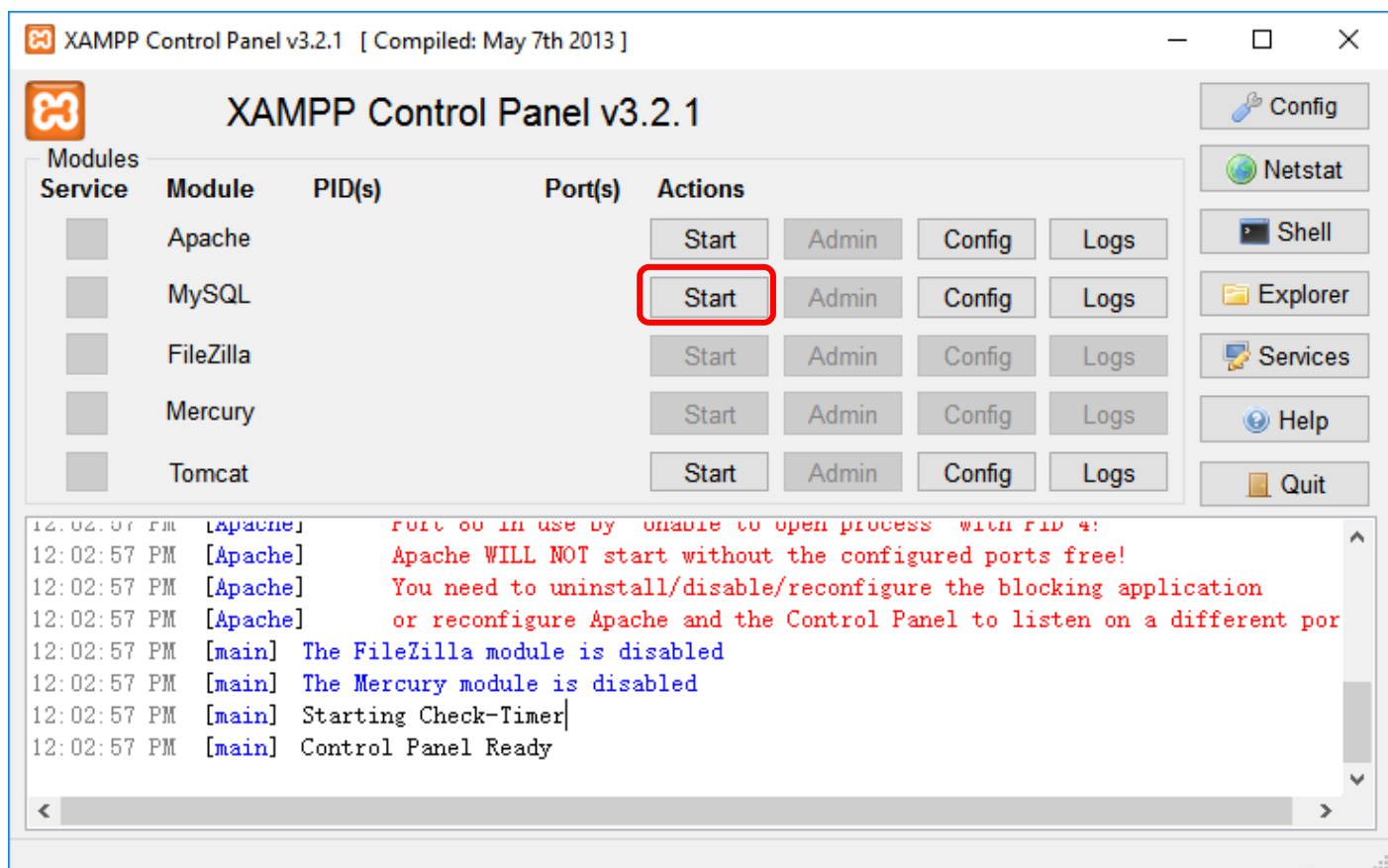
7. 将 MySQL 的 Service Name 从 mysql 更改为 mysql-xampp:



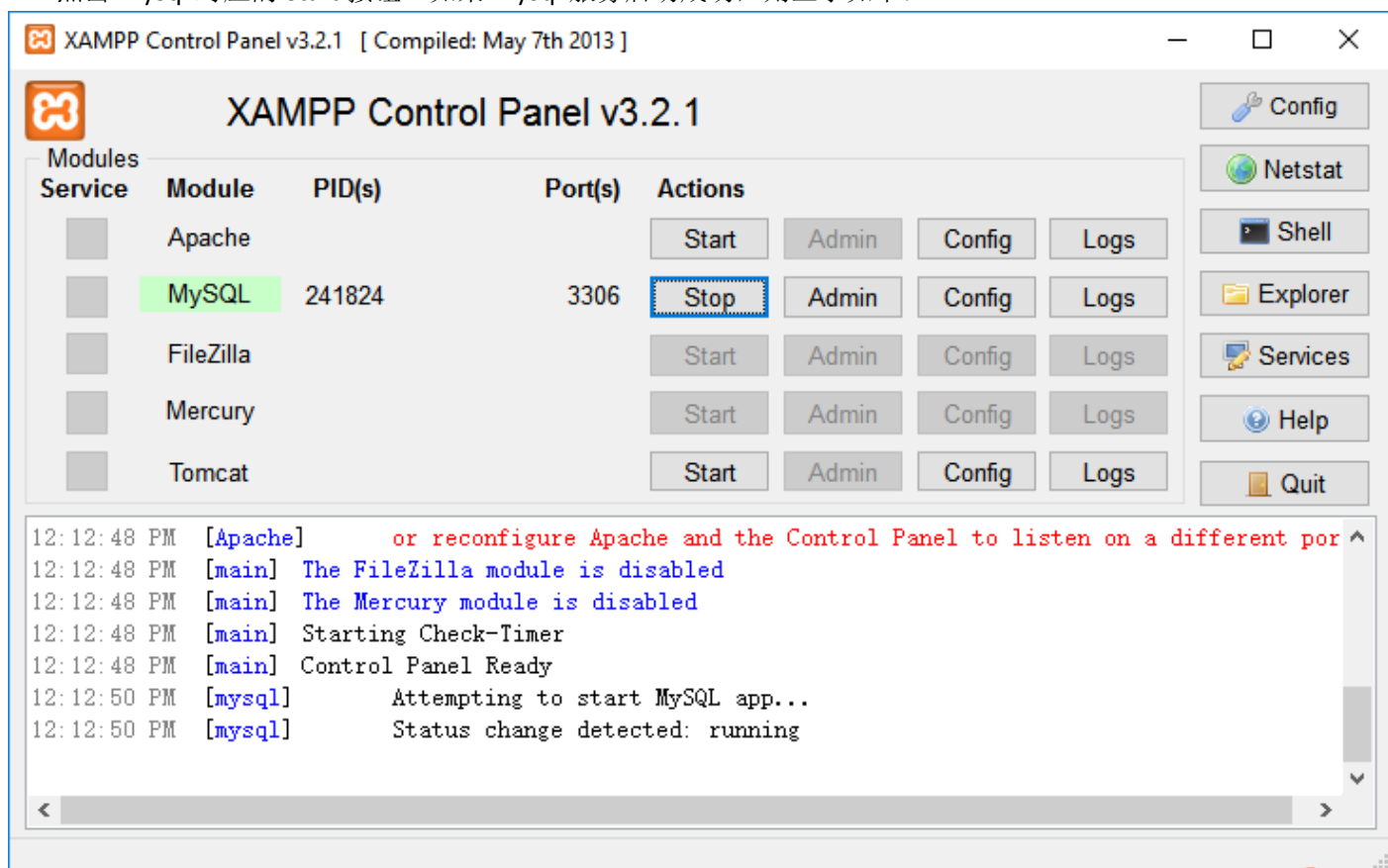
8. 点击保存。

9. 关闭 xampp。（注意如果关闭 xampp 后，xampp 任然在后台运行，必须从右下角系统托盘处完全关闭。）

10. 重新打开 xampp-control.exe 文件：



11. 点击 MySQL 对应的 Start 按钮。如果 MySQL 服务启动成功，则显示如下：

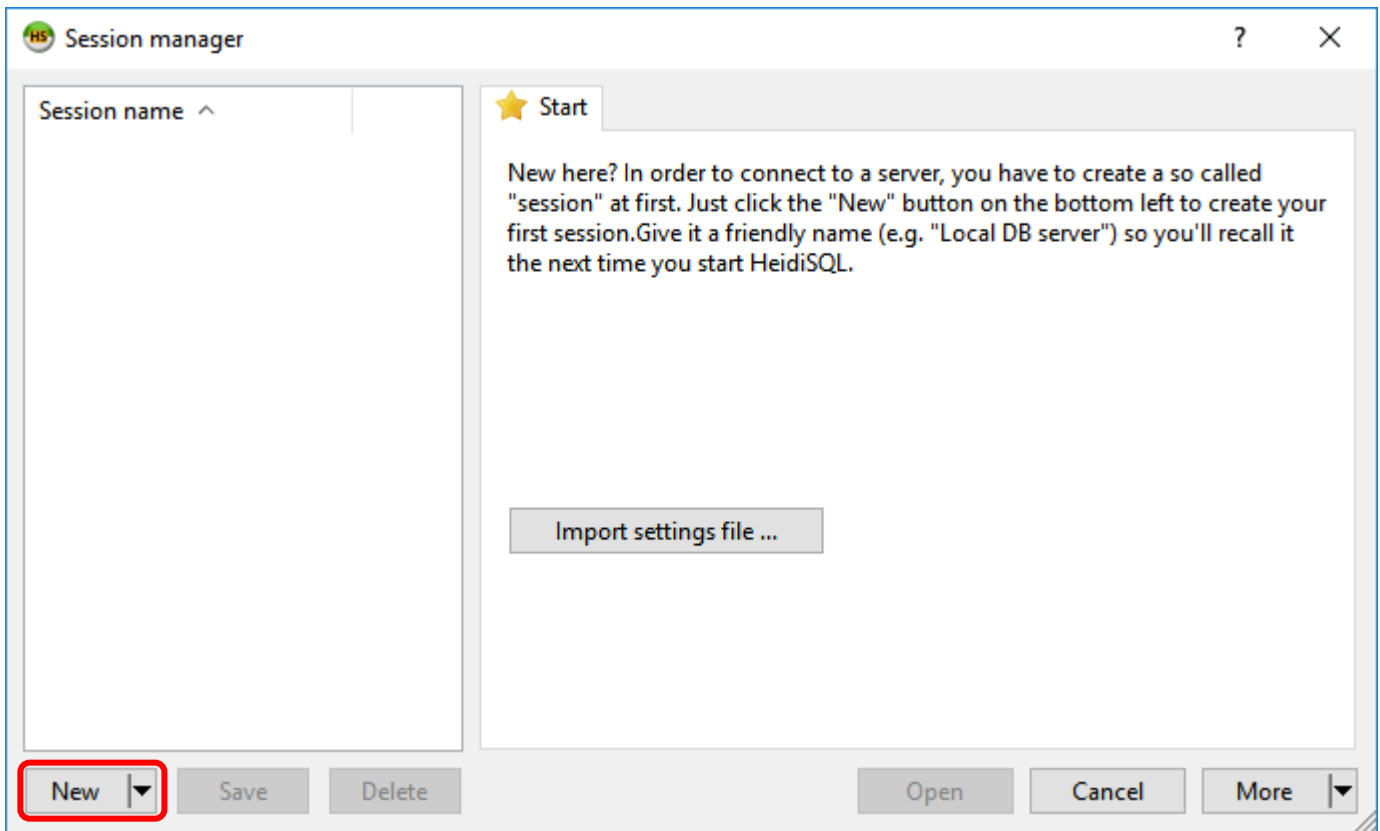


12. 其中，MySQL 字符的背景色变为绿色表示启动成功，PID 表示 MySQL 在操作系统中的进程号，Port 表示 MySQL

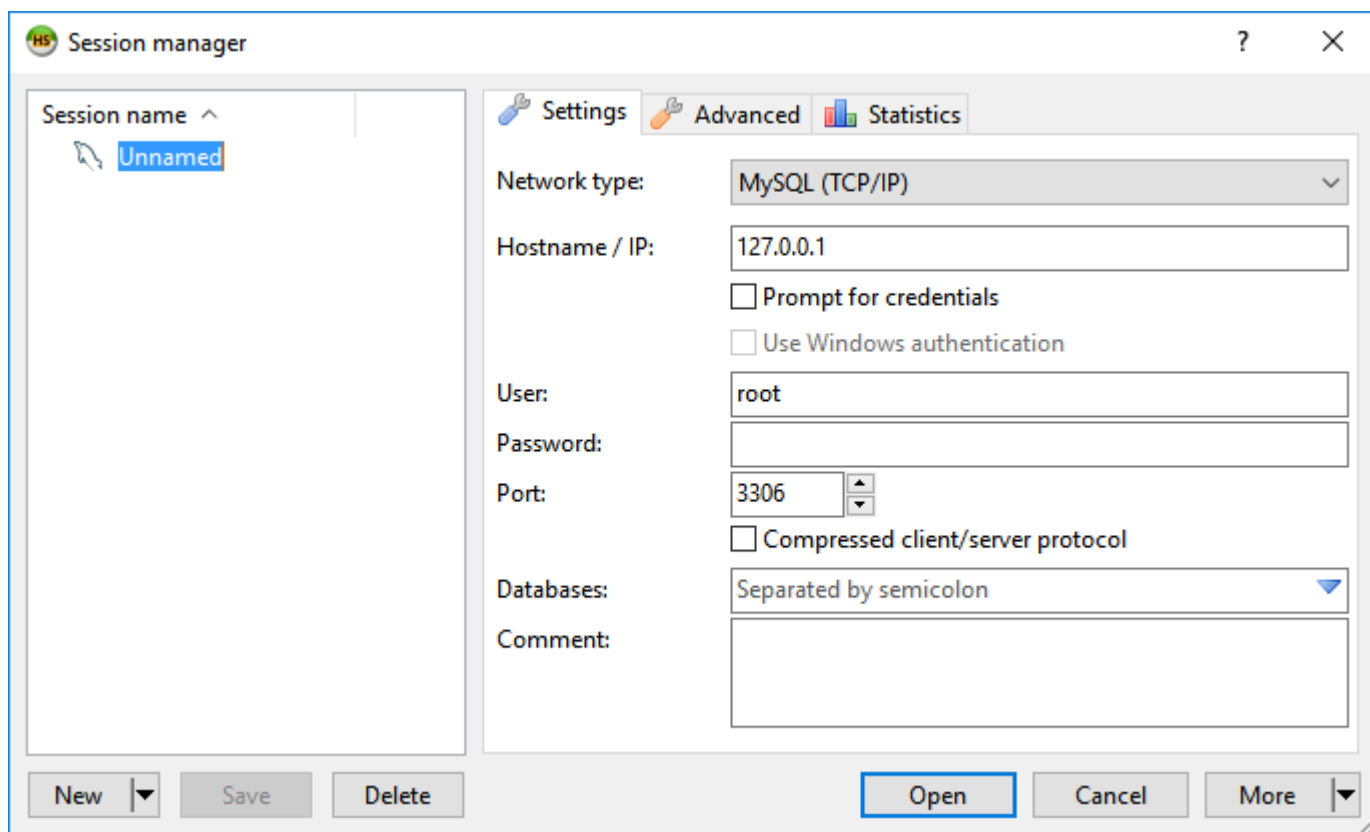
进程所监听的端口号。

【任务四】使用 MySQL 数据库，并建立数据库和表。

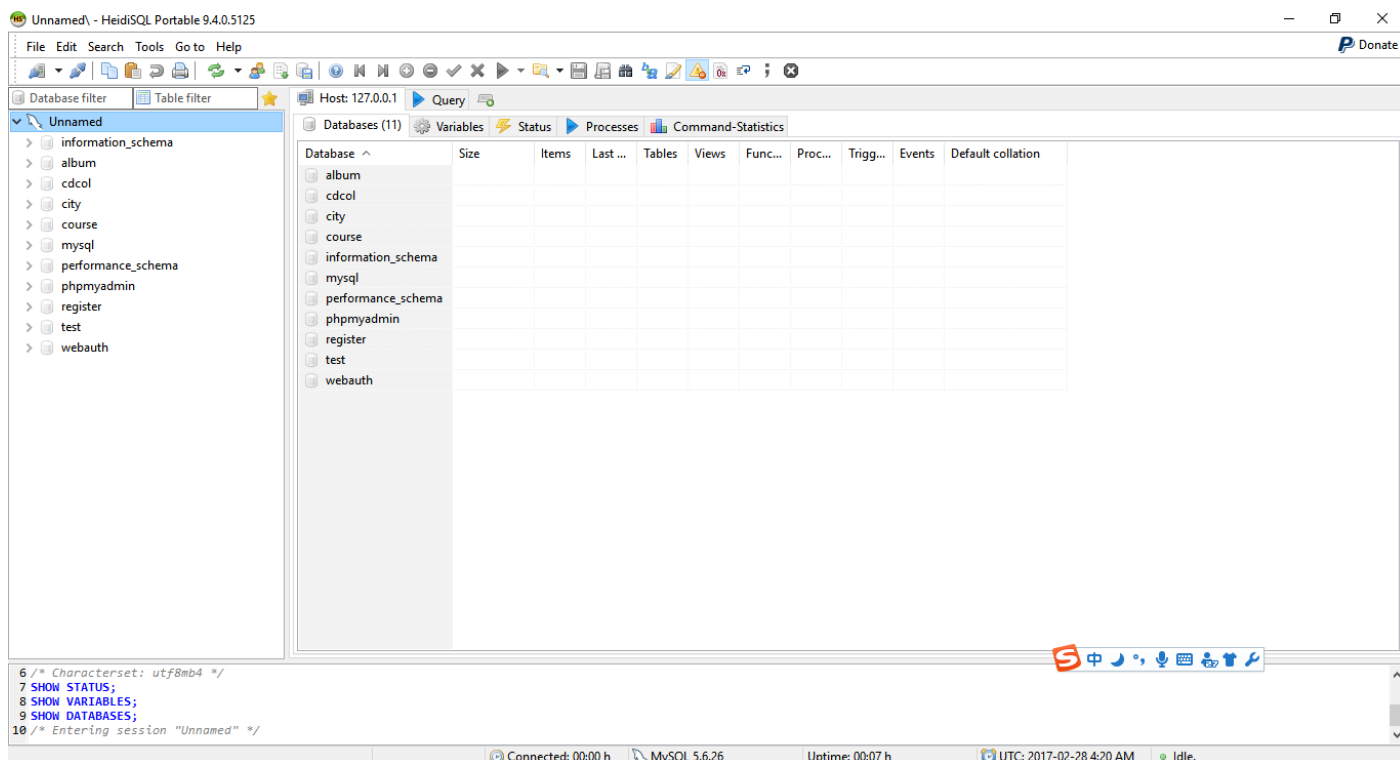
1. 解压并运行提供的 HeidiSQL 工具（绿色软件），将显示如下界面：



2. 点击 New 按钮，新建 MySQL 连接。



3. 在 hostname/IP 处填写：127.0.0.1 或者 localhost。在 User 处填写：root（这是 MySQL 的登录用户名）。在 Password 处填写：root（这是 MySQL 的登录密码。**注意：如果是自己下载的 xampp 工具，则密码为空。使用我提供的 xampp 工具，则 MySQL 登录密码被我改成 root。**）在 Port 处填写：3306（这是 MySQL 默认的监听端口）。
4. 随后点击“Open”按钮，将显示如下界面：

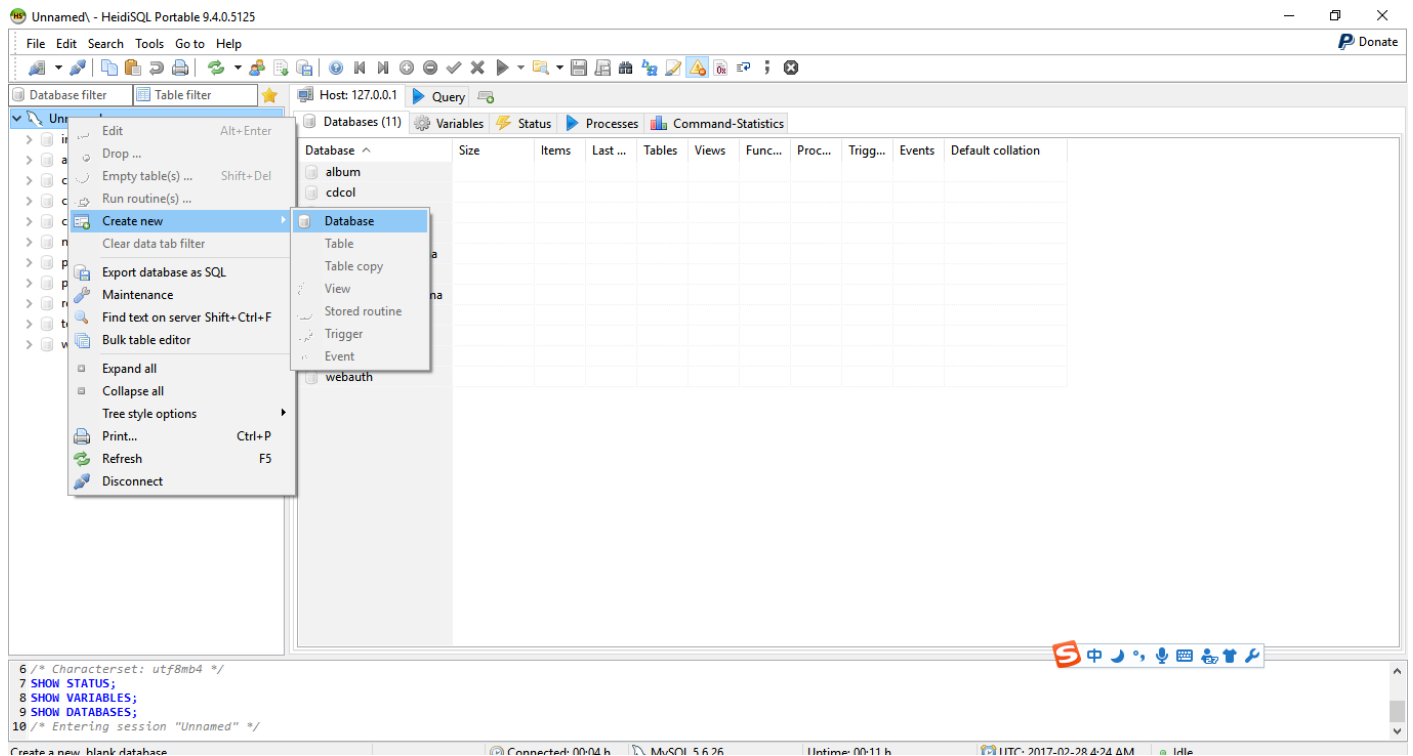


5. 左边列出了目前 MySQL 中的所有数据库，右边是当前数据库的内容。（请勿更改和删除默认数据库的内容）

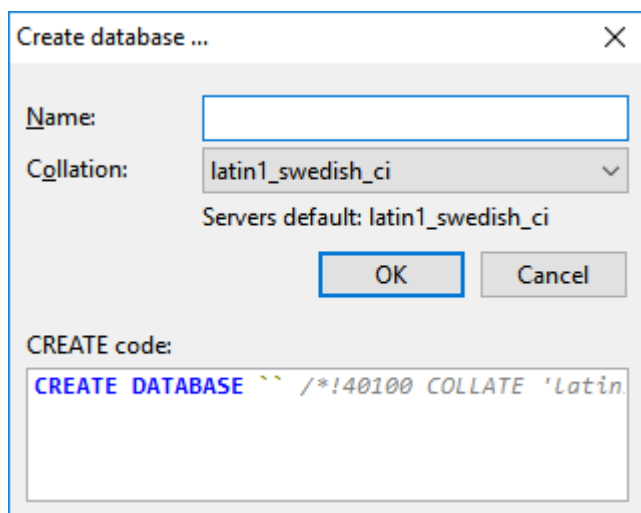
【任务五】：在 MySQL 中创建数据库、导入和导出数据。

步骤：

1. 在 HeidiSQL 界面中左边数据库树的最顶层点击鼠标右键，上图中显示 “Unnamed”。

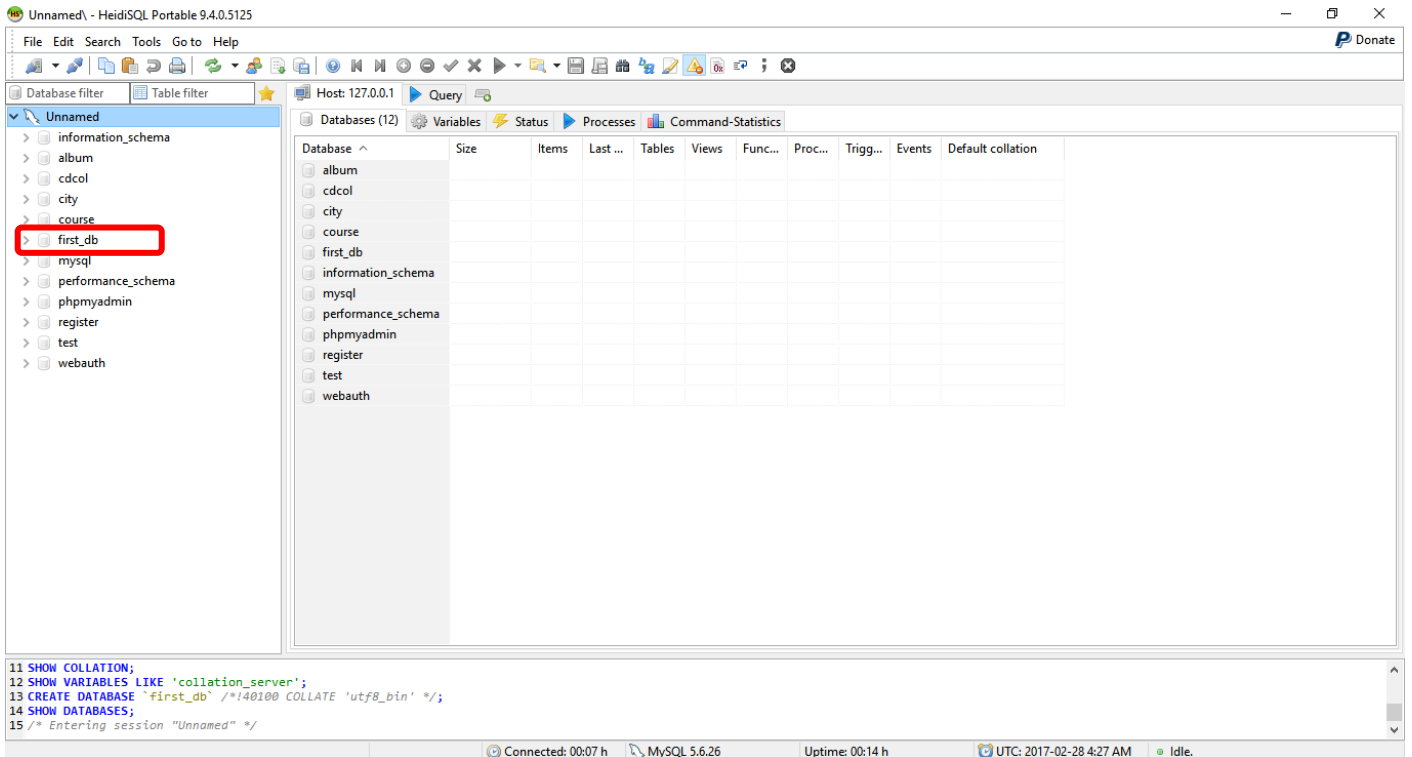


2. 选中菜单项：“Create new” → “Database” 来创建新数据库：

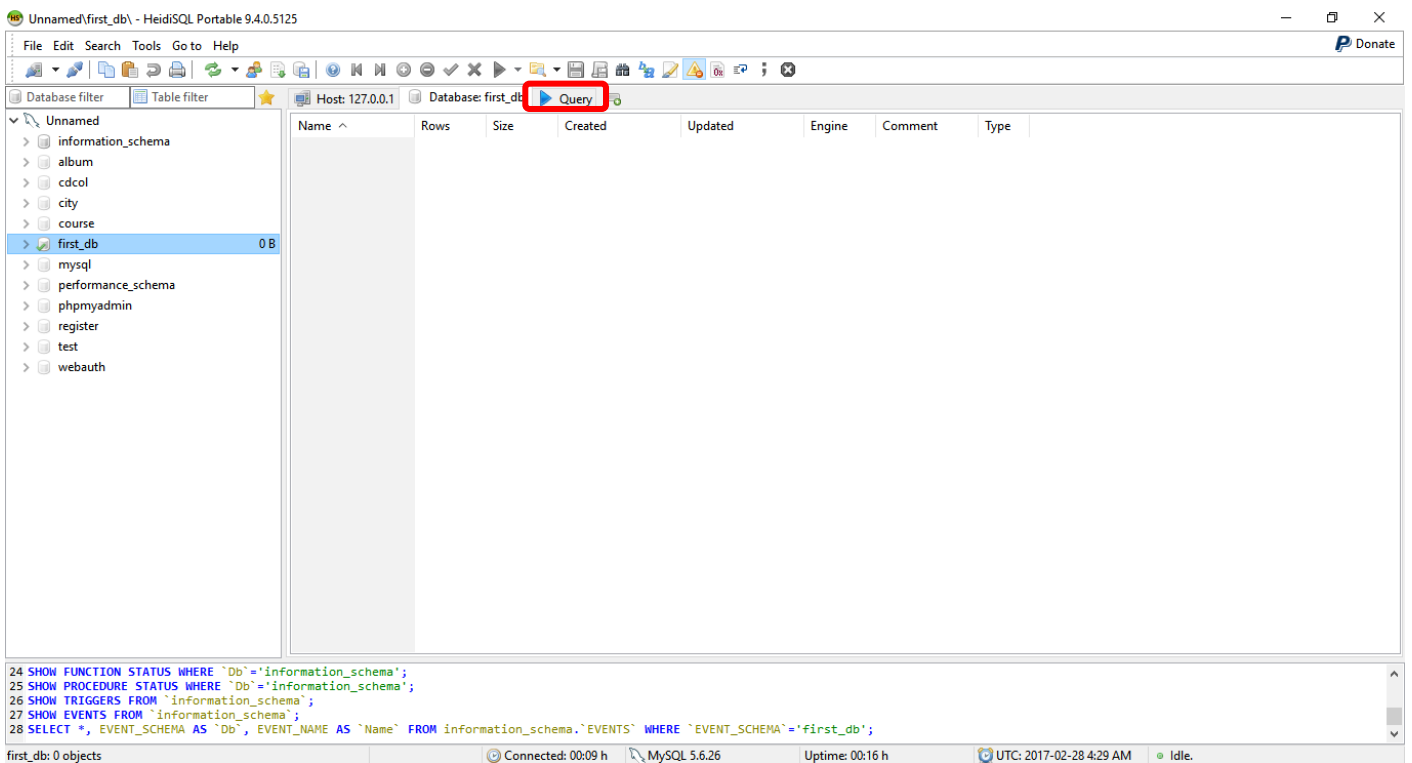


3. 在 Name 处填入数据库的名字：first_db（注意 first 和 db 中间为下划线），并在 Collation 处选择字符编码格式为：utf8_bin。最后点击确定创建数据库。

4. 此时，界面左边数据库栏中将多出一项 first_db 数据库。



5. 选中 first_db 数据库，并选择右边界面中的 Query 标签：



6. 在文本框当中输入语句创建一个数据库表：

```

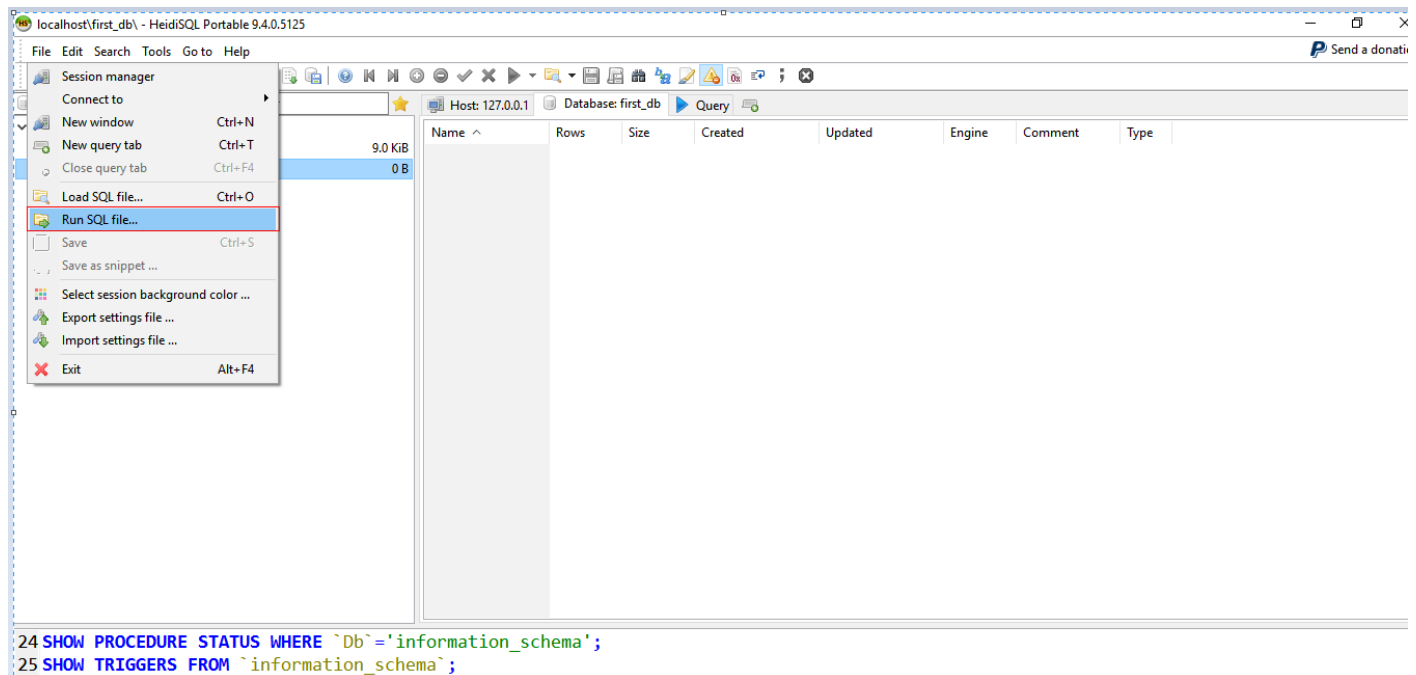
CREATE TABLE `student` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(50) COLLATE utf8_bin NOT NULL DEFAULT '0' COMMENT '姓名',
  `student_no` varchar(50) COLLATE utf8_bin NOT NULL DEFAULT '0' COMMENT '学号',

```

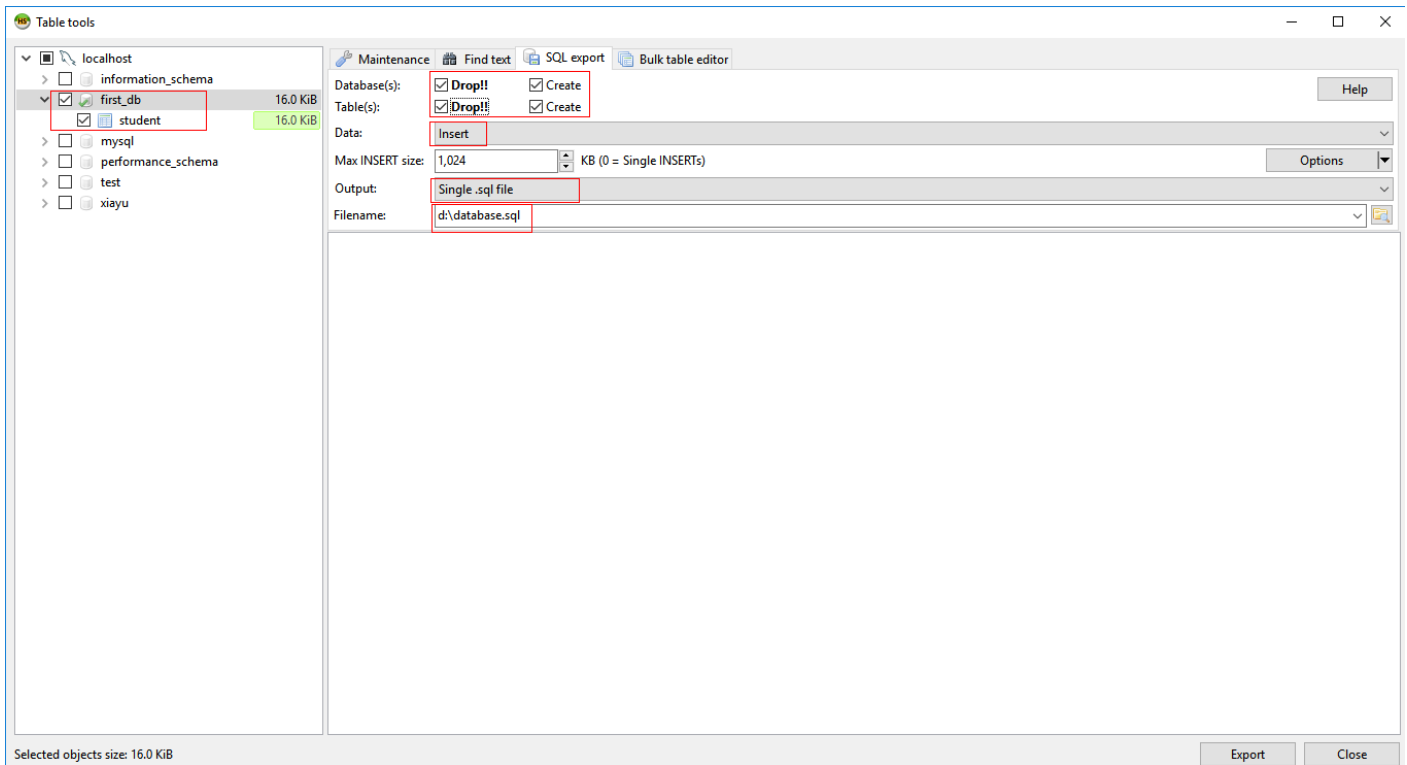
```
`phone` varchar(50) COLLATE utf8_bin NOT NULL DEFAULT '0' COMMENT '电话',
PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=37 DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

注意：其中数据库表的名字 **student**，以及列的名字如 **id**、**name** 等前后的引号并不是真正的引号，而是键盘上数字键 **1** 左边的英文重音符号 `。

7. 点击工具栏中的刷新按钮 “🔄”（或者按 F5 键）刷新数据库列表。
8. 展开 **first_db** 数据库，此时将看到 **first_db** 当中的表。
9. 选择 **File->Run SQL file...**，然后选择提供的 **data.sql** 文件运行，将导入数据。



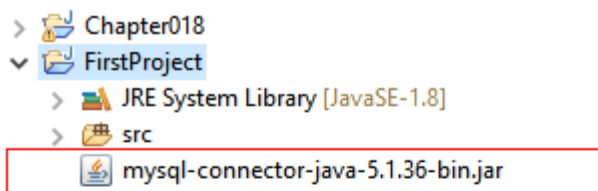
10. 导入数据成功。
11. 在数据库 **first_db** 上点鼠标右键，在菜单中单击 **Export database as SQL**。
12. 按下图的配置选择导出设置，并点击 **Export** 按钮，然后可以获取结果。导出结果请打开，并截图。



【任务六】：在 Eclipse 中引用第三方类库。

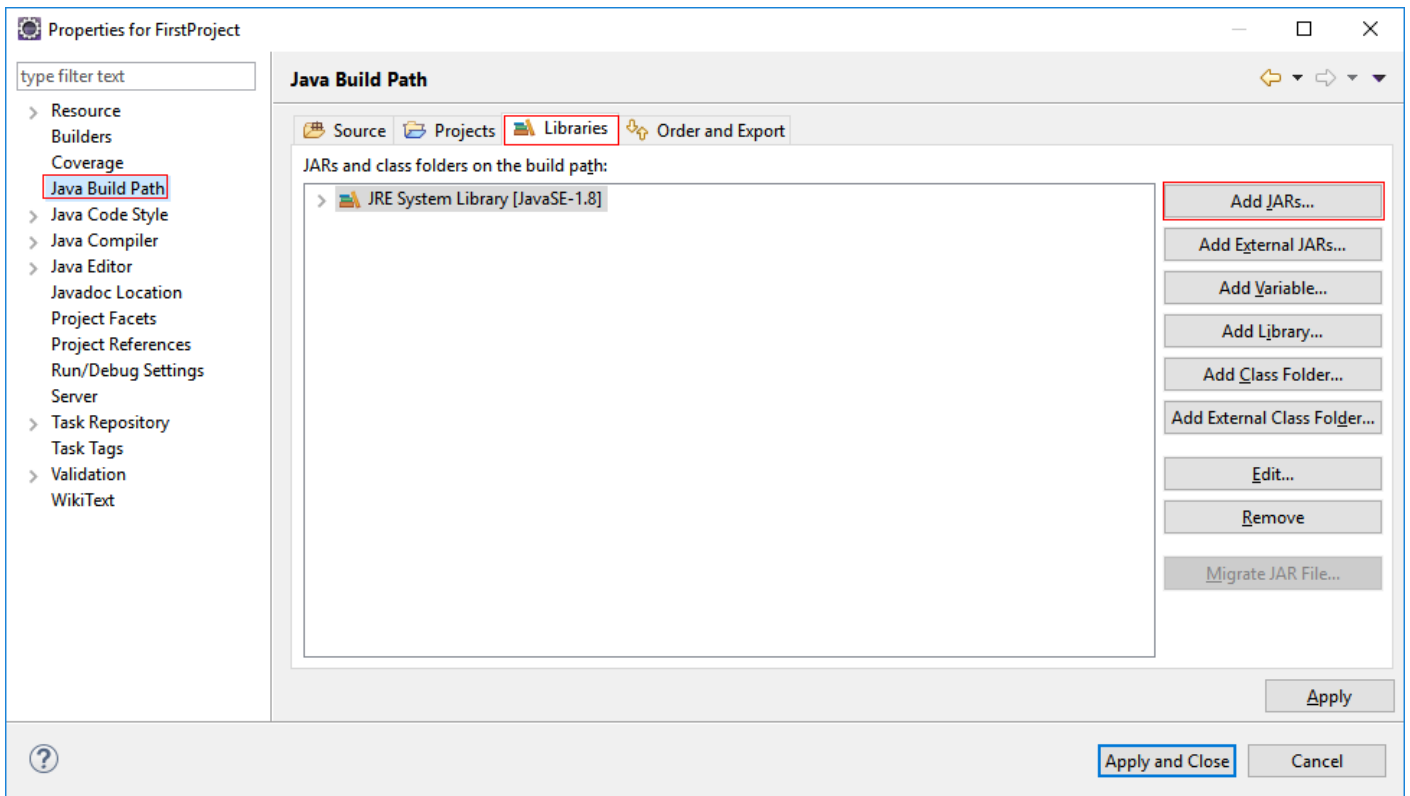
步骤：

1. 将提供的 MySQL 的 JDBC 驱动文件 mysql-connector-java-xxxx-bin.jar 文件复制到工程目录当中。
2. 将提供的 usedb.jsp 文件复制到 first_webapp 目录下。

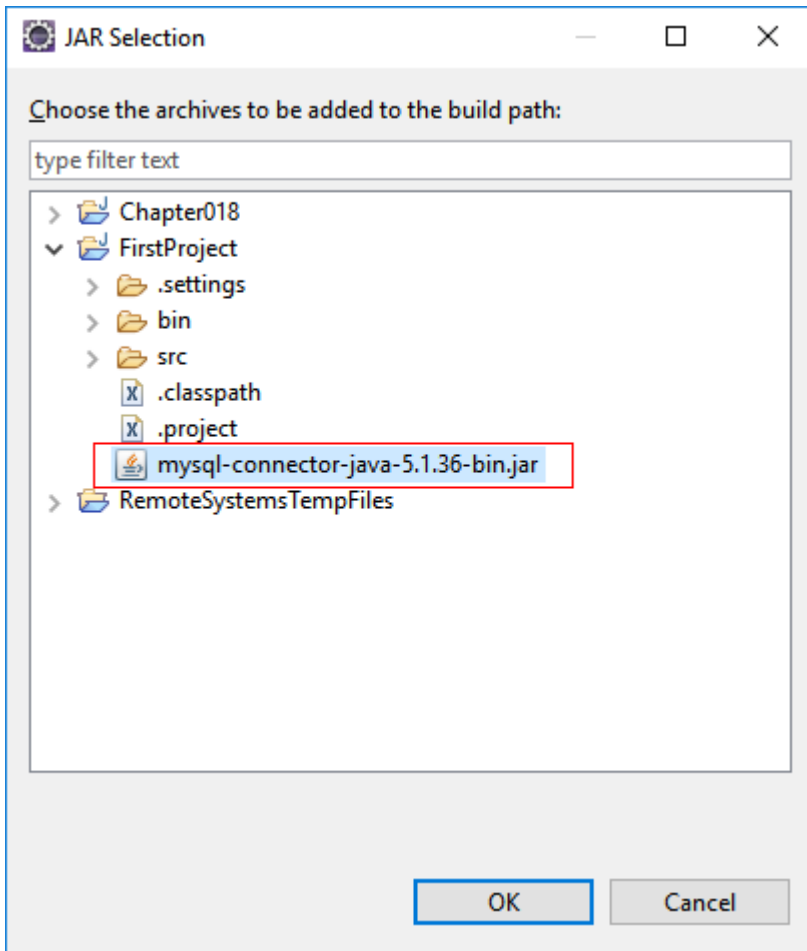


3. 在工程上点鼠标右键，选择 Properties。

4. 在弹出的对话框当中选择 Java Build Path ->Libraries->Add JARs...



5. 展开 FirstProject 项目，选中刚刚复制的 jar 文件。



6. 此时会看到 jar 文件已经被导入到工程依赖当中。
7. 最后点击 Apply and Close 完成设置。

【任务七】：在 Eclipse 中测试数据库连接。

1. 将提供的 conn.ini 复制到工程目录下。
2. 将提供的 DataManager.java 文件复制到 src 目录下。
3. 最后运行 DataManager 类，如果看到数据则说明运行成功。