

# 实验 1 Mybatis 环境搭建及入门案例

## 一、环境准备（工具及版本）

- 1、Java 版本：JDK 1.8
- 2、构建工具：Maven 3.6.3
- 3、中间件服务器：Tomcat 9.0.44
- 4、浏览器：Google Chrome 100+ 以上
- 5、开发工具：IntelliJ IDEA Ultimate 2022.3.3
- 6、数据库：MySQL 5.7+ 以上
- 7、数据库工具：Navicat 10+ 以上

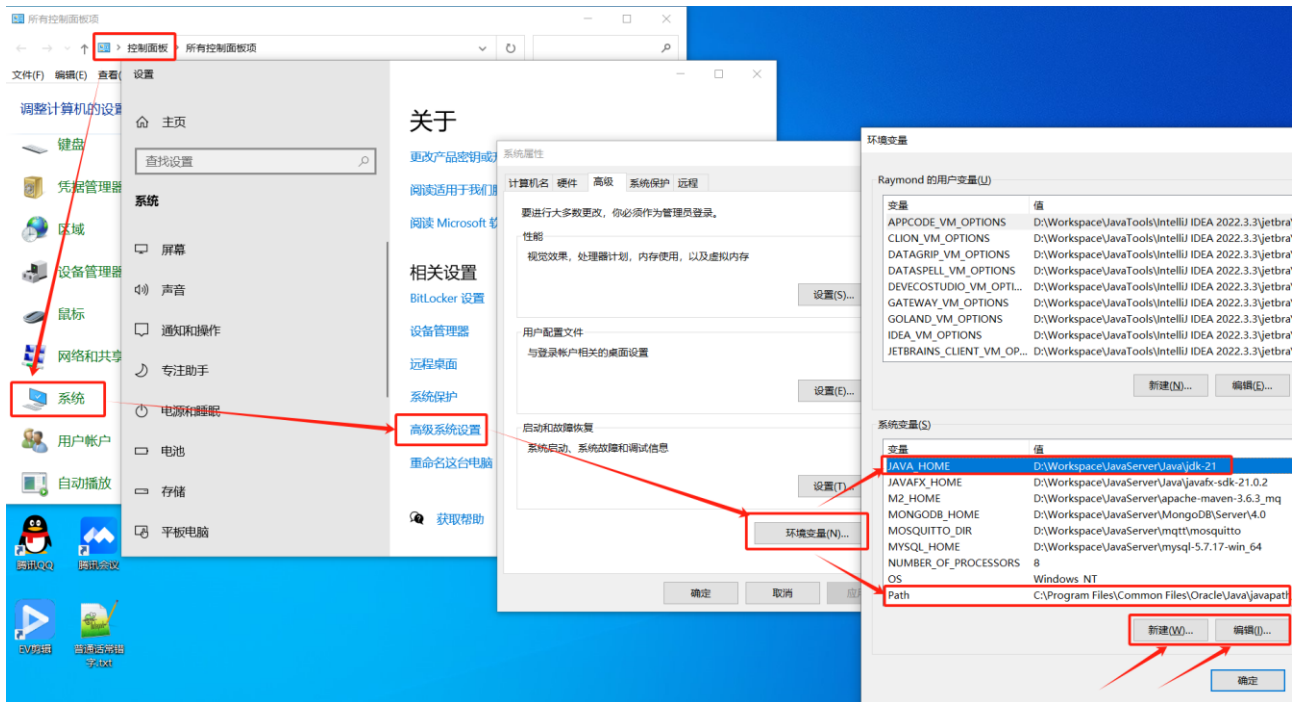
## 二、Mybatis 环境搭建

### （一）、环境安装

- 1、首先安装好 JDK（可下载免安装版）
- 2、安装好开发工具 IntelliJ IDEA Ultimate 2022.3.3（可下载免安装版）
- 3、安装好 MySQL 数据库
- 4、安装好数据库工具 Navicat
- 4、Maven、Tomcat 和 Chrome 浏览器等直接解压即可使用

### （二）、环境配置

总述：环境变量的进入路径，见下图示：



## 1、配置 JDK 环境变量，如：

JAVA\_HOME=C:\Program Files\Java\jdk1.8.0\_191

Path = %JAVA\_HOME%\bin;

注：默认情况下，安装好 Jdk 后，会自动配置环境变量。如果检查 JDK 环境已经正常，则可以忽略上面配置。检查方法如下：

```
C:>java -version
```

回车后，能出现 java 版本号即说明 Jdk 已经配置完成。

## 2、配置 Maven 环境变量，如：

M2\_HOME= C:\Program Files\Maven\apache-maven-3.6.3

Path = %M2\_HOME%\bin;

另外，Maven 默认情况下，是使用国外的仓库地址，这会下载依赖包非常慢。因此，我们必须将 Maven 的仓库地址换成我们国内的。我们国内的仓库地址一般是阿里的地址。修改方法：

进入到%M2\_HOME%\conf 目录，打开 settings.xml 配置文件，在<mirrors> </mirrors>节点中添加如下节点：

```
<mirror>

    <id>nexus-aliyun</id>

    <name>Nexus aliyun</name>

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

    <mirrorOf>central</mirrorOf>

</mirror>
```

再有就是，为了更好的管理我们的依赖包，我们最好指定 Maven 下载下来的依赖包存放的目录，这样方便我们管理。方法就是在上面的 settings.xml 配置文件中，在如下节点配置你的存储路径即可。注意这个路径必须真实存在的。比如：

```
<localRepository>C:/Program Files/Maven/apache-maven-3.6.3/Local_REPO</localRepository>
```

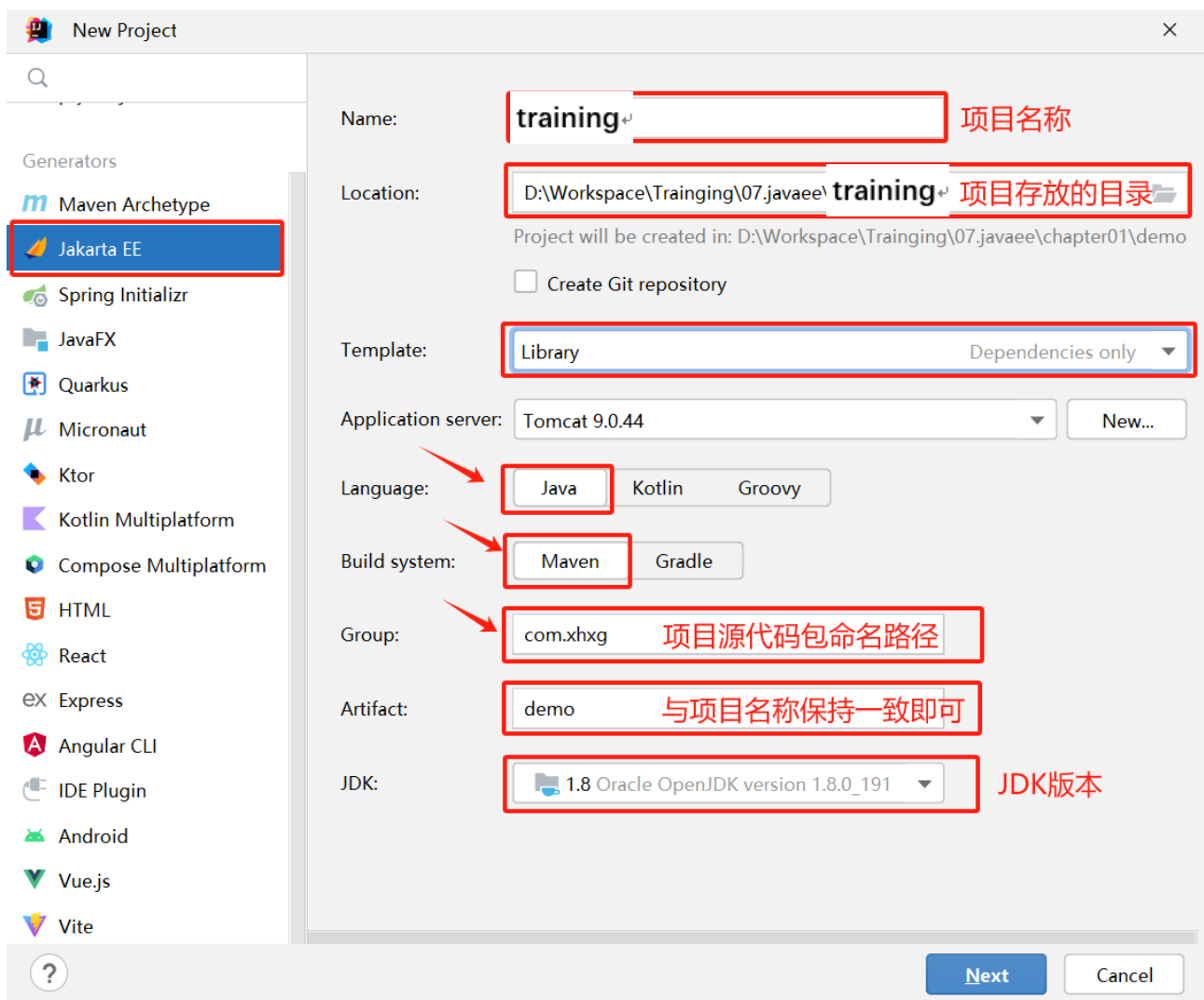
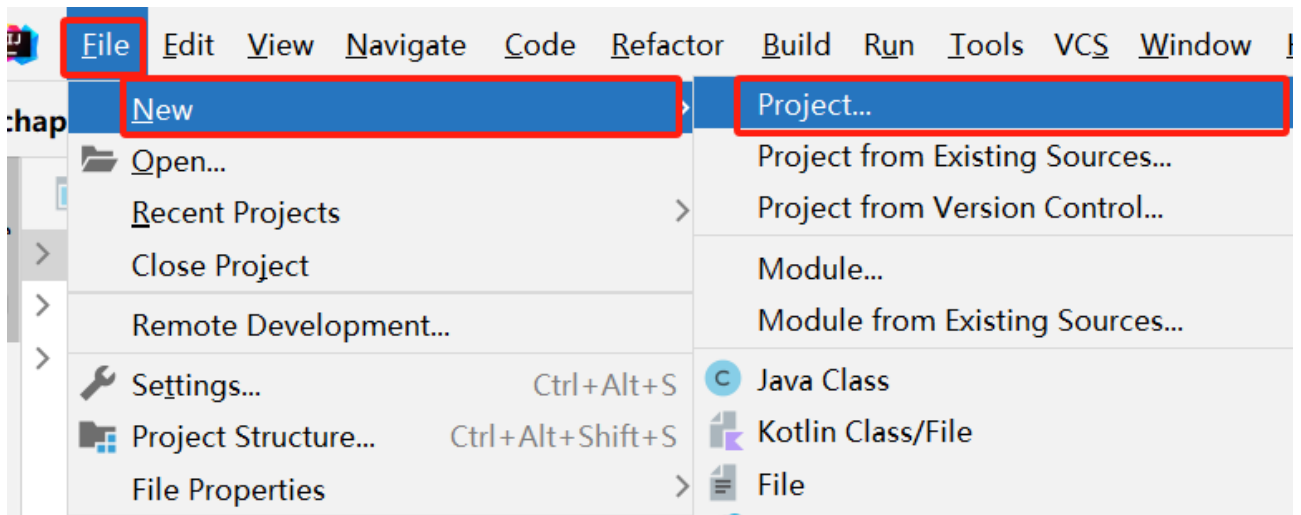
### 3、配置 Tomcat 环境变量，如：

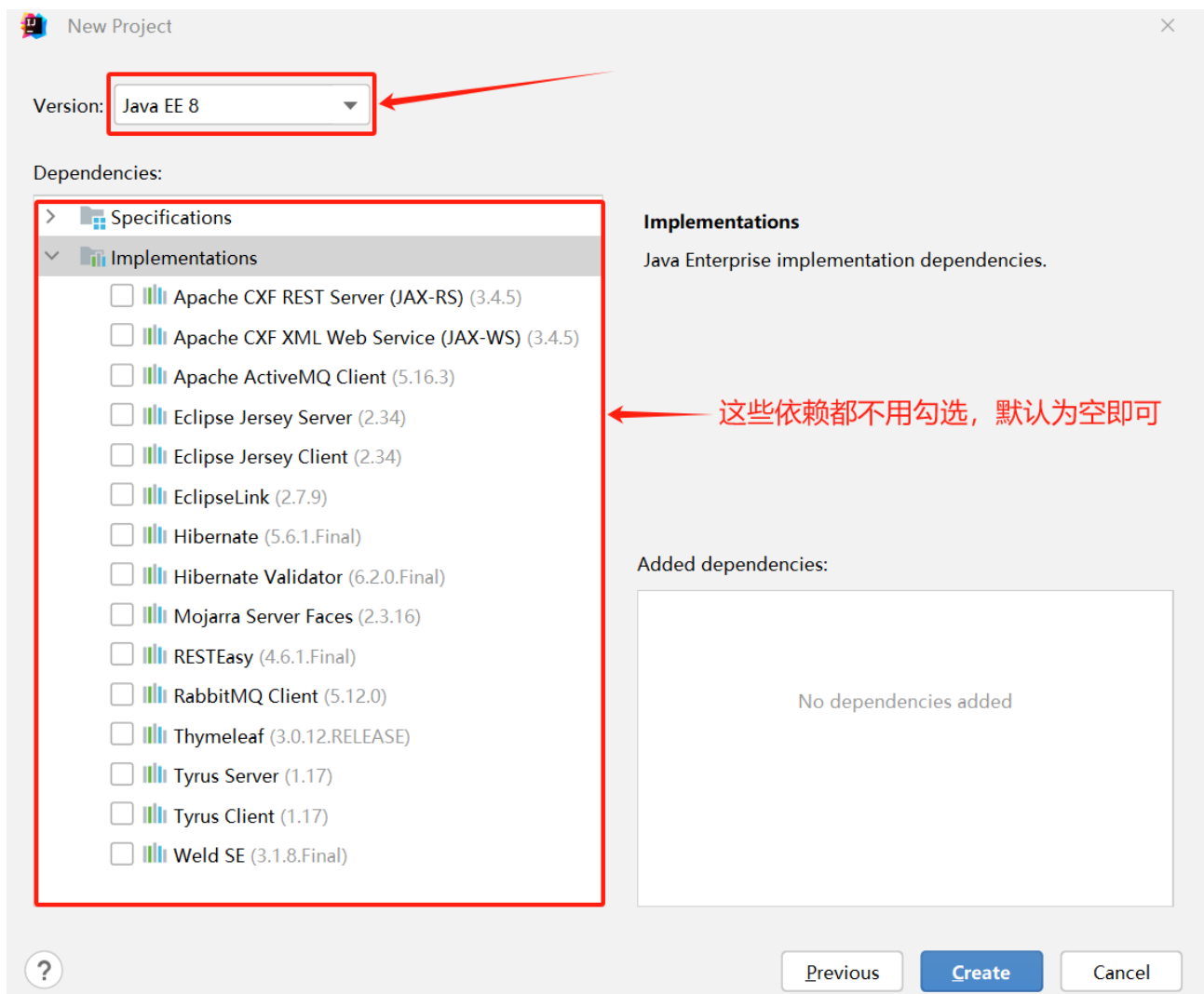
TOMCAT\_HOME = C:\Program Files\Tomcat\apache-tomcat-9.0.44

Path = %TOMCAT\_HOME%\bin;

## 三、Maven 项目工程创建

### 1、创建 Maven 项目工程，按如下示例，进行创建





最后一步，点击 **【Create】** 即可完成工程创建。

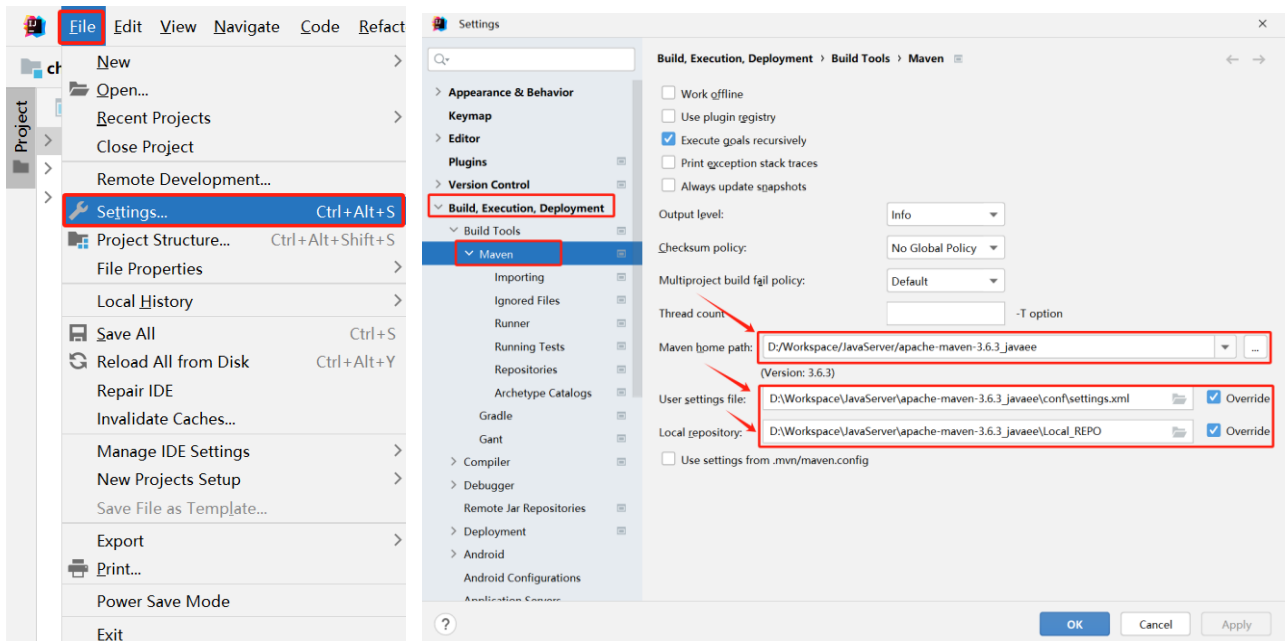
**注意：**如果不手工创建工程，也可直接下载 training 工程。路径：从 **【学习通】** → **【章节】** → **【第一章…】** → **【上机实验 1…】**，找到 training.zip 压缩包，下载到本地，并解压成 training 目录。

## 2、配置 Maven 构建工具

项目工程创建后，我们需要在本项目配置使用我们自己的 Maven 构建工具，以加快我们项目工程的构建。配置方法如下：

打开我们创建的 Maven 项目工程后，选择 **【File】** → **【Setting…】** 然后找到菜单 **【Build, Execution, Deployment】** → **【Build Tools】** → **【Maven】**，按如下图所示，将你的本地 Maven

路径和配置文件设置好即可。

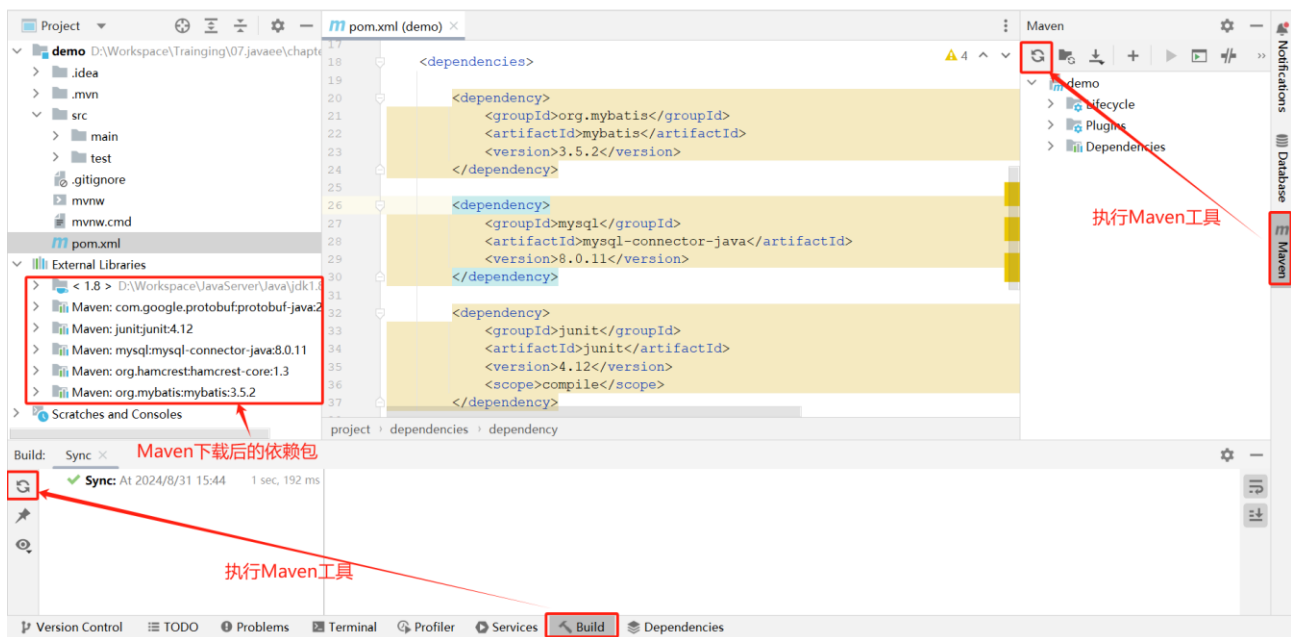


### 3、配置项目工程的 pom.xml 文件，在这里加入我们需要的依赖包。

根据 Mbatis 的使用需要，我们需要加入 Mbatis 依赖库，MySQL 数据库驱动依赖库，另外，为了做单元测试，我们还需要加入 Junit 依赖库。完整的配置如下：

```
pom.xml x
1  <?xml version="1.0" encoding="UTF-8"?>
2  <project xmlns="http://maven.apache.org/POM/4.0.0"
3      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4      xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
5      <modelVersion>4.0.0</modelVersion>
6
7      <groupId>com.xhcg</groupId>
8      <artifactId>demo</artifactId>
9      <version>1.0-SNAPSHOT</version>
10     <name>demo</name>
11
12     <properties>
13         <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
14         <maven.compiler.target>1.8</maven.compiler.target>
15         <maven.compiler.source>1.8</maven.compiler.source>
16     </properties>
17
18     <dependencies>
19
20         <dependency>
21             <groupId>org.mybatis</groupId>
22             <artifactId>mybatis</artifactId>
23             <version>3.5.2</version>
24         </dependency>
25
26         <dependency>
27             <groupId>mysql</groupId>
28             <artifactId>mysql-connector-java</artifactId>
29             <version>8.0.11</version>
30         </dependency>
31
32         <dependency>
33             <groupId>junit</groupId>
34             <artifactId>junit</artifactId>
35             <version>4.12</version>
36             <scope>compile</scope>
37         </dependency>
38     </dependencies>
39
40     <build>
41         <resources>
42             <resource>
43                 <directory>src/main/java</directory>
44                 <includes>
45                     <include>/**/*.properties</include>
46                     <include>/**/*.xml</include>
47                 </includes>
48                 <filtering>true</filtering>
49             </resource>
50         </resources>
51     </build>
52 </project>
```

pom.xml 配置文件配置完成后，即可执行 Maven 工具，拉取在 pom.xml 中配置的依赖库，如下：



经过以上步骤，即可搭建完成 Mybatis 开发环境，下面进入 Mybatis 实例开发的过程。

#### 四、Mybatis 入门案例

**实验目标：**在 Mybatis 开发环境下，实现一个入门程序来演示 Mybatis 框架的使用，该程序要求实现根据 id 查询用户信息的操作。

##### 第 1 步：数据库开发

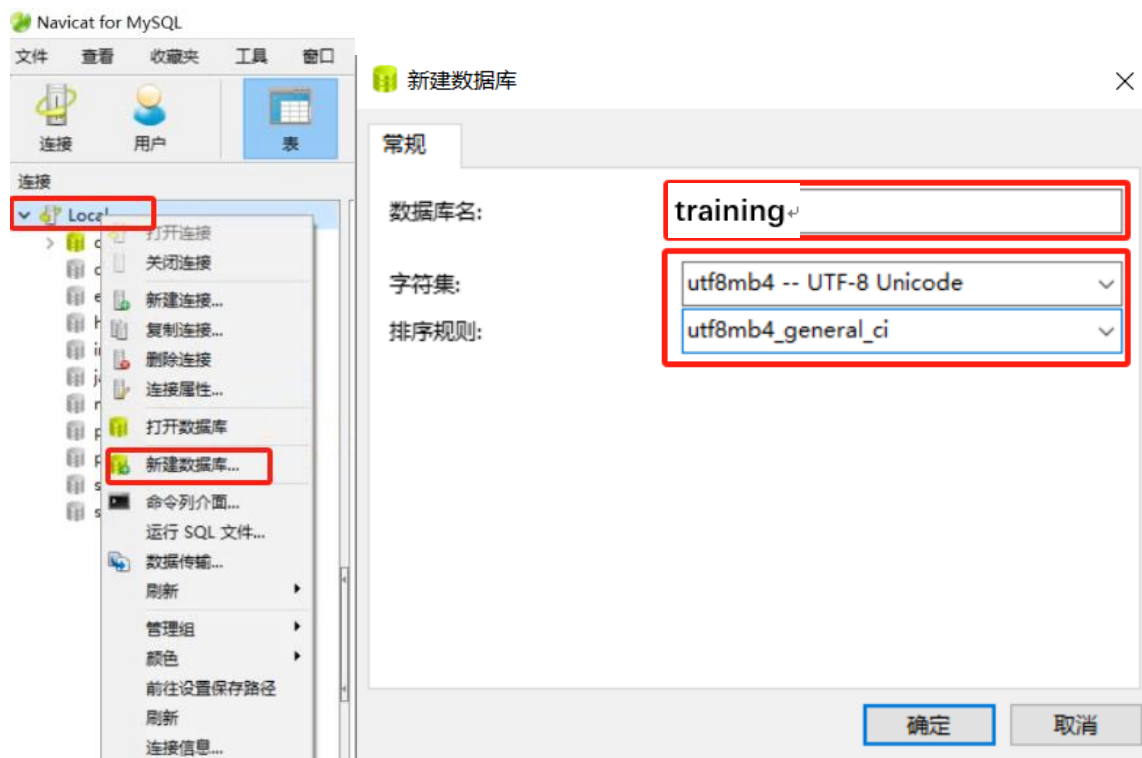
先使用 MySQL 数据库创建一个 `training` 数据库，然后创建一个学生表 `t_users`，方法如下：

通过使用 Navicat 工具，连接上你安装的 MySQL 数据库，输入用户名和密码进行登录。

假设建立的 MySQL 连接名为 `Local`，数据库登录用户名和密码为：`root/123456`。

登录后，手动创建数据库，需要创建的数据库名为：`training`，如图：选中 `Local` 连接名，右键选择【新建数据库】





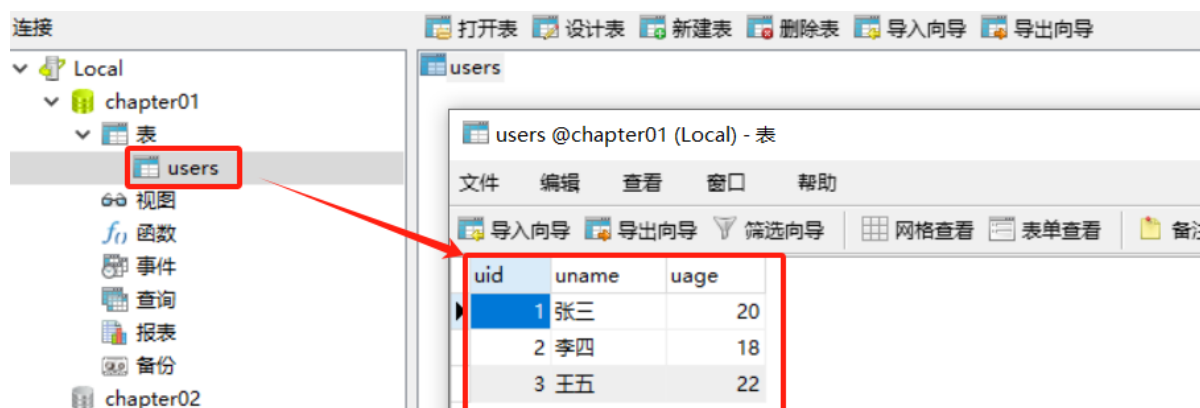
数据库创建好后，创建表 `t_users`，表结构如下。并按要求加入几条测试数据。

```

1 use training;
2
3 create table t_user(
4     uid int primary key auto_increment,
5     uname varchar(20) not null,
6     uage int not null
7 );
8 insert into t_user(uid,uname,uage) values (null,'张三',20),
9                                           (null,'李四',18),
10                                          (null,'王五',22);

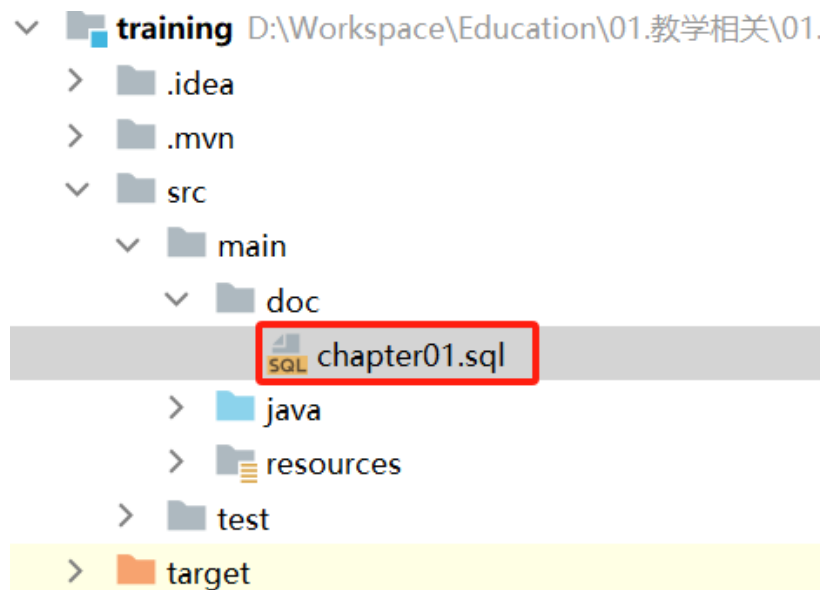
```

全部创建完成后，效果如下：

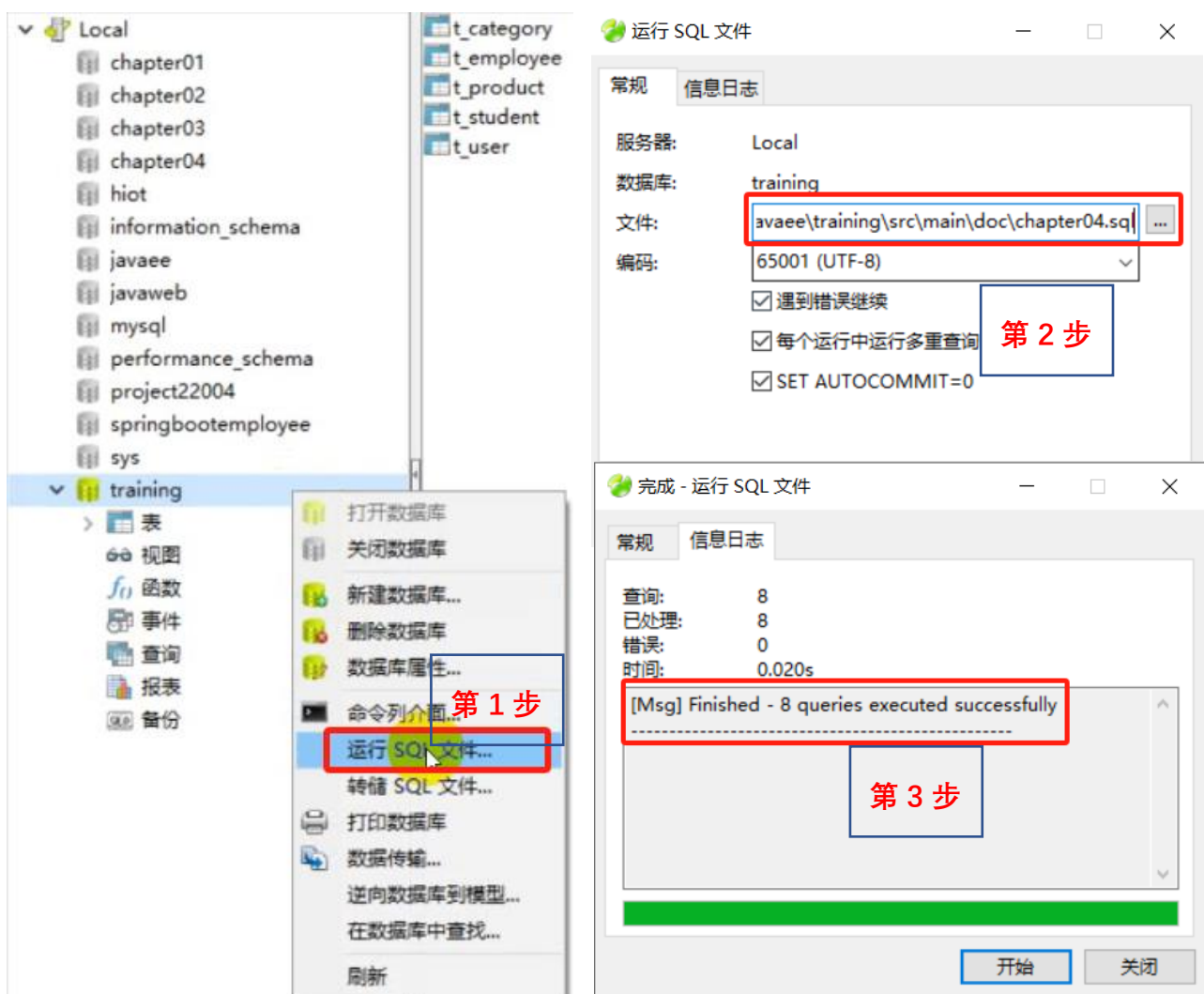


至此，数据库部分开发完毕。

注：以上建表数据库脚本已放置工程的 doc 目录，文件路径如下图所示：



可以通过 Navicat 工具直接读取此脚本文件，以完成数据表的创建。

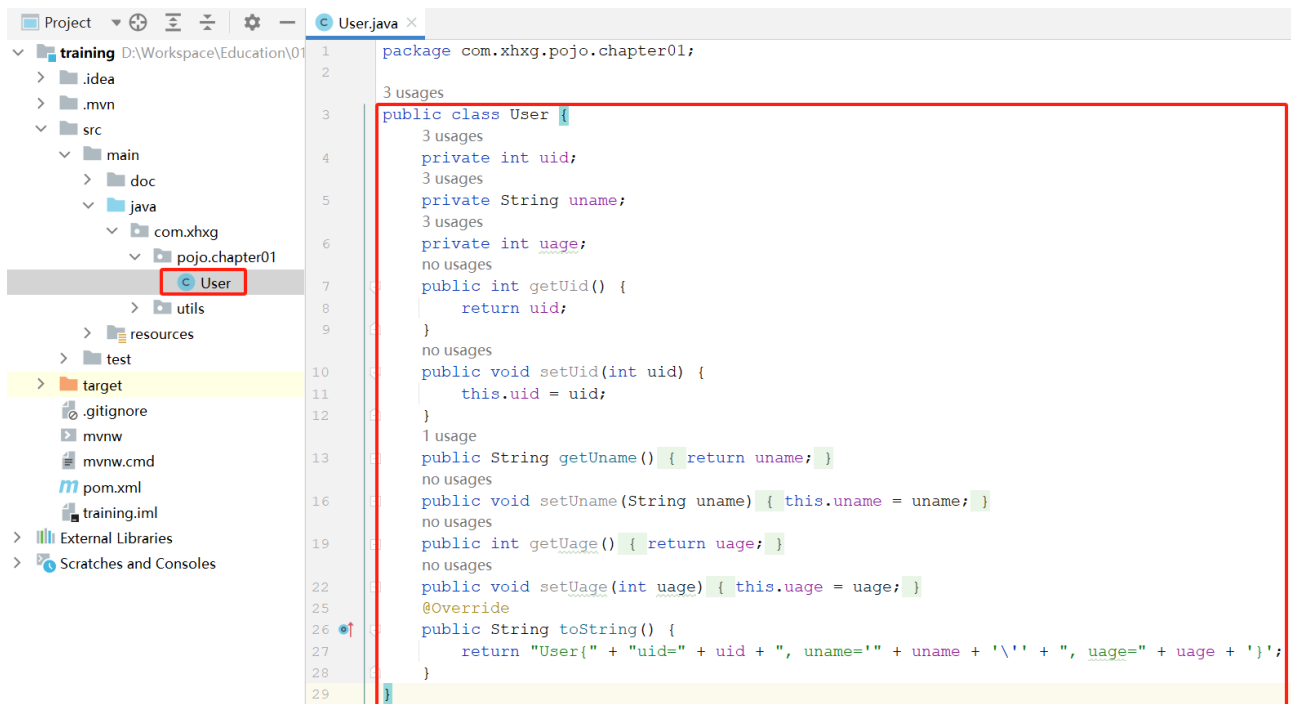


## 第 2 步：程序代码开发

### 1、创建 POJO 实体

打开前面创建的项目工程, 工程名称是 **training**。创建代码包为: `com.xhcg.pojo.chapter01`,

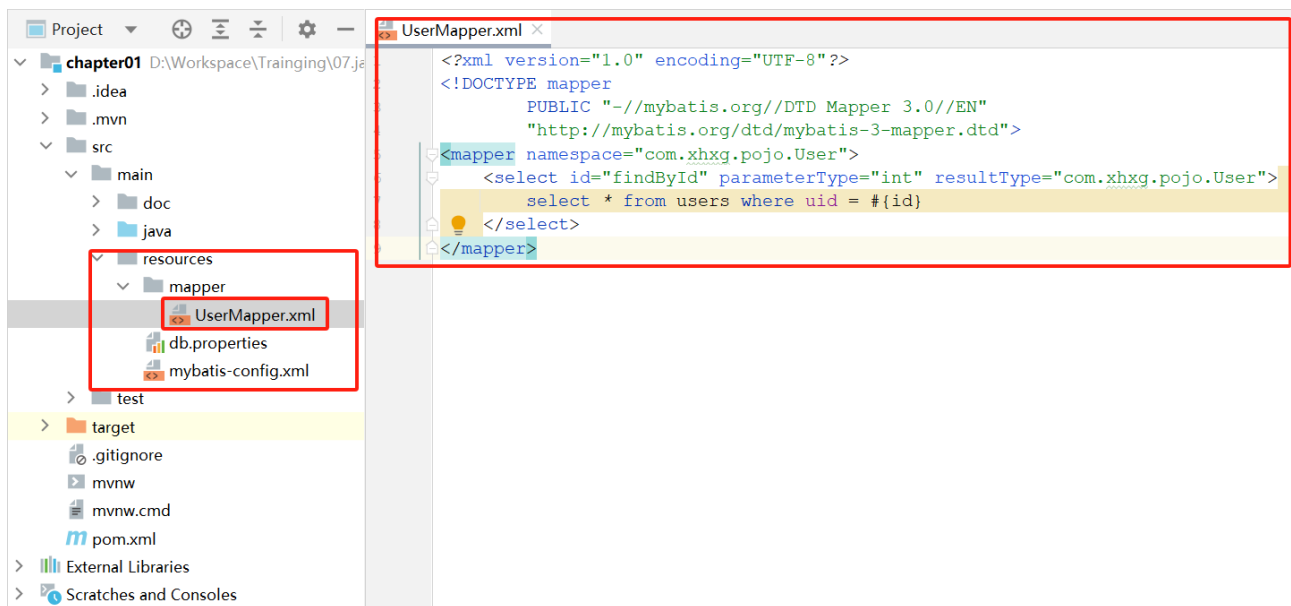
并且创建 `Users` 类



### 2、创建映射文件 UserMapper.xml

在 `src/main/resources` 目录下创建目录 `mapper`, 然后创建一个 `UserMapper.xml` 文件,

内容如下。



### 3、创建 mybatis-config.xml 配置文件

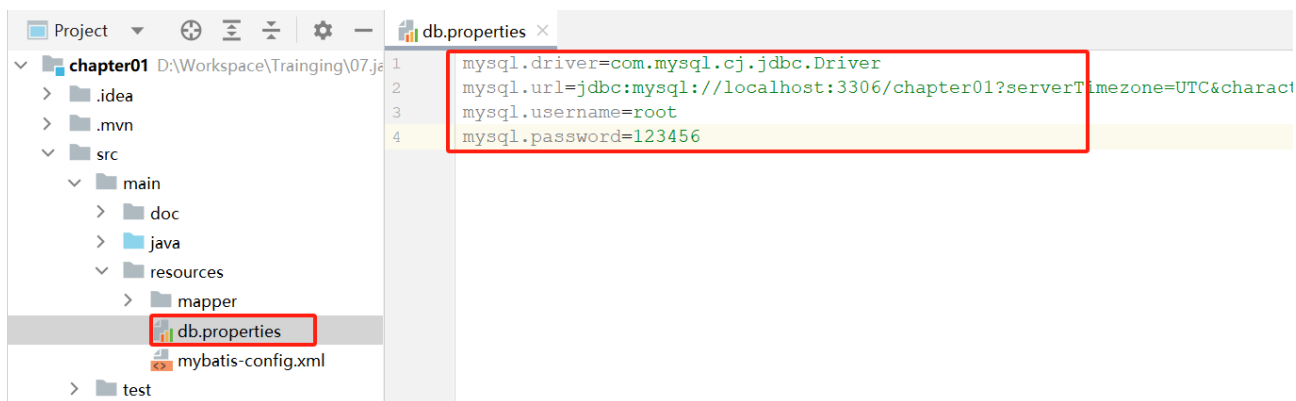
在 src/main/resources 目录下创建一个 mybatis-config.xml 文件，内容如下：



里面的配置项，需要创建另一个配置文件 db.properties 进行配置，所以我们跟着创建。

### 4、创建 db.properties 配置文件

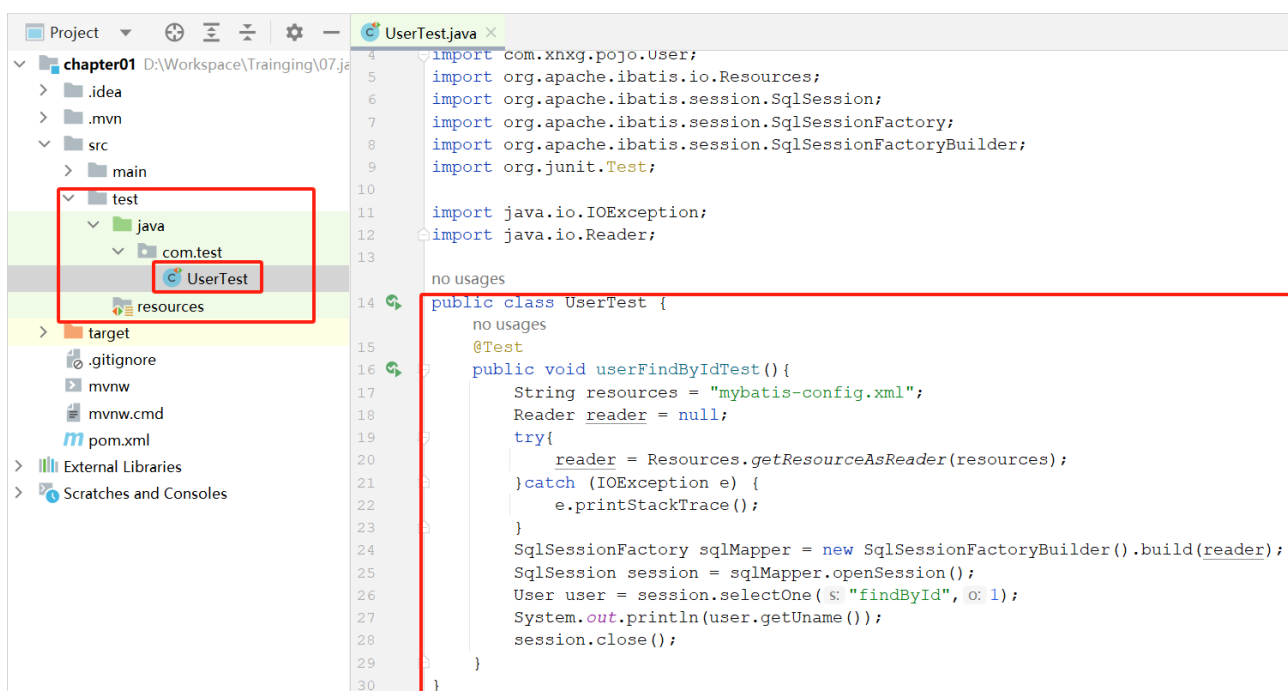
在 src/main/resources 目录下创建一个 db.properties 文件，内容如下：



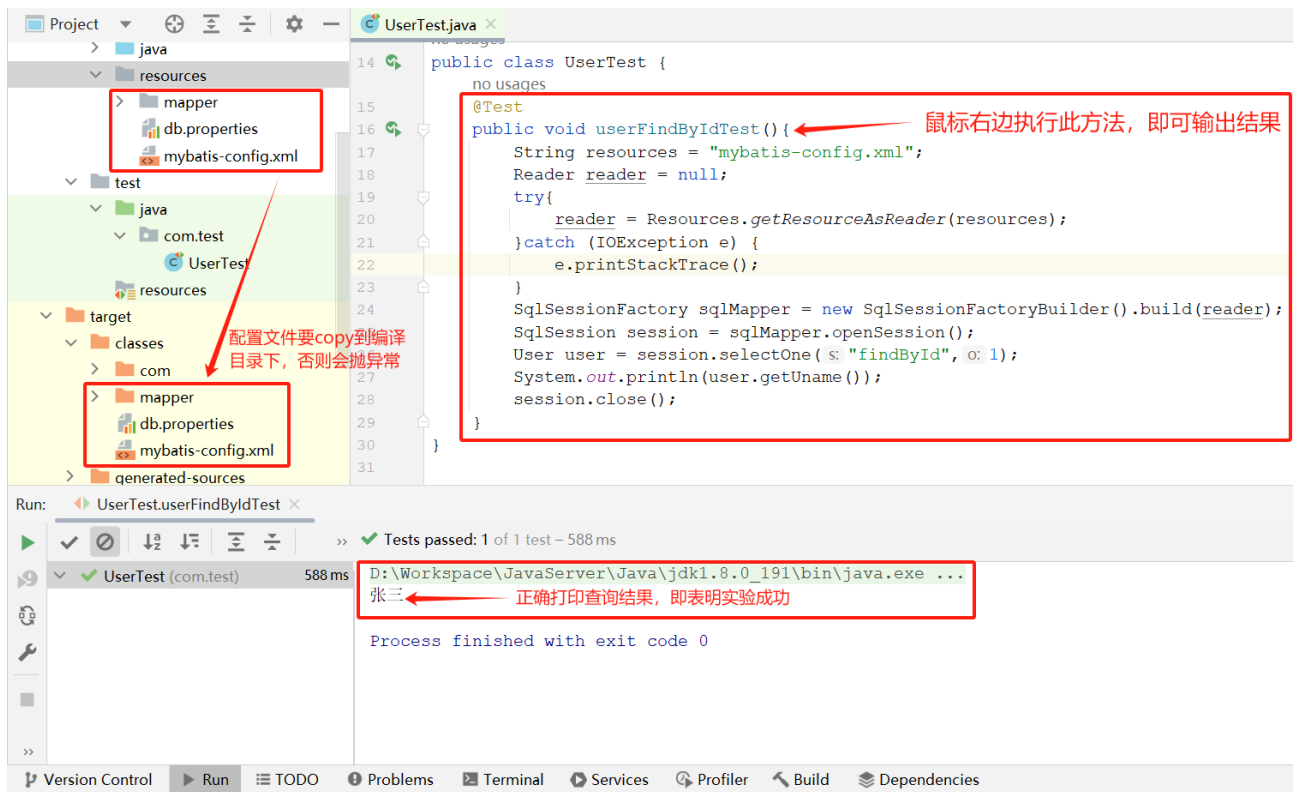
## 5、编写测试类

以上代码和配置创建好后，就可以编写测试类进行验证测试。

我们在 src/test/java 目录下，建议测试包，路径为：com.test，然后创建测试类 UserTest，测试的方法如下：



执行此单元测试后，如果正确会输出查询结果，如下图：



至此，本实验圆满成功，建立了 Java 代码与数据库的关联，验证了通过 Java 调用，正确获取数据库中的记录。

## 五、常见问题

有时候 Maven 不能正确的复制配置文件到 target 目录中，此时运行测试程序就会报如下异常：Could not find resource mybatis-config.xml

解决办法，就是手动将 src/main/resources/目录下的所有配置文件（包含目录）复制到 target/classes/目录下即可解决。

Project

- src
  - main
    - doc
    - java
    - resources
      - mapper
        - UserMapper.xml
        - db.properties
        - mybatis-config.xml
    - test
  - target
    - classes
      - com
      - xhcg

需要copy这些配置文件到classes目录下

```
public class UserTest {  
    no usages  
    @Test  
    public void userFindByIdTest() {  
        String resources = "mybatis-config.xml";  
        Reader reader = null;  
        try {  
            reader = Resources.getResourceAsReader(resources);  
        } catch (IOException e) {  
            e.printStackTrace();  
        }  
        SqlSessionFactory sqlMapper = new SqlSessionFactoryBuilder().build(reader);  
        SqlSession session = sqlMapper.openSession();  
        User user = session.selectOne(s: "findById", 0: 1);  
        System.out.println(user.getUserName());  
        session.close();  
    }  
}
```

测试程序

Run: UserTest.userFindByIdTest

Tests failed: 1 of 1 test - 35 ms

UserTest (com.test) 35 ms

userFindByIdTest 35 ms

D:\Workspace\JavaServer\Java\jdk1.8.0\_191\bin\java.exe ...  
java.io.IOException Create breakpoint : Could not find resource mybatis-config.xml  
at org.apache.ibatis.io.Resources.getResourceAsStream(Resources.java:114)  
at org.apache.ibatis.io.Resources.getResourceAsStream(Resources.java:100)  
at org.apache.ibatis.io.Resources.getResourceAsReader(Resources.java:160)  
at com.test.UserTest.userFindByIdTest(UserTest.java:20) <24 internal lines>  
异常信息  
java.lang.NullPointerException Create breakpoint  
at org.apache.ibatis.session.SqlSessionFactoryBuilder.build(SqlSessionFactoryBuilder.