南京邮电大学

实 验 报 告

（ 2024 / 2025 学年 第 二 学期）

课程名称	数据库系统基础
实验名称	实验一：MySQL 基础开发
实验时间	2025 年 4 月 2 日
指导单位	计算机学院、软件学院、网络空间安全学院
指导教师	黄楠

学生姓名	于明宏	班级学号	B23041011
学院(系)	计软院	专 业	信息安全

实 验 报 告

实验名称	实验一：MySQL 基础开发		
实验类型	验证	实验学时	2

一、 实验目的和要求

（1）掌握 MySQL 在 Windows 平台上的安装与卸载的方法，熟悉服务器的启动与停止，登录与断开，以及配置文件等；

（2）掌握 Navicat for MySQL 的安装与配置，实现客户端与服务器的连接；

（3）理解数据库对象及其作用，使用 SQL 数据定义语言（DDL）语句创建数据库和基本表；

（4）初步掌握数据的插入，查询，修改，删除方法；

（5）理解数据的完整性约束概念，掌握实现完整性约束的 SQL 子句

二、实验环境(实验设备)

硬件: 微型计算机

软件: Windows 操作系统、MySQL 5.6 或更高版本、Navicat for MySQL 15 或更高版本

三、实验原理及内容

1 MySQL 的安装和设置

1.1 下载并安装 MySQL

网址： https://mysql.com/

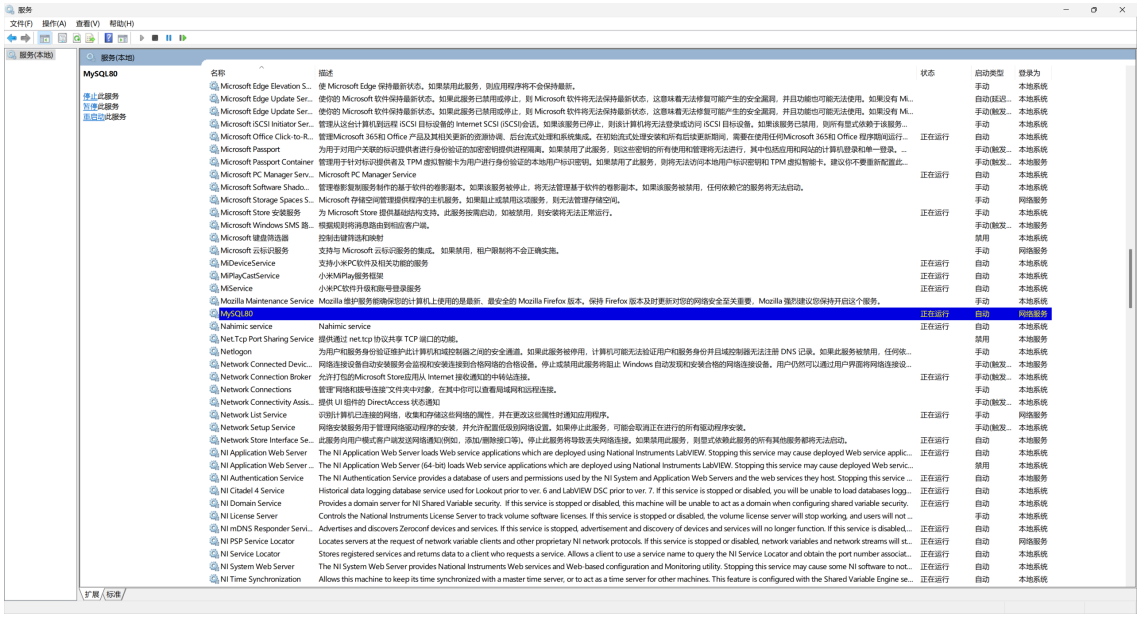
所下载版本及版本号：MySQL Community Server 8.0.33

1.2 MySQL 服务器的启动或停止

（1）通过系统服务管理器来启动或停止 MySQL 服务

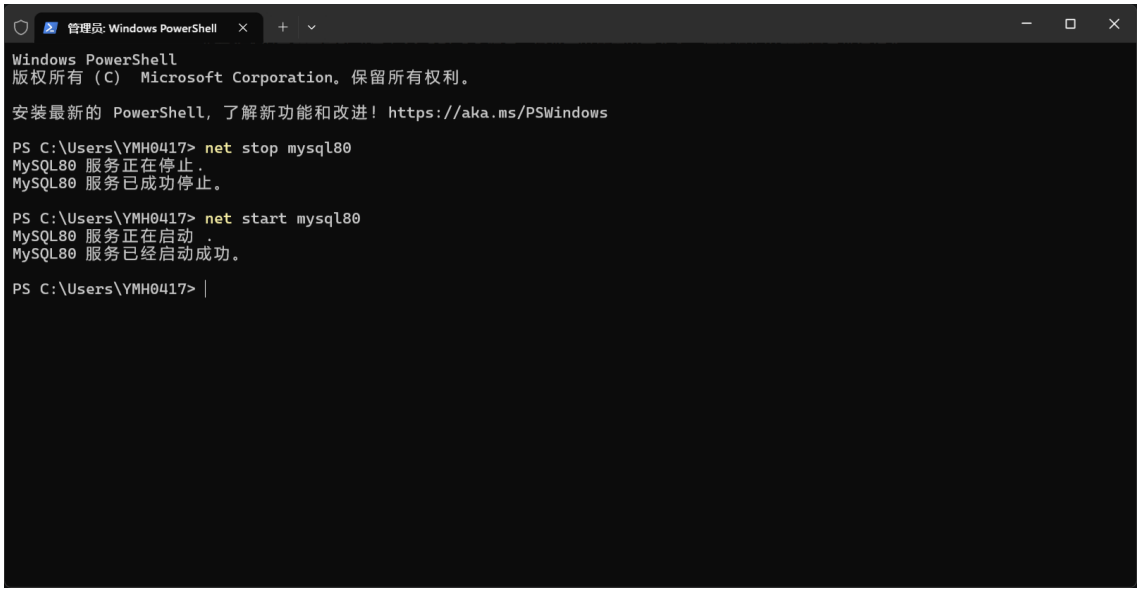
操作的截图结果是：

实 验 报 告



（2）在命令提示符下启动或停止 MySQL 服务

操作的截图结果是：



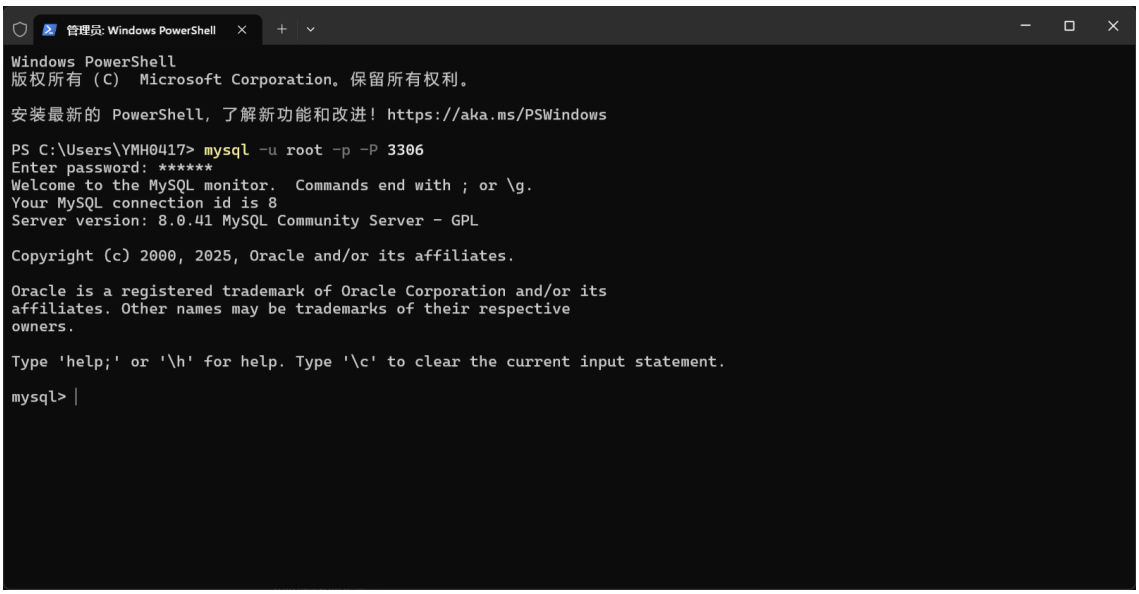
1.3 命令提示符登录 MySQL 服务器

登录程序：

```
mysql -u root -p -P 3306
```

程序的运行结果是：

实 验 报 告



2 Navicat for MySQL 的安装和配置

2.1 下载并安装 Navicat for MySQL

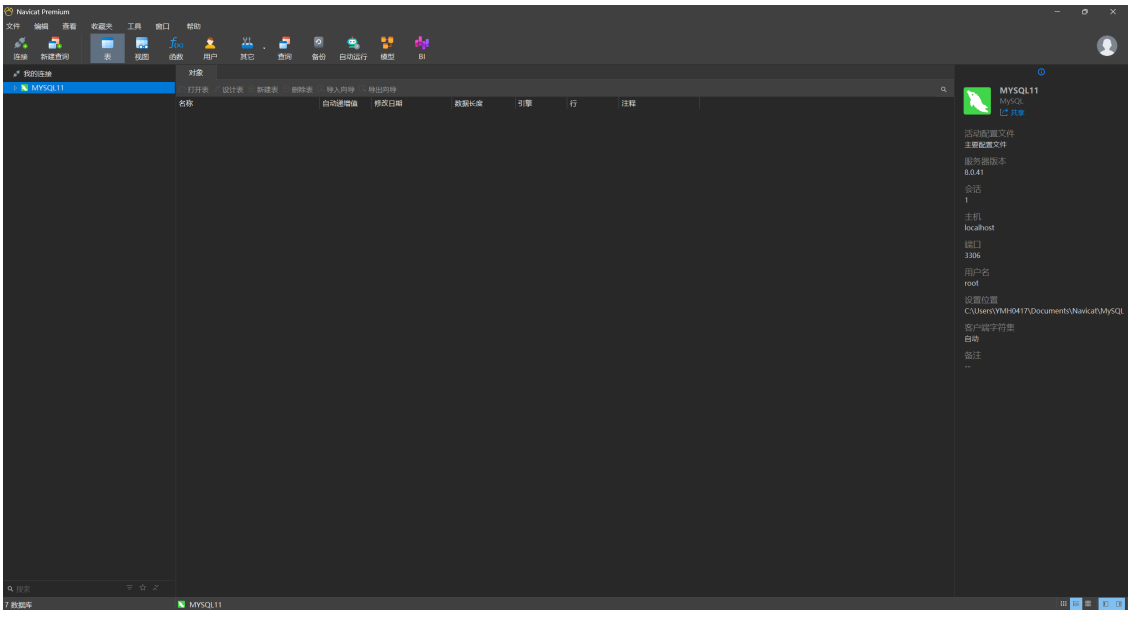
网址：https://navicat.com.cn/products/navicat-for-mysql

所下载版本及版本号：Navicat for MySQL 15

2.2 Navicat for MySQL 的配置

建立客户端程序与 MySQL 服务器之间的新连接，并命名为 MySQL+学号后两位，例如 MySQL01.

操作的截图结果是：



3 数据库的基本操作

3.1 创建数据库和数据表

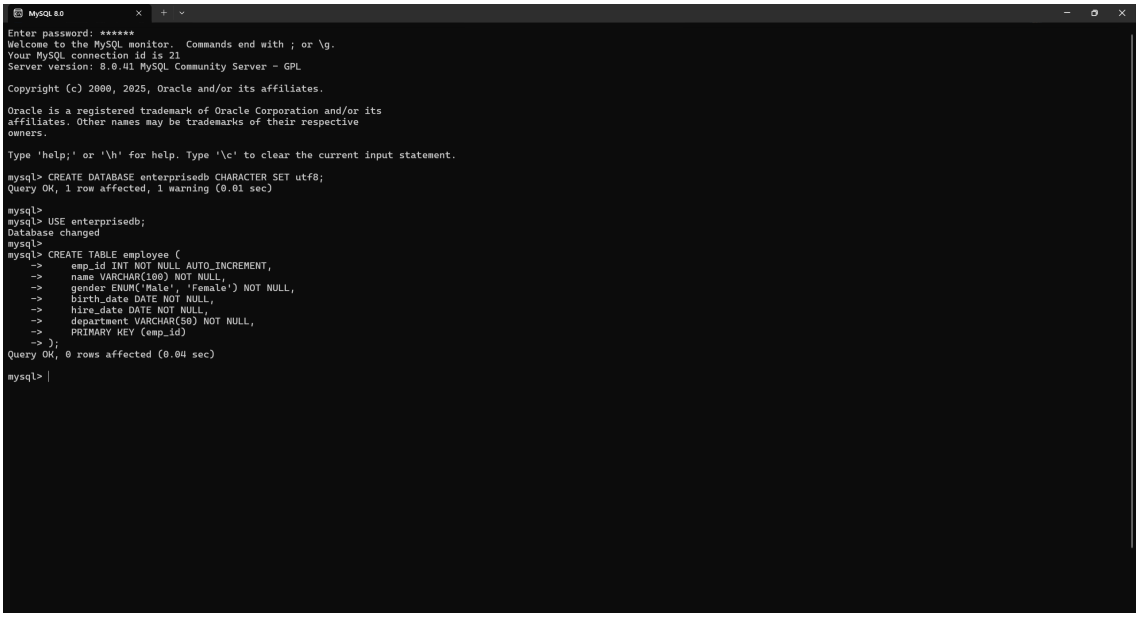
实 验 报 告

创建一个企业管理系统数据库，并命名为 enterprisedb，字符集选择 utf8；在 enterprisedb 库中创建职工表 employee，包括如下字段：职工编号（从 10001 开始递增）、姓名、性别、生日、入职日期、部门名称（例如财务部、人事部、规划部等），英文列名自拟，并根据字段自行选择合适的数据类型，给出 SQL 语句并输出截图：

```
CREATE DATABASE enterprisedb CHARACTER SET utf8;


USE enterprisedb;


CREATE TABLE employee (
    emp_id INT NOT NULL AUTO_INCREMENT,
    name VARCHAR(100) NOT NULL,
    gender ENUM('Male', 'Female') NOT NULL,
    birth_date DATE NOT NULL,
    hire_date DATE NOT NULL,
    department VARCHAR(50) NOT NULL,
    PRIMARY KEY (emp_id)
);
```

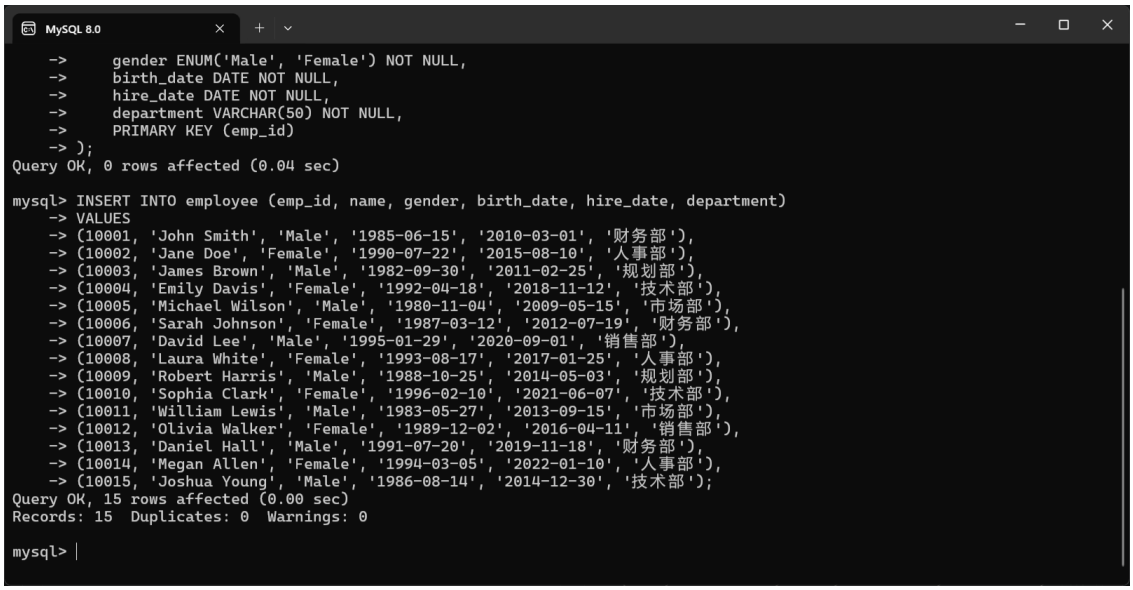


3.2 插入数据操作

（1）插入 15 条不同记录的职工信息，属性值自定，给出 SQL 语句并输出截图：

```
INSERT INTO employee (emp_id, name, gender, birth_date, hire_date, department)
VALUES
```

(10001, 'John Smith', 'Male', '1985-06-15', '2010-03-01', '财务部'),
(10002, 'Jane Doe', 'Female', '1990-07-22', '2015-08-10', '人事部'),
(10003, 'James Brown', 'Male', '1982-09-30', '2011-02-25', '规划部'),
(10004, 'Emily Davis', 'Female', '1992-04-18', '2018-11-12', '技术部'),
(10005, 'Michael Wilson', 'Male', '1980-11-04', '2009-05-15', '市场部'),
(10006, 'Sarah Johnson', 'Female', '1987-03-12', '2012-07-19', '财务部'),
(10007, 'David Lee', 'Male', '1995-01-29', '2020-09-01', '销售部'),
(10008, 'Laura White', 'Female', '1993-08-17', '2017-01-25', '人事部'),
(10009, 'Robert Harris', 'Male', '1988-10-25', '2014-05-03', '规划部'),
(10010, 'Sophia Clark', 'Female', '1996-02-10', '2021-06-07', '技术部'),
(10011, 'William Lewis', 'Male', '1983-05-27', '2013-09-15', '市场部'),
(10012, 'Olivia Walker', 'Female', '1989-12-02', '2016-04-11', '销售部'),
(10013, 'Daniel Hall', 'Male', '1991-07-20', '2019-11-18', '财务部'),
(10014, 'Megan Allen', 'Female', '1994-03-05', '2022-01-10', '人事部'),
(10015, 'Joshua Young', 'Male', '1986-08-14', '2014-12-30', '技术部');

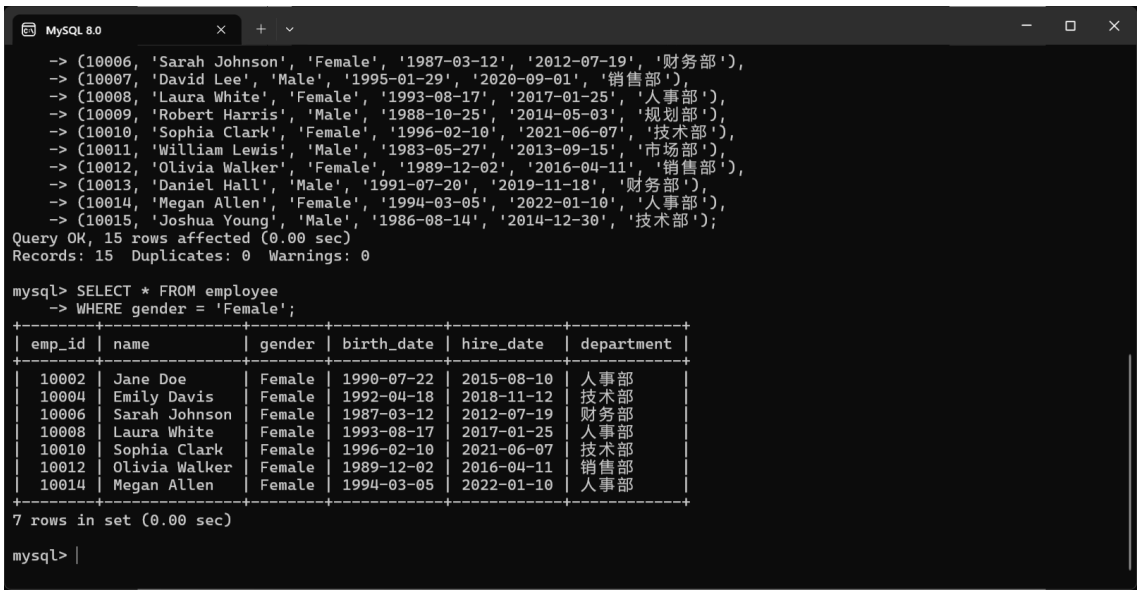


（2）显示所有女职工信息，给出 SQL 语句并输出截图：

SELECT * FROM employee

WHERE gender = 'Female';

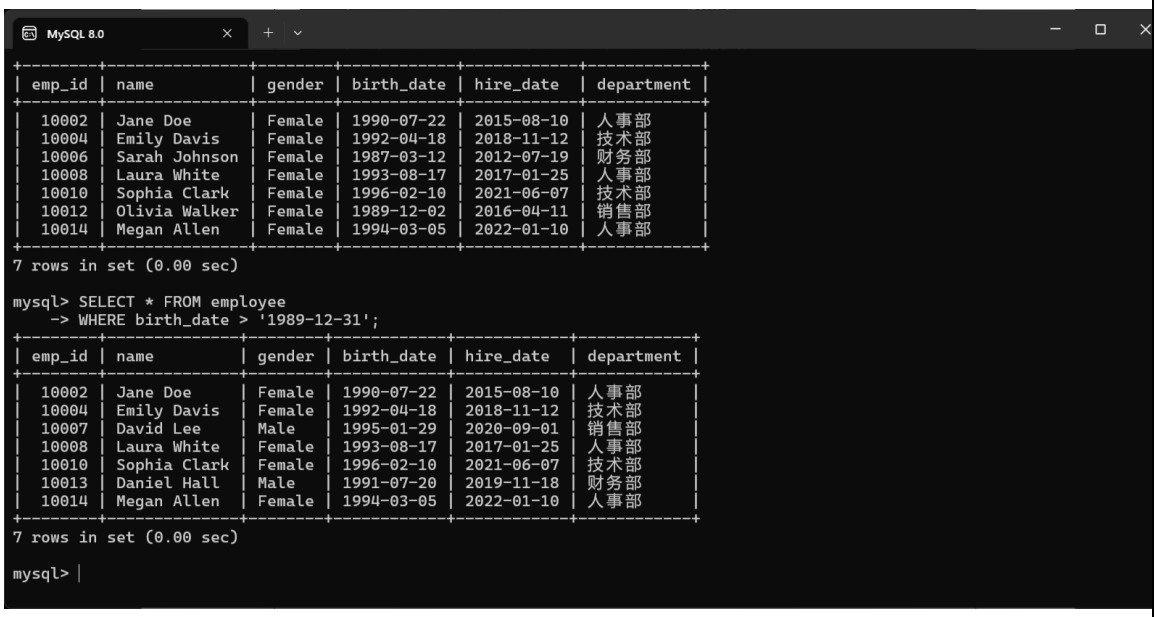
实 验 报 告



（3）显示所有 1989 年 12 月 31 日以后出生的职工信息，给出 SQL 语句并输出截图：

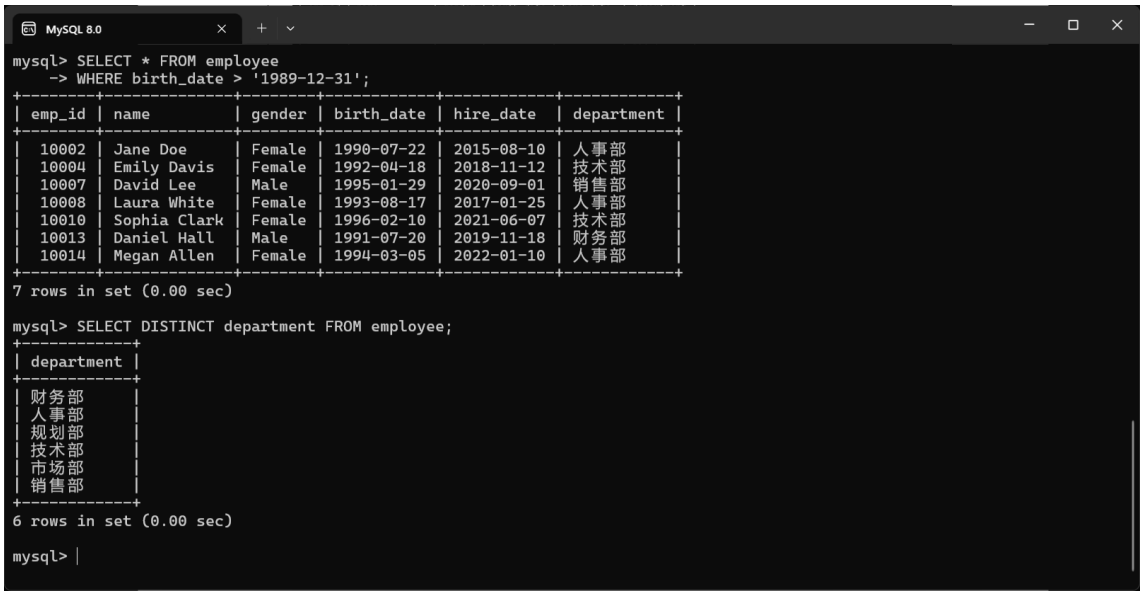
SELECT * FROM employee

WHERE birth_date > '1989-12-31';



（4）假设某职工所属部门不清楚，给出 SQL 语句显示可能的部门名称并输出截图：

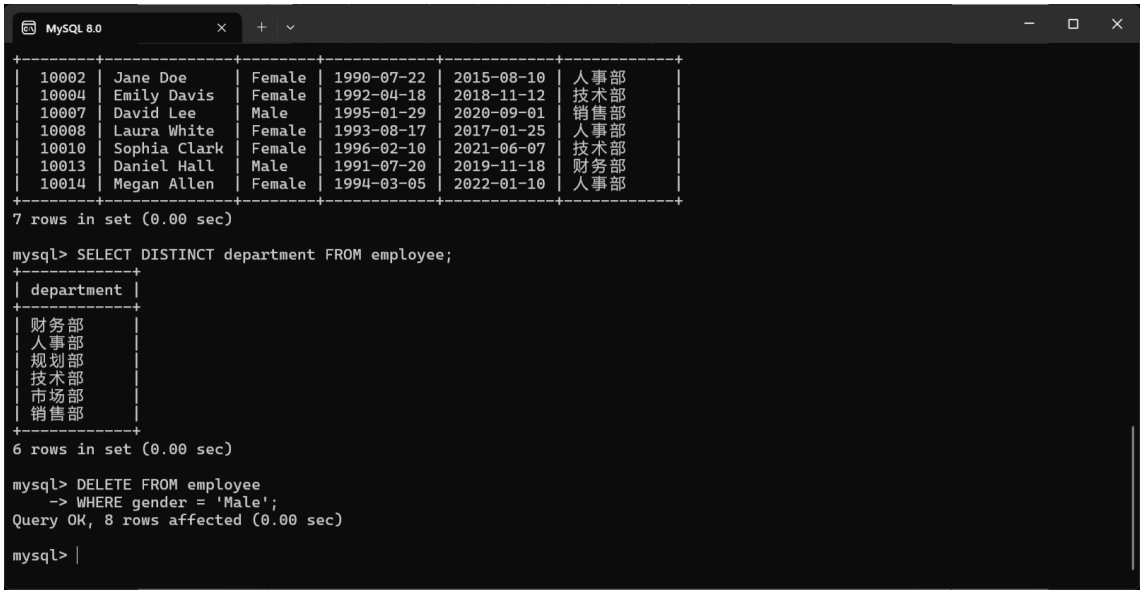
SELECT DISTINCT department FROM employee;



3.3 删除数据操作

（1）删除所有男职工的信息，给出 SQL 语句并输出截图：

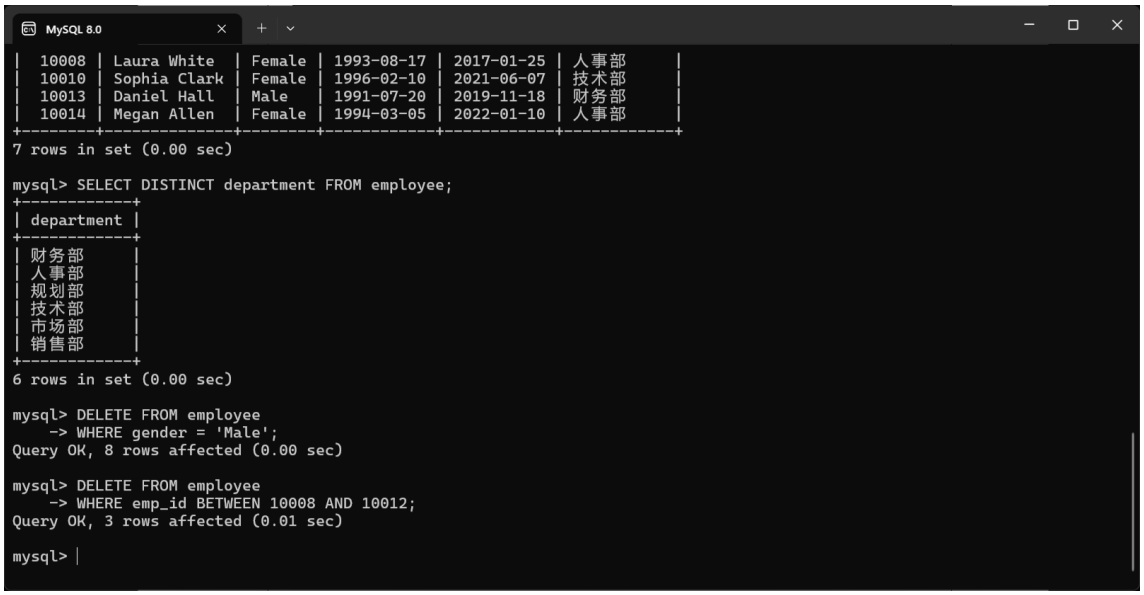
```
DELETE FROM employee
WHERE gender = 'Male';
```



（2）删除职工编号在 10008—10012 范围的职工信息，给出 SQL 语句并输出截图：

```
DELETE FROM employee
WHERE emp_id BETWEEN 10008 AND 10012;
```

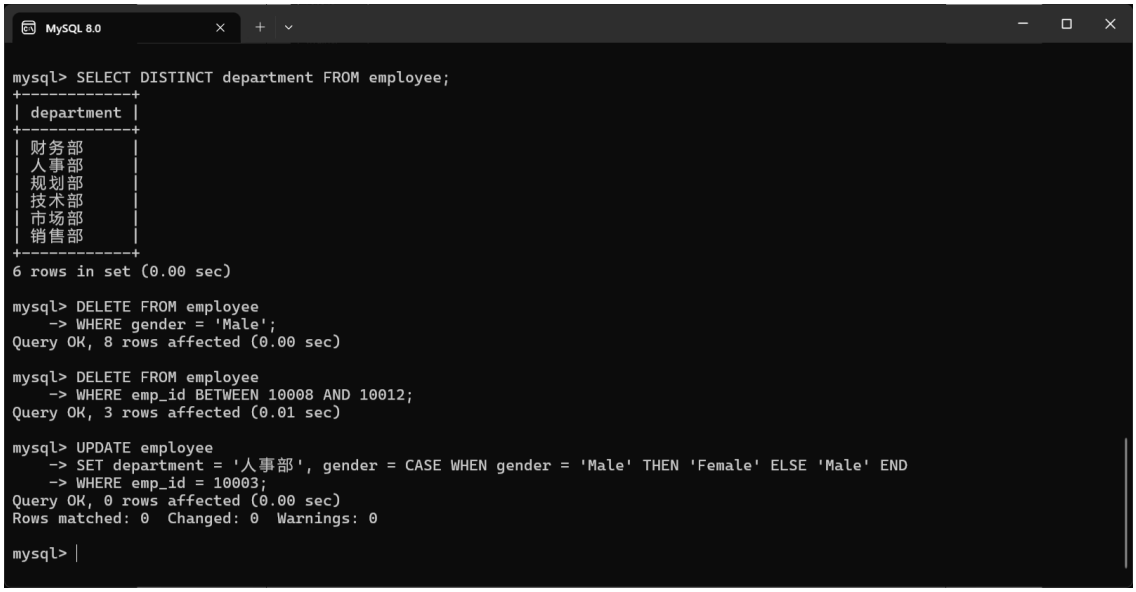
实 验 报 告



3.4 修改数据

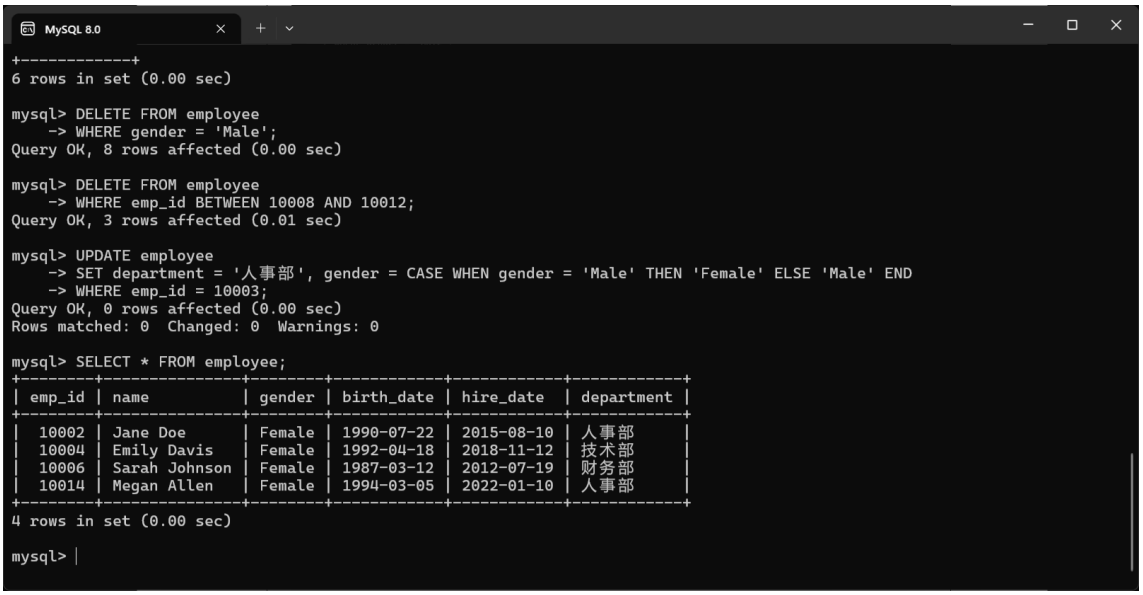
（1）修改职工编号 10003 的雇员，部门为“人事部”，性别为相反性别，给出 SQL 语句并输出截图：

```
UPDATE employee
SET department = '人事部', gender = CASE WHEN gender = 'Male' THEN 'Female' ELSE 'Male' END
WHERE emp_id = 10003;
```



（2）显示所有职工信息，给出 SQL 语句并输出截图：

```
SELECT * FROM employee;
```



3.5 数据完整性约束

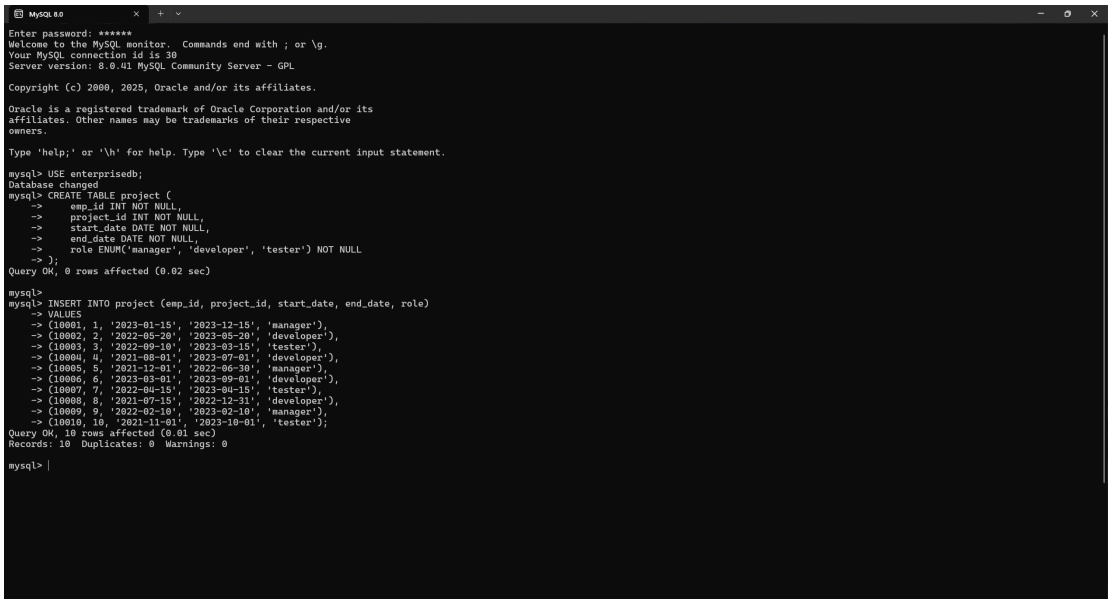
新建一个职工参与项目表 project，包含职工编号、参与项目号、项目开始时间、项目结束时间，以及在项目担任的角色（manager、developer、tester 等），数据类型根据实际情况自拟，并插入 10 条记录信息，给出 SQL 语句并输出截图：

（此处已将其恢复为 3.2 的状态）

```
CREATE TABLE project (
    emp_id INT NOT NULL,
    project_id INT NOT NULL,
    start_date DATE NOT NULL,
    end_date DATE NOT NULL,
    role ENUM('manager', 'developer', 'tester') NOT NULL
);

INSERT INTO project (emp_id, project_id, start_date, end_date, role)
VALUES
(10001, 1, '2023-01-15', '2023-12-15', 'manager'),
(10002, 2, '2022-05-20', '2023-05-20', 'developer'),
(10003, 3, '2022-09-10', '2023-03-15', 'tester'),
(10004, 4, '2021-08-01', '2023-07-01', 'developer'),
```

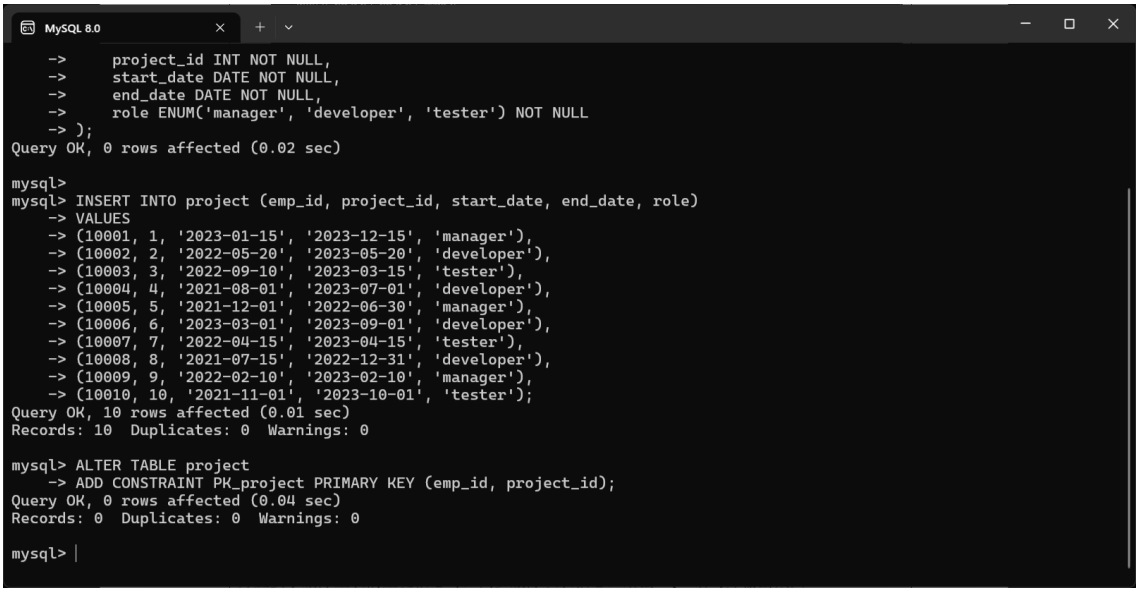
```
(10005, 5, '2021-12-01', '2022-06-30', 'manager'),
(10006, 6, '2023-03-01', '2023-09-01', 'developer'),
(10007, 7, '2022-04-15', '2023-04-15', 'tester'),
(10008, 8, '2021-07-15', '2022-12-31', 'developer'),
(10009, 9, '2022-02-10', '2023-02-10', 'manager'),
(10010, 10, '2021-11-01', '2023-10-01', 'tester');
```



（1）以表级完整性约束方式添加主键，并指定主键约束名称为 PK_ project，给出 SQL 语句并输出截图：

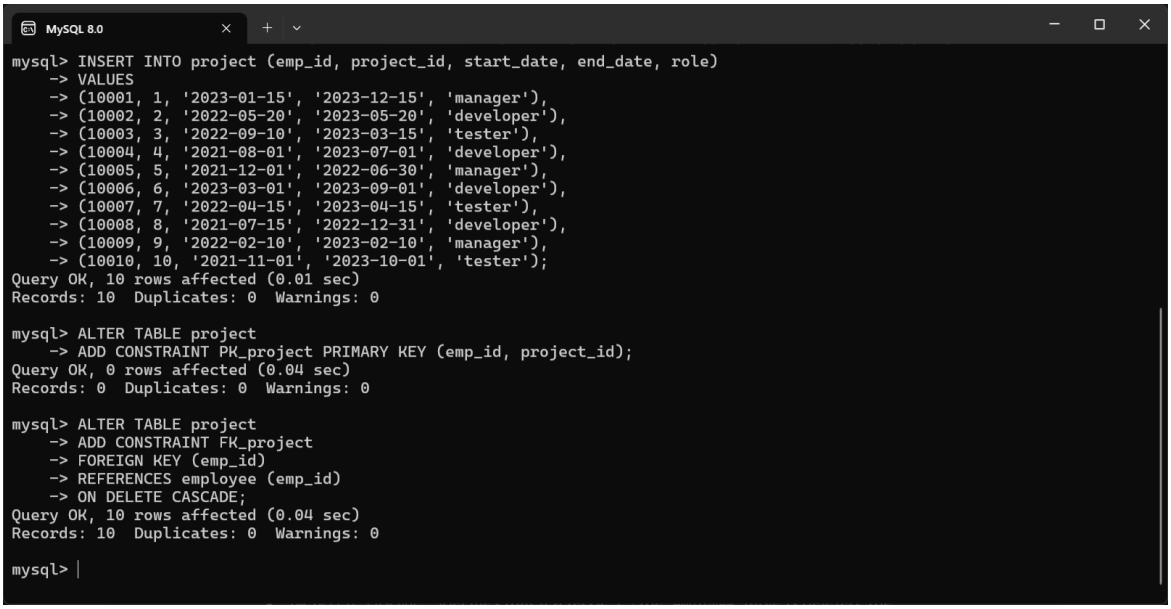
```
ALTER TABLE project
ADD CONSTRAINT PK_project PRIMARY KEY (emp_id, project_id);
```

实 验 报 告



（2）通过 ALTER TABLE 语句添加与 employee 表的外键约束，外键约束名称为 FK_ project，并且要求删除主表的主键信息时，同步删除从表信息，给出 SQL 语句并输出截图：

```
ALTER TABLE project
ADD CONSTRAINT FK_project
FOREIGN KEY (emp_id)
REFERENCES employee (emp_id)
ON DELETE CASCADE;
```



（3）在 project 表中删除已定义的外键约束，给出 SQL 语句：

```
ALTER TABLE project
```

```
DROP FOREIGN KEY FK_project;
```

实 验 报 告

四、实验小结（包括问题和解决方法、心得体会、意见与建议等）

（一） 实验中遇到的主要问题及解决方法

无。

（二）实验心得

通过本次实验，我掌握了 MySQL 数据库的基本操作，包括安装配置、数据库与表的创建、数据的增删改查以及完整性约束的实现。在实验过程中，我熟悉了 Navicat for MySQL 的使用，并通过实践加深了对 SQL 语句的理解，尤其是外键约束和级联操作的应用。此外，通过完成不同需求的查询和修改任务，我进一步认识到数据完整性的重要性以及约束条件在实际场景中的作用。本次实验不仅巩固了课堂所学知识，也提升了我的数据库操作能力和问题解决能力，为后续学习更复杂的数据库应用打下了坚实基础。

（三）意见与建议（没有可省略）

无。

实 验 报 告

五、支撑毕业要求指标点

☑ 4.2-M 能够根据实验方案，配置实验环境、开展实验，综合分析实验结果以获得合理有效的结论。

☑ 5.2-M 能够针对计算机及应用领域中的复杂工程问题，合理选择使用恰当的技术、资源和现代工程工具进行预测和模拟，并理解其局限性。

六、指导教师评语

	评分项	优秀	良好	中等	合格	不合格
评分细则	遵守实验室规章制度					
	学习态度					
	算法思想准备情况					
	程序设计能力					
	解决问题能力					
	算法设计合理性					
	算法效能评价					
	报告书写认真程度					
	内容详实程度					
	文字表达熟练程度					
	其它评价意见					
	本次实验能力达成评价（总成绩）		批阅人		日期	