西安欧亚学院2024－2025学年第一学期课程设计说明

**专业：软件工程专升本**

**课程：数据库系统与应用 年级：2024级**

**一、背景**

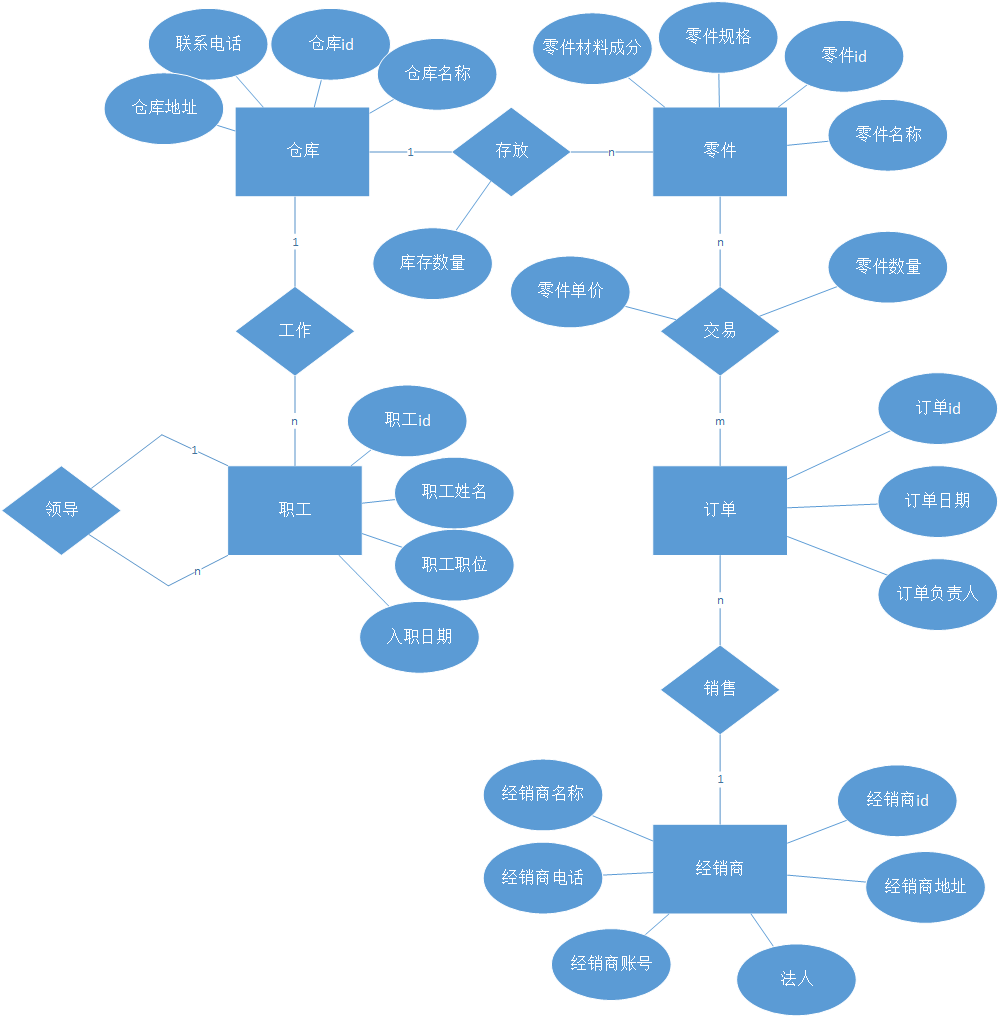
某个工厂物资管理处进行数字化转型，它有若干个大小不一的仓库，仓库之间可以电话联系，仓库中存放的是工厂中生产的不同名称，规格，材料构成的零件，这些零件是卖给全国的不同的经销商的，物资管理科会备有这些经销商的基本信息，如名称，地址，电话，法人，账号，工厂为多个经销商提供的是货物零件，通过一个订单号，可以查询这个单号中所包含的零件名称，数量，单价，订单负责人等信息，仓库也有专门的人员负责，

一个仓库可以存放多种零件，一种零件可以存放在多个仓库中，用库存量来表示某种零件在某个仓库中的数量；一个仓库有多个职工当仓库保管员，一个职工只能在一个仓库工作；职工之间具有领导与被领导关系，即仓库主任领导若干保管员；仓库为经销商提供若干不同的零件，依据订单号可以查询详细的订单信息。

**二、任务**

**请根据以上的背景信息，完成**

1. **ER模型设计，要求用工具画出ER模型，可以对ER模型作必要的说明。ER模型为一个整体的图，不需要各个实体分开。**



**2、逻辑模型设计，要求给出实体表的名称，表属性，数据类型，约束，备注说明等要素（建议用表格形式给出，要给出主键，外键等约束）**

**1. 仓库表 (warehouses)**

| **字段名** | **数据类型** | **描述** |
| --- | --- | --- |
| **warehouse\_id** | **INT** | **仓库ID (主键)** |
| **warehouse\_name** | **VARCHAR(255)** | **仓库名称** |
| **warehouse\_phone** | **VARCHAR(20)** | **仓库联系电话** |
| **warehouse\_address** | **VARCHAR(255)** | **仓库地址** |

**2. 零件表 (parts)**

| **字段名** | **数据类型** | **描述** |
| --- | --- | --- |
| **part\_id** | **INT** | **零件ID (主键)** |
| **part\_name** | **VARCHAR(255)** | **零件名称** |
| **specification** | **VARCHAR(255)** | **零件规格** |
| **material\_composition** | **VARCHAR(255)** | **零件的材料成分** |

**3. 经销商表 (distributors)**

| **字段名** | **数据类型** | **描述** |
| --- | --- | --- |
| **distributor\_id** | **INT** | **经销商ID (主键)** |
| **distributor\_name** | **VARCHAR(255)** | **经销商名称** |
| **address** | **VARCHAR(255)** | **经销商地址** |
| **phone** | **VARCHAR(20)** | **经销商电话** |
| **legal\_person** | **VARCHAR(100)** | **法人** |
| **account\_number** | **VARCHAR(50)** | **账户号** |

**4. 订单表 (orders)**

| **字段名** | **数据类型** | **描述** |
| --- | --- | --- |
| **order\_id** | **INT** | **订单ID (主键)** |
| **distributor\_id** | **INT** | **经销商ID (外键)** |
| **order\_date** | **DATE** | **订单日期** |
| **order\_responsible** | **VARCHAR(255)** | **订单负责人** |

**5. 订单零件关联表 (order\_parts)**

| **字段名** | **数据类型** | **描述** |
| --- | --- | --- |
| **order\_id** | **INT** | **订单ID (外键)** |
| **part\_id** | **INT** | **零件ID (外键)** |
| **quantity** | **INT** | **零件数量** |
| **unit\_price** | **DECIMAL(10, 2)** | **零件单价** |

**6. 职工表 (employees)**

| **字段名** | **数据类型** | **描述** |
| --- | --- | --- |
| **employee\_id** | **INT** | **职工ID (主键)** |
| **employee\_name** | **VARCHAR(255)** | **职工姓名** |
| **position** | **VARCHAR(50)** | **职工职位（如仓库主任、保管员）** |
| **warehouse\_id** | **INT** | **所属仓库ID (外键)** |
| **supervisor\_id** | **INT** | **领导职工ID (外键)** |

**7. 仓库零件库存表 (warehouse\_inventory)**

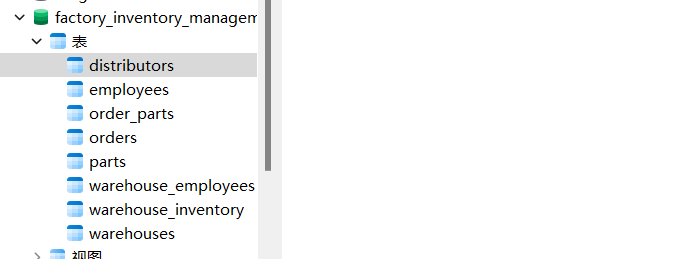
| **字段名** | **数据类型** | **描述** |
| --- | --- | --- |
| **warehouse\_id** | **INT** | **仓库ID (外键)** |
| **part\_id** | **INT** | **零件ID (外键)** |
| **stock\_quantity** | **INT** | **库存数量** |

**8. 仓库职工关联表 (warehouse\_employees)**

| **字段名** | **数据类型** | **描述** |
| --- | --- | --- |
| **warehouse\_id** | **INT** | **仓库ID (外键)** |
| **employee\_id** | **INT** | **职工ID (外键)** |

**3、问题操作，要有代码及结果截图，注意载图要清晰。**

3.1创建你所设计的表（截一个图即可）。



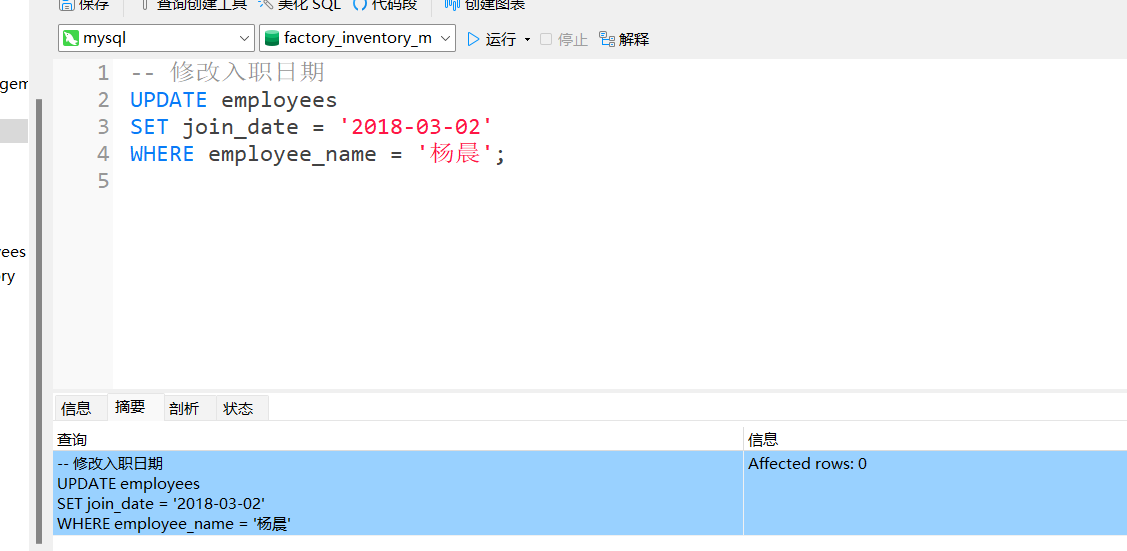
3.2模拟3条关于职工的数据，插入到你的表中，职工数据中要包含自己的姓名。（代码与结果截一个图即可）

3.3 模拟对职工信息的修改与删除操作，如将你的入职日期修改到2018年3月2日，删除掉一条数据。

-- 修改入职日期

UPDATE employees

SET join\_date = '2018-03-02'

WHERE employee\_name = '杨晨';

-- 删除一条数据

DELETE FROM employees

WHERE employee\_id = 3; -- 假设删除 ID 为 3 的职工



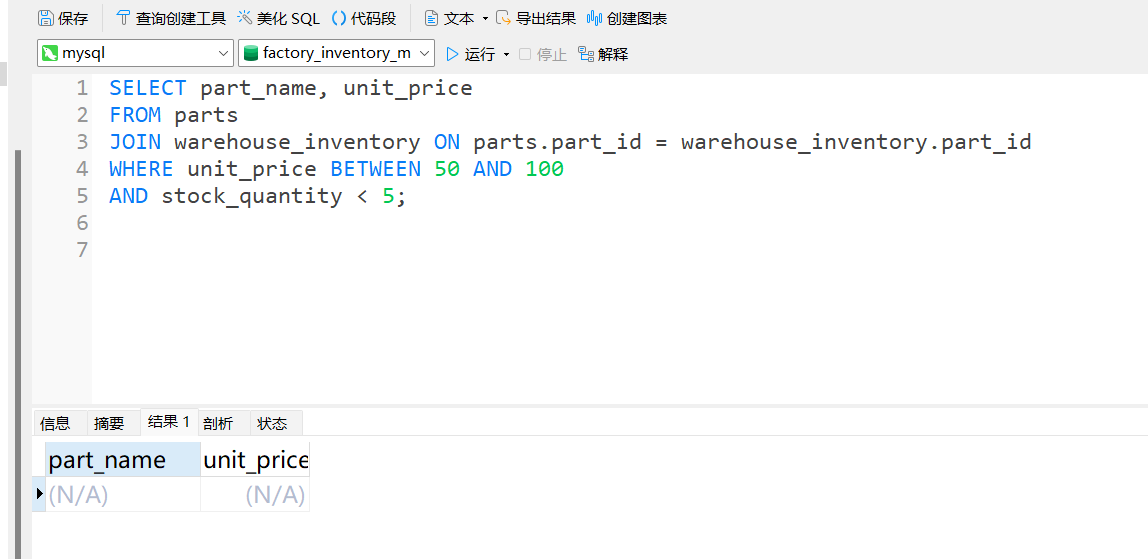
3.4查询标价在50和100之间且库存量小于5的所有产品名称和价格。

SELECT part\_name, unit\_price

FROM parts

JOIN warehouse\_inventory ON parts.part\_id = warehouse\_inventory.part\_id

WHERE unit\_price BETWEEN 50 AND 100

AND stock\_quantity < 5;

3.5查询各订单的编号和总金额，并要求订单总金额大于10000元且按照订单日期降序排列。

SELECT orders.order\_id, SUM(order\_parts.quantity \* order\_parts.unit\_price) AS total\_amount

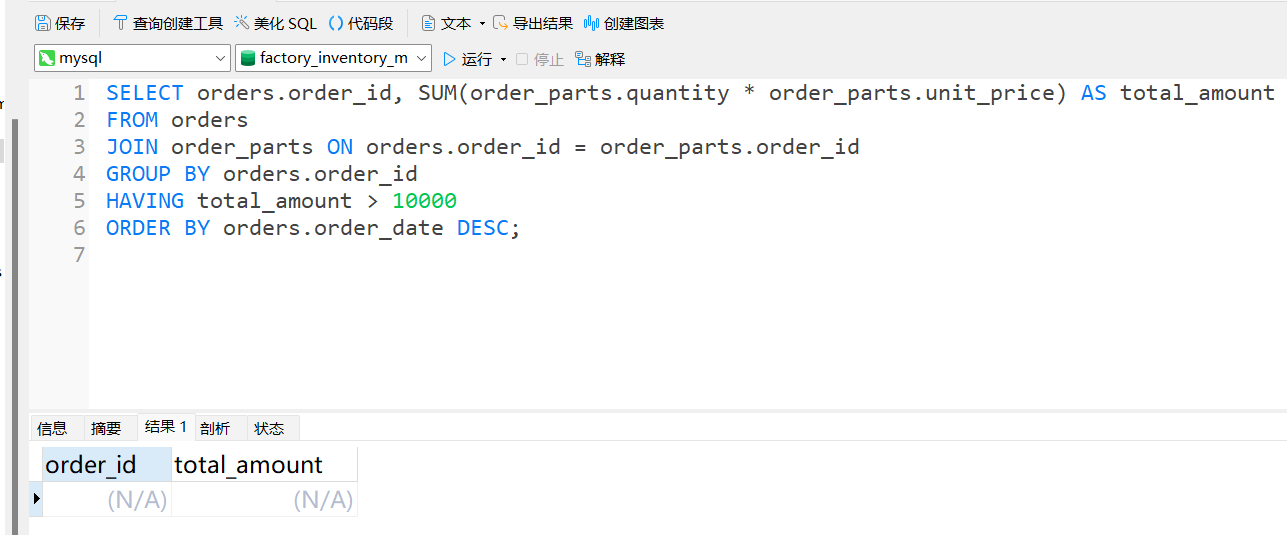
FROM orders

JOIN order\_parts ON orders.order\_id = order\_parts.order\_id

GROUP BY orders.order\_id

HAVING total\_amount > 10000

ORDER BY orders.order\_date DESC;



3.6查询销售日期在2018-11-15之前的订单编号、产品名称、经销商姓名、经销商电话及订单日期。

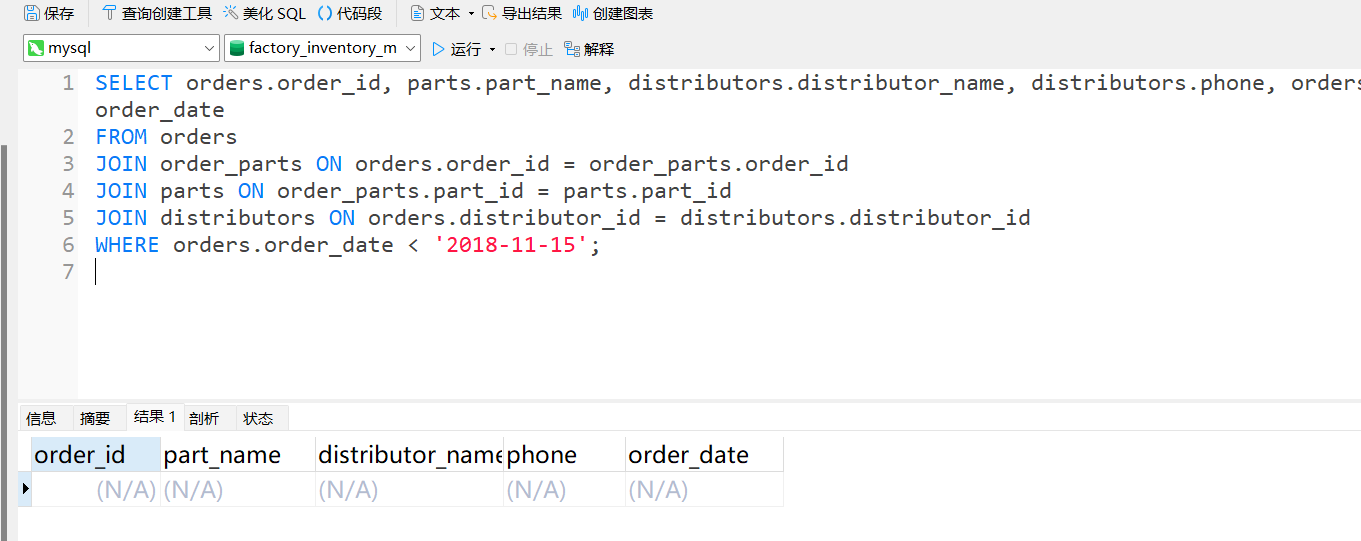
SELECT orders.order\_id, parts.part\_name, distributors.distributor\_name, distributors.phone, orders.order\_date

FROM orders

JOIN order\_parts ON orders.order\_id = order\_parts.order\_id

JOIN parts ON order\_parts.part\_id = parts.part\_id

JOIN distributors ON orders.distributor\_id = distributors.distributor\_id

WHERE orders.order\_date < '2018-11-15';

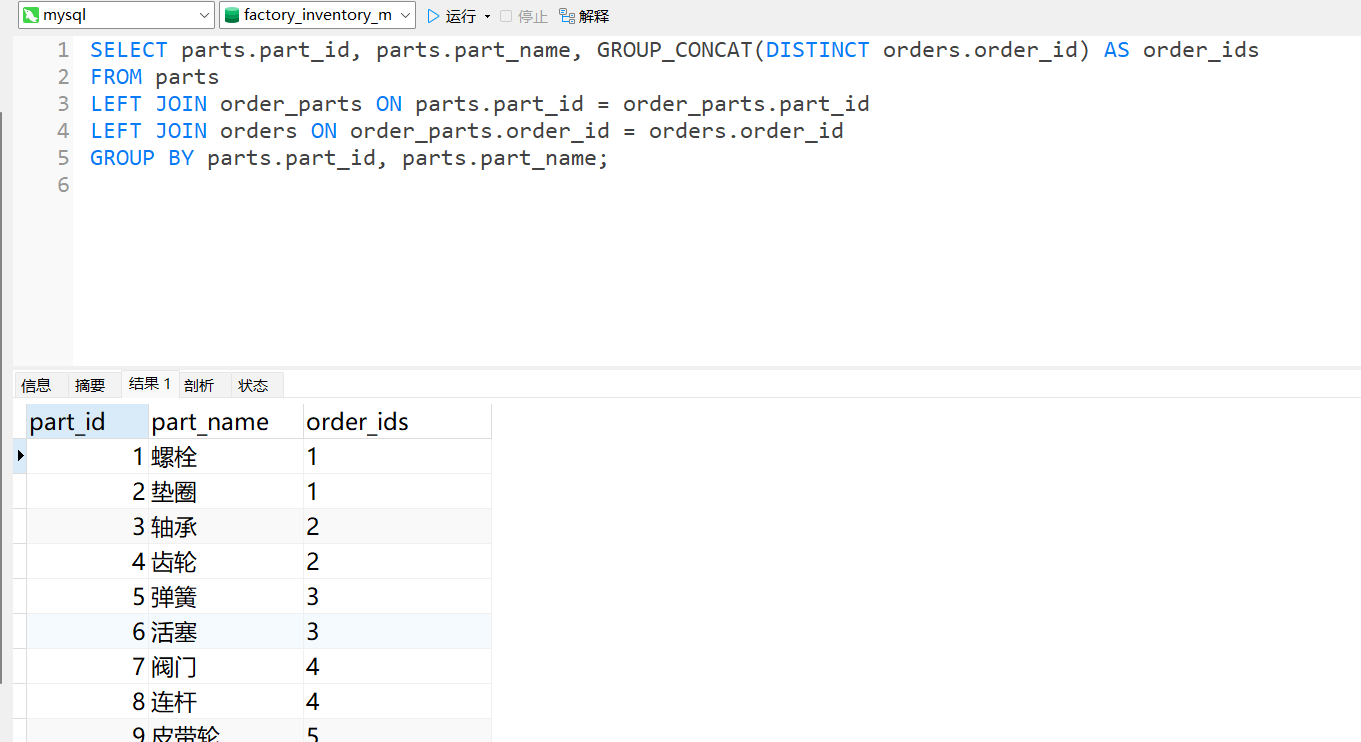
3.7查询所有的产品编号、产品名称及含有该产品的订单编号，如果产品没有销售也要列出产品编号、产品名称。

SELECT parts.part\_id, parts.part\_name, GROUP\_CONCAT(DISTINCT orders.order\_id) AS order\_ids

FROM parts

LEFT JOIN order\_parts ON parts.part\_id = order\_parts.part\_id

LEFT JOIN orders ON order\_parts.order\_id = orders.order\_id

GROUP BY parts.part\_id, parts.part\_name;

3.8在经销息表中为供应商名称创建唯一性索引index\_sailer。

3.9请使用SQL语句查询当前用户是否具有select\_priv和create\_view\_priv权限，如果没有，请修改权限使其具有创建视图的权限.

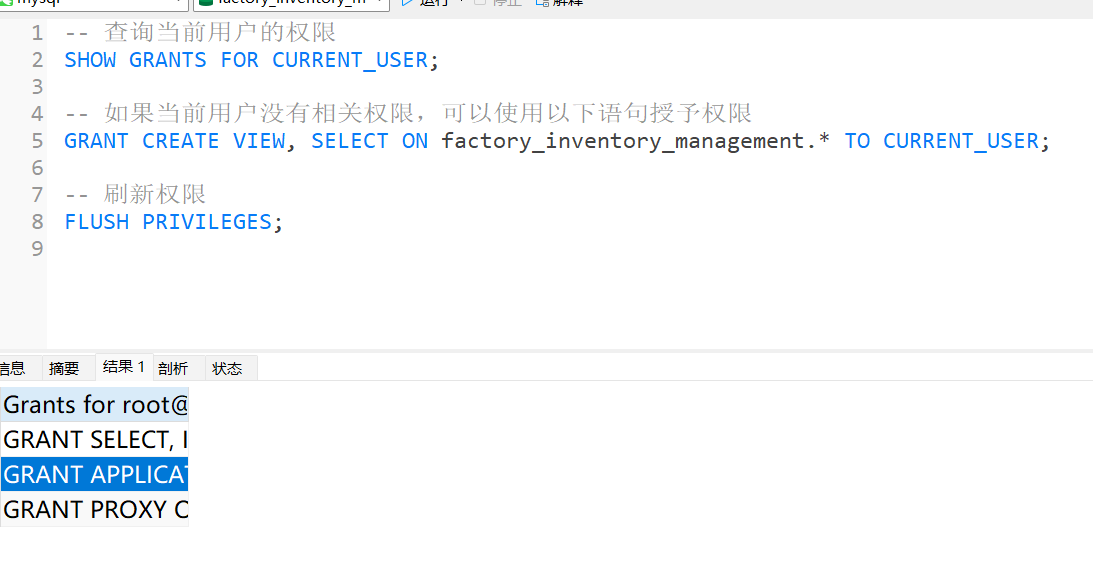
-- 查询当前用户的权限

SHOW GRANTS FOR CURRENT\_USER;

-- 如果当前用户没有相关权限，可以使用以下语句授予权限

GRANT CREATE VIEW, SELECT ON factory\_inventory\_management.\* TO CURRENT\_USER;

-- 刷新权限

FLUSH PRIVILEGES;

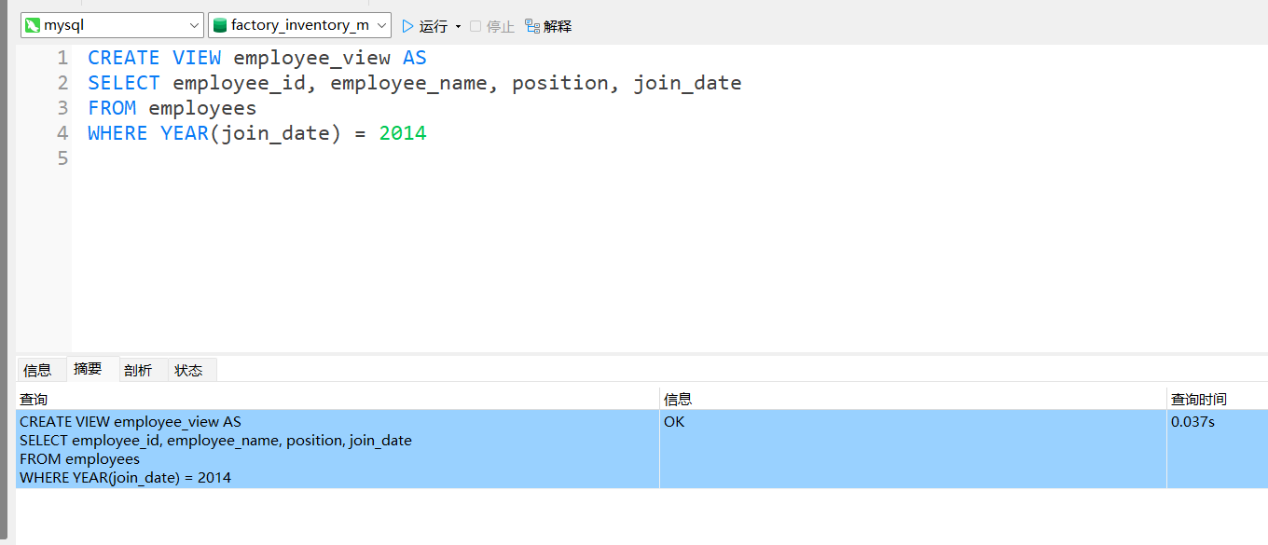
3.10创建视图employee\_view，包含所有入职日期是2014年的女职工的工号、姓名、出生日期、职位、入职日期.

CREATE VIEW employee\_view AS

SELECT employee\_id, employee\_name, position, join\_date

FROM employees

WHERE YEAR(join\_date) = 2014



3.11设计存储过程p\_products，返回指定产品编号的产品名称、产品单价，调用存储过程进行测试（2分）。

-- 创建存储过程

DELIMITER $$

CREATE PROCEDURE p\_products(IN part\_id INT)

BEGIN

SELECT part\_name, unit\_price

FROM parts

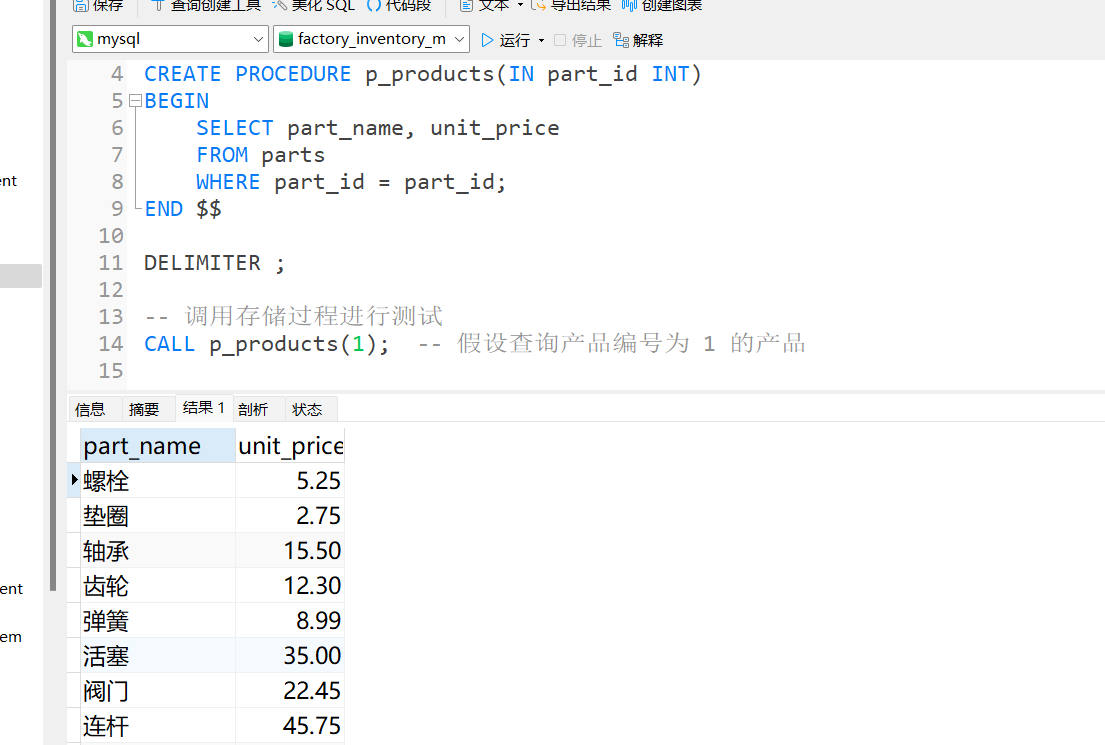
WHERE part\_id = part\_id;

END $$

DELIMITER ;

-- 调用存储过程进行测试

CALL p\_products(1); -- 假设查询产品编号为 1 的产品

3.12创建用户账号，用户名是自己的姓名\_班级，如li’ang\_2301，主机名是localhost，密码是自己学号后3位，给这个用户授予其对本数据库除了经销商表之外的所有表的的查询权限及修改权限。

-- 创建用户

CREATE USER '杨晨\_2402'@'localhost' IDENTIFIED BY '319';

-- 授予对除了经销商表(distributors)之外的所有表的查询和修改权限

GRANT SELECT, INSERT, UPDATE, DELETE ON factory\_inventory\_management.warehouses TO '杨晨\_2402'@'localhost';

GRANT SELECT, INSERT, UPDATE, DELETE ON factory\_inventory\_management.parts TO '杨晨\_2402'@'localhost';

GRANT SELECT, INSERT, UPDATE, DELETE ON factory\_inventory\_management.orders TO '杨晨\_2402'@'localhost';

GRANT SELECT, INSERT, UPDATE, DELETE ON factory\_inventory\_management.order\_parts TO '杨晨\_2402'@'localhost';

GRANT SELECT, INSERT, UPDATE, DELETE ON factory\_inventory\_management.employees TO '杨晨\_2402'@'localhost';

GRANT SELECT, INSERT, UPDATE, DELETE ON factory\_inventory\_management.warehouse\_inventory TO '杨晨\_2402'@'localhost';

GRANT SELECT, INSERT, UPDATE, DELETE ON factory\_inventory\_management.warehouse\_employees TO '杨晨\_2402'@'localhost';

-- 刷新权限

FLUSH PRIVILEGES;

SHOW GRANTS FOR '杨晨\_2402'@'localhost';

