## Mysql练习一

#创建数据库

CREATE DATABASE lianxi01 CHARACTER SET utf8;

#使用数据库lianxi01

USE lianxi01;

#创建表

CREATE TABLE product(

pid INT PRIMARY KEY,

pname VARCHAR(20),

price DOUBLE ,

category\_name VARCHAR(32)

);

#导入数据

INSERT INTO product(pid,pname,price,category\_name) VALUES(1,'联想电脑',5000,'电脑办公');

INSERT INTO product(pid,pname,price,category\_name) VALUES(2,'海尔电脑',3000,'电脑办公');

INSERT INTO product(pid,pname,price,category\_name) VALUES(3,'雷神电脑',5000,'电脑办公');

INSERT INTO product(pid,pname,price,category\_name) VALUES(4,'JACK JONES',800,'服装');

INSERT INTO product(pid,pname,price,category\_name) VALUES(5,'真维斯',200,'服装');

INSERT INTO product(pid,pname,price,category\_name) VALUES(6,'花花公子',440,'服装');

INSERT INTO product(pid,pname,price,category\_name) VALUES(7,'劲霸',2000,'服装');

INSERT INTO product(pid,pname,price,category\_name) VALUES(8,'香奈儿',800,'女士用品');

INSERT INTO product(pid,pname,price,category\_name) VALUES(9,'相宜本草',200,'女士用品');

INSERT INTO product(pid,pname,price,category\_name) VALUES(10,'面霸',5,'女士用品');

INSERT INTO product(pid,pname,price,category\_name) VALUES(11,'雪碧',56,'饮料饮品');

INSERT INTO product(pid,pname,price,category\_name) VALUES(12,'香飘飘奶茶',1,'饮料饮品');

INSERT INTO product(pid,pname,price,category\_name) VALUES(13,'iPhone9',8000,NULL);

#1.查询所有的商品

SELECT \* FROM product;



#2.查询商品名和商品价格

SELECT

pname AS '商品名',

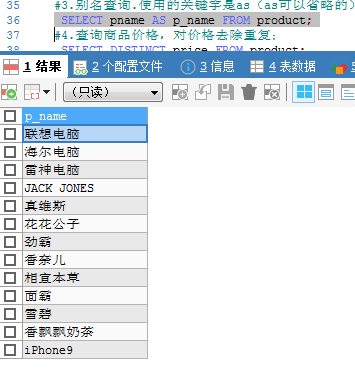
price AS '价格'

FROM product;



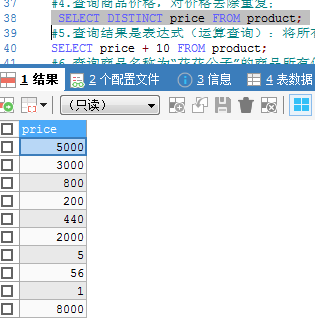
#3.别名查询.使用的关键字是as（as可以省略的）.

SELECT pname AS p\_name FROM product;



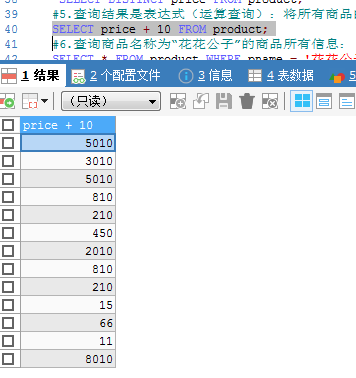
#4.查询商品价格，对价格去除重复；

SELECT DISTINCT price FROM product;



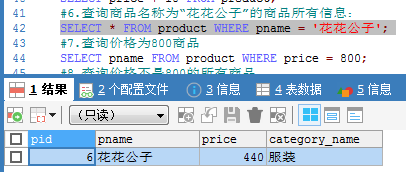
#5.查询结果是表达式（运算查询）：将所有商品的价格+10元进行显示.

SELECT price + 10 FROM product;



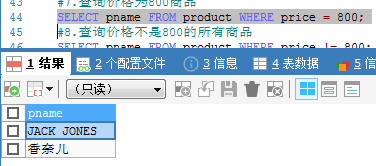
#6.查询商品名称为“花花公子”的商品所有信息：

SELECT \* FROM product WHERE pname = '花花公子';



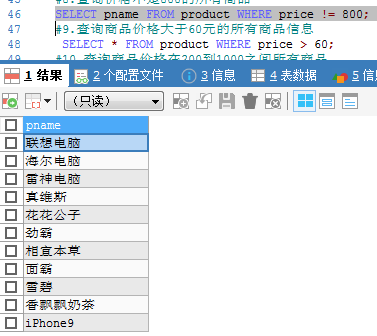
#7.查询价格为800商品

SELECT pname FROM product WHERE price = 800;



#8.查询价格不是800的所有商品

SELECT pname FROM product WHERE price != 800;



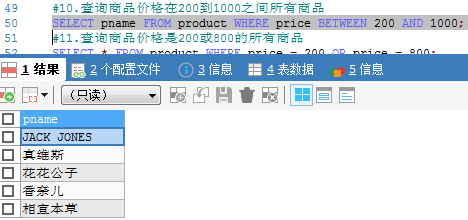
#9.查询商品价格大于60元的所有商品信息

SELECT \* FROM product WHERE price > 60;



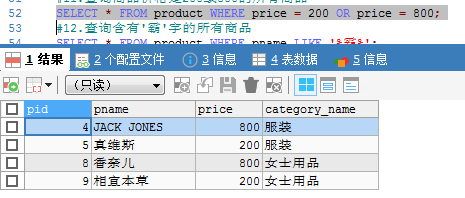
#10.查询商品价格在200到1000之间所有商品

SELECT pname FROM product WHERE price BETWEEN 200 AND 1000;



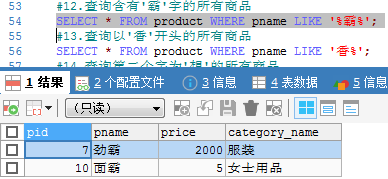
#11.查询商品价格是200或800的所有商品

SELECT \* FROM product WHERE price = 200 OR price = 800;



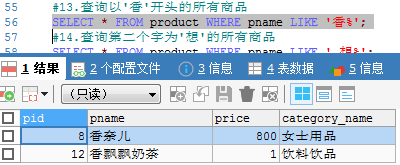
#12.查询含有'霸'字的所有商品

SELECT \* FROM product WHERE pname LIKE '%霸%';



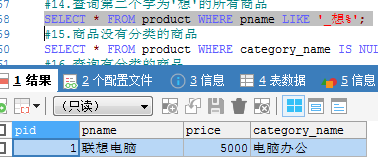
#13.查询以'香'开头的所有商品

SELECT \* FROM product WHERE pname LIKE '香%';



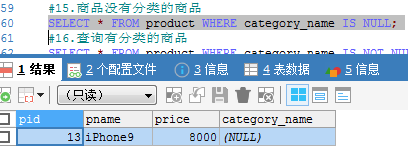
#14.查询第二个字为'想'的所有商品

SELECT \* FROM product WHERE pname LIKE '\_想%';



#15.商品没有分类的商品

SELECT \* FROM product WHERE category\_name IS NULL;



#16.查询有分类的商品

SELECT \* FROM product WHERE category\_name IS NOT NULL;



-- 排序练习：

#1.使用价格对商品信息排序(降序)

-- 提示：使用order by语句

SELECT \* FROM product ORDER BY price DESC;



#2.在价格排序(降序)的基础上，以主键排序(降序)

-- 提示：若价格相同，相同价格的数据以pid降序排序

SELECT \*FROM product ORDER BY price DESC,pid ASC



#3.显示商品的价格(去重复)，并排序(降序)

-- 提示：DISTINCT 关键字去重

SELECT DISTINCT \* FROM product ORDER BY price DESC;

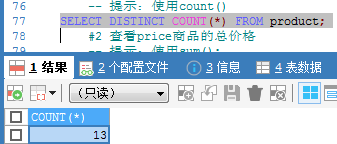


-- 聚合函数/分组函数练习：

#1 查询商品的总条数(两种方式)

-- 提示：使用count()

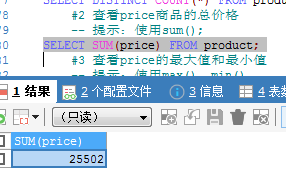
SELECT DISTINCT COUNT(\*) FROM product;



#2 查看price商品的总价格

-- 提示：使用sum();

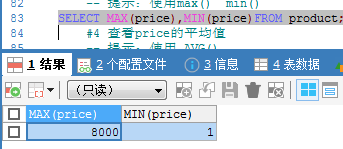
SELECT SUM(price) FROM product;



#3 查看price的最大值和最小值

-- 提示：使用max() min()

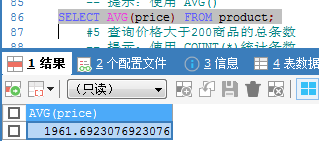
SELECT MAX(price),MIN(price)FROM product;



#4 查看price的平均值

-- 提示：使用 AVG()

SELECT AVG(price) FROM product;



#5 查询价格大于200商品的总条数

-- 提示：使用 COUNT(\*)统计条数

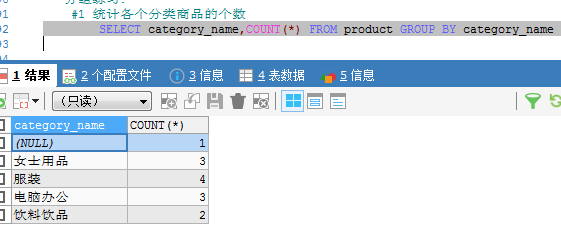
SELECT COUNT(1) FROM product WHERE price > 200;



-- 分组练习：

#1 统计各个分类商品的个数

SELECT category\_name,COUNT(\*) FROM product GROUP BY category\_name ;



## Mysql练习二

#1. 创建部门表(id,name)

# 创建部门表

CREATE TABLE dept(

id INT PRIMARY KEY AUTO\_INCREMENT,

NAME VARCHAR(20)

);

INSERT INTO dept (NAME) VALUES ('开发部'),('市场部'),('财务部');

#2. 员工表，员工表(id, 姓名，性别，工资，入职日期，部门ID)

# 创建员工表

CREATE TABLE employee (

id INT PRIMARY KEY AUTO\_INCREMENT,

NAME VARCHAR(10),

gender CHAR(1), -- 性别

salary DOUBLE, -- 工资

join\_date DATE, -- 入职日期

dept\_id INT,

FOREIGN KEY (dept\_id) REFERENCES dept(id) -- 外键，关联部门表(部门表的主键)

);

INSERT INTO employee(NAME,gender,salary,join\_date,dept\_id) VALUES('孙悟空','男',7200,'2013-02-24',1);

INSERT INTO employee(NAME,gender,salary,join\_date,dept\_id) VALUES('猪八戒','男',3600,'2010-12-02',2);

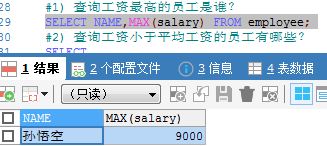
INSERT INTO employee(NAME,gender,salary,join\_date,dept\_id) VALUES('唐僧','男',9000,'2008-08-08',2);

INSERT INTO employee(NAME,gender,salary,join\_date,dept\_id) VALUES('白骨精','女',5000,'2015-10-07',3);

INSERT INTO employee(NAME,gender,salary,join\_date,dept\_id) VALUES('蜘蛛精','女',4500,'2011-03-14',1);

#1) 查询工资最高的员工是谁？

SELECT NAME,MAX(salary) FROM employee;



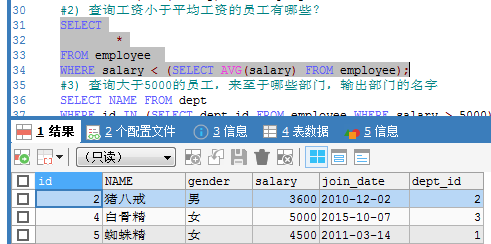
#2) 查询工资小于平均工资的员工有哪些？

SELECT

\*

FROM employee

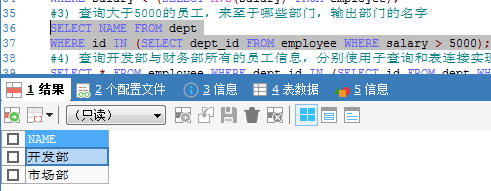
WHERE salary < (SELECT AVG(salary) FROM employee);



#3) 查询大于5000的员工，来至于哪些部门，输出部门的名字

SELECT NAME FROM dept

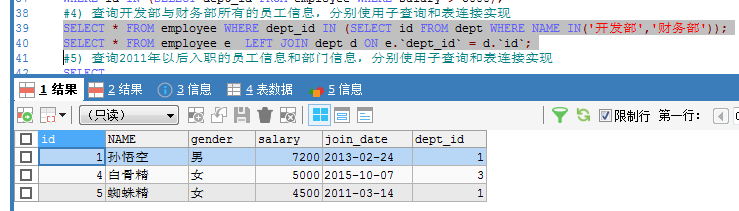
WHERE id IN (SELECT dept\_id FROM employee WHERE salary > 5000);



#4) 查询开发部与财务部所有的员工信息，分别使用子查询和表连接实现

SELECT \* FROM employee WHERE dept\_id IN (SELECT id FROM dept WHERE NAME IN('开发部','财务部'));

SELECT \* FROM employee e LEFT JOIN dept d ON e.`dept\_id` = d.`id`;



#5) 查询2011年以后入职的员工信息和部门信息，分别使用子查询和表连接实现

SELECT

\*

FROM employee e,dept d WHERE e.`join\_date` > '2011-12-31' AND e.`dept\_id` = d.`id`;



## Mysql练习三

/\*第一题

-- 1、查询平均成绩大于70分的同学的学号,姓名,和平均成绩

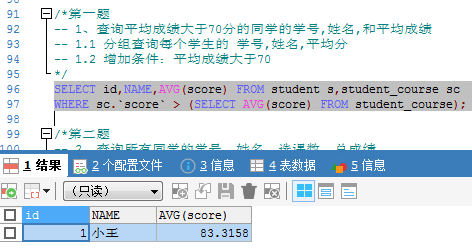
-- 1.1 分组查询每个学生的 学号,姓名,平均分

-- 1.2 增加条件：平均成绩大于70

\*/

SELECT id,NAME,AVG(score) FROM student s,student\_course sc

WHERE sc.`score` > (SELECT AVG(score) FROM student\_course);



/\*第二题

-- 2. 查询所有同学的学号、姓名、选课数、总成绩

-- 2.1 需要查询两张表 student表和 student\_course表

-- 2.2 需要使用 student\_id 学号字段,进行分组

-- 2.3 需要使用到 count函数 sum函数

\*/

SELECT DISTINCT

s.`id`,

s.`NAME`,

COUNT(sc.`course\_id`),

SUM(sc.`score`)

FROM student s,student\_course sc WHERE s.`id` = sc.`student\_id` GROUP BY s.`NAME`,s.`id`;



/\*第三题

-- 3. 查询学过赵云老师课程的同学的学号、姓名

-- 3.1 查询赵云老师的id

-- 3.2 根据老师ID,在课程表中查询所教的课程编号

-- 3.3 将上面的子查询作为 where 后面的条件

\*/

SELECT

t.`id`

FROM teacher t WHERE t.`NAME` = '赵云';

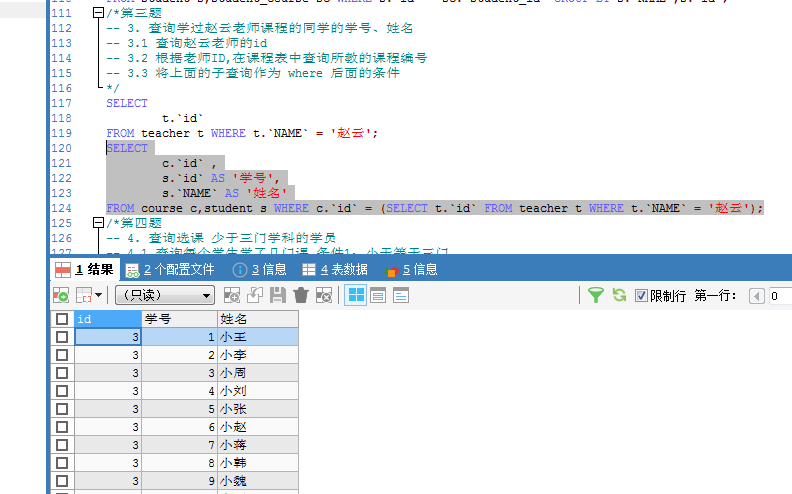
SELECT

c.`id` ,

s.`id` AS '学号',

s.`NAME` AS '姓名'

FROM course c,student s WHERE c.`id` = (SELECT t.`id` FROM teacher t WHERE t.`NAME` = '赵云');



/\*第四题

-- 4. 查询选课 少于三门学科的学员

-- 4.1 查询每个学生学了几门课 条件1：小于等于三门

-- 4.2 查询 学号和姓名， 将4.1 作为临时表

\*/

SELECT

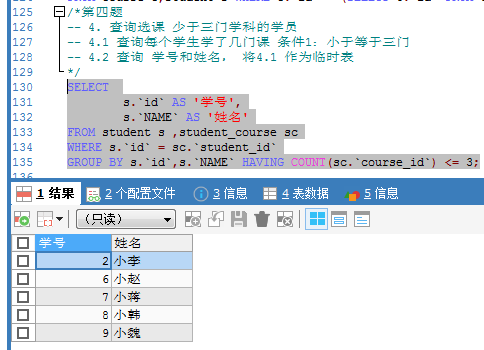
s.`id` AS '学号',

s.`NAME` AS '姓名'

FROM student s ,student\_course sc

WHERE s.`id` = sc.`student\_id`

GROUP BY s.`id`,s.`NAME` HAVING COUNT(sc.`course\_id`) <= 3;



## Mysql练习四

CREATE DATABASE lianxi02 DEFAULT CHARACTER SET utf8;

/\*讲师表

讲师ID 主键 int类型

讲师姓名 VARCHAR类型

讲师简介 VARCHAR类型

讲师级别 char类型 高级讲师&首席讲师

为讲师姓名添加索引

\*/

CREATE TABLE teacher(

id INT PRIMARY KEY AUTO\_INCREMENT,

NAME VARCHAR(20),

intro VARCHAR(50),

LEVEL CHAR(20)

);

ALTER TABLE teacher DEFAULT CHARACTER SET utf8;

ALTER TABLE teacher ADD INDEX index\_teacher\_name(NAME);

INSERT INTO teacher(id,NAME,intro,LEVEL) VALUES (1, '刘德华', '毕业于清华大学，主攻前端技术,授课风格生动活泼,深受学员喜爱', '高级讲师');

INSERT INTO teacher(id,NAME,intro,LEVEL) VALUES (2, '郭富城', '毕业于北京大学，多年的IT经验，研发多项Java课题,授课经验丰富', '首席讲师');

/\*

课程分类表

课程分类ID 主键 int类型

课程分类名称 VARCHAR类型 比如前端开发 后端开发 数据库DBA......

课程分类描述 VARCHAR类型

创建时间 datetime类型

更新时间 datetime类型

\*/

CREATE TABLE lagou\_subject(

subject\_id INT PRIMARY KEY AUTO\_INCREMENT,

subject\_name VARCHAR(50),

subject\_descp VARCHAR(50),

create\_time DATETIME,

update\_time DATETIME

);

INSERT INTO lagou\_subject VALUES (1, '后端开发', '后端课程包括 Java PHP Python', '2020-03-27 00:44:04', '2020-03-27 00:44:04');

INSERT INTO lagou\_subject VALUES (2, '前端开发', '前端课程包括 JQuery VUE angularJS', '2020-02-27 10:00:04', '2020-02-27 18:44:04');

/\*课程表

课程ID 主键 int类型

课程讲师ID 外键 用于描述课程的授课老师

课程分类ID 外键 用于描述课程所属的分类 比如 Java课程就属于后端分类

课程标题 VARCHAR类型 比如Java VUE PHP ......

总课时 int类型

浏览数量 bigint类型

课程状态 char 类型, 0 未发布(默认) 1 已发布

为 课程标题字段添加索引

为 teacher\_id & subject\_id,添加外键约束

\*/

CREATE TABLE lagou\_course(

course\_id INT PRIMARY KEY AUTO\_INCREMENT,

teacher\_id INT,

subject\_id INT,

course\_title VARCHAR(50),

all\_class INT,

browse\_num BIGINT,

course\_sta CHAR(2)

);

ALTER TABLE lagou\_course ADD fk\_course\_teacher FOREIGN KEY(teacher\_id) REFERENCES teacher(id);

ALTER TABLE lagou\_course ADD fk\_course\_subject FOREIGN KEY(subject\_id) REFERENCES SUBJECT(subject\_id);

INSERT INTO lagou\_course VALUES (1,1,1 ,'Java', 300,250000, '1');

INSERT INTO lagou\_course VALUES (2,2,2, 'VUE', 400,200000,'1');

#查询刘德华老师所教的课程属于哪个课程分类

SELECT

\*

FROM lagou\_course

WHERE(SELECT NAME FROM teacher WHERE NAME = '刘德华');

