**每日作业卷答案**

**MySQL多表查询**

传智播客.黑马程序员

# 关卡1

## 训练案例：子查询的练习

### 训练描述

有两张表，一张员工表，一张部门表。建表和数据的代码如下：

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);

按要求实现下面的查询功能。

### 操作步骤描述

1. 查询工资最高的员工是谁？
2. 查询工资小于平均工资的员工有哪些？
3. 查询大于5000的员工，来至于哪些部门，输出部门的名字
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
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

### 答案：

1) 查询工资最高的员工是谁？

select \* from employee where salary = (select max(salary) from employee);

2) 查询工资小于平均工资的员工有哪些？

2.1) 查询平均工资

select avg(salary) from employee;

2.2) 小于平均工资的员工

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

3.1) 查询大于5000的员工，来至于哪些部门的id

select dept\_id from employee where salary >5000;

3.2) 外查询 Subquery returns more than 1 row

select d.\* from dept d where d.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

4.1) 子查询

select \* from employee where dept\_id in (select d.id from dept d where d.name='开发部' or d.name='财务部');

4.2) 表连接

select e.\* from employee e inner join dept d on e.dept\_id = d.id where d.name='开发部' or d.name='财务部';

5) 查询2011年以后入职的员工信息和部门信息

● 使用子查询：

5.1) 查询出2011年以后入职的员工信息

select \* from employee where join\_date >='2011-1-1';

5.2) 查询所有的部门信息，与上面的虚拟表中的信息比对，找出所有部门ID相等的员工。

select \* from dept d ,(select \* from employee where join\_date >='2011-1-1') e where d.id = e.dept\_id;

● 使用表连接：

select d.\*, e.\* from employee e inner join dept d on e.dept\_id = d.id where e.join\_date >= '2011-1-1';

# 关卡2

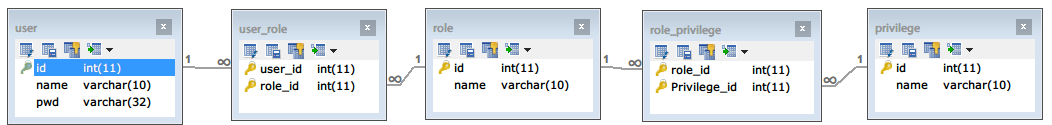
## 训练案例：权限系统设计五张表

### 训练描述

某管理系统需要开发一套权限管理系统。众所周知，权限系统是每个系统里面必备的最基本的系统。每个用户有不同的角色，每个角色有不同的权限。请设计一套权限系统，使用多表连接查询，通过不同的用户名得到这个用户有哪些权限。

### 操作步骤描述

* 设计的思想：

首先由用户、角色、权限三大基础表组成，加上用户-角色、角色-权限两个中间表。这样就可以通过登录的用户来获取权限列表，或判断用户是否拥有某个权限。五张表的关系如下：

* 各表的字段：

用到５张表：用户表（user）、角色表（role）、权限表（privilege）、用户角色表（user\_role）、角色权限表（role\_privilege）。各表的大体表结构如下：

1. 用户表（user）：id、name、pwd
2. 角色表（role）：id、name
3. 权限表（privilege）：id、name
4. 用户角色表（user\_role）：user\_id、role\_id

角色权限表（role\_privilege）：role\_id、privilege\_id

* 要求：

1) 创建5张表，添加三个用户，三种角色(1总经理，2经理，3员工)

2) 添加三种权限：员工：1请假；经理：1请假、2审批请假；总经理：1请假、2审批请假、3辞退员工

3) 用户分配角色：1号员工是3号角色；2号员工是2、3号角色；3号员工是1，2号角色

4) 权限与角色的关系，1号角色有3种权限，2号角色有2种权限，3号角色有1种权限

5) 查询Jack、Rose、NewBoy登录以后有哪些权限，查询后要去掉重复的项。

### 答案：

1)、用户信息表：

create table user (

id int primary key auto\_increment,

name varchar(10),

pwd varchar(32)

)

# 添加三个员工，Jack是员工，Rose是经理，NewBoy是总经理

insert into user (name,pwd) values ('Jack', '111');

insert into user (name,pwd) values ('Rose', '222');

insert into user (name,pwd) values ('NewBoy', '333');

2)、角色表：

create table role (

id int primary key auto\_increment,

name varchar(10)

)

# 添加三种角色

insert into role(name) values('总经理'),('经理'),('员工');

3)、权限表

create table Privilege (

id int primary key auto\_increment,

name varchar(10)

)

# 添加三种权限

# 员工：请假

# 部门经理：请假，审批请假

# 总经理：请假，审批请假，辞退员工

insert into Privilege (name) values ('请假'),('审批请假'),('辞退员工');

4)、用户与角色关系表

create table user\_role (

user\_id int ,

role\_id int,

primary key(user\_id, role\_id), -- 联合主键

foreign key (user\_id) references user(id),

foreign key (role\_id) references role(id)

)

# 用户分配角色：1号员工是3号角色，2号员工是2，3号角色，3号员工是1，2号角色

insert into user\_role values(1,3),(2,3),(2,2),(3,1),(3,2);

5)、角色与权限关系表

create table role\_Privilege (

role\_id int ,

Privilege\_id int,

primary key(role\_id, Privilege\_id), -- 联合主键

foreign key (role\_id) references role(id),

foreign key (Privilege\_id) references Privilege(id)

)

#权限与角色的关系，1号角色有3种权限，2号角色有2种权限，3号角色有1种权限

insert into role\_Privilege values (1,1),(1,2),(1,3),(2,1),(2,2),(3,1);

6)、查询指定用户有哪些权限

# 分别查询Jack、Rose、NewBoy登录以后有哪些权限

# 去掉重复的

SELECT

distinct(Privilege.name) 权限

FROM role\_Privilege

INNER JOIN role

INNER JOIN Privilege

INNER JOIN user\_role

INNER JOIN user

ON (user\_role.user\_id = user.id)

AND (user\_role.role\_id = role.id)

AND (role\_Privilege.role\_id = role.id)

AND (role\_Privilege.Privilege\_id = Privilege.id)

where user.name = 'NewBoy';

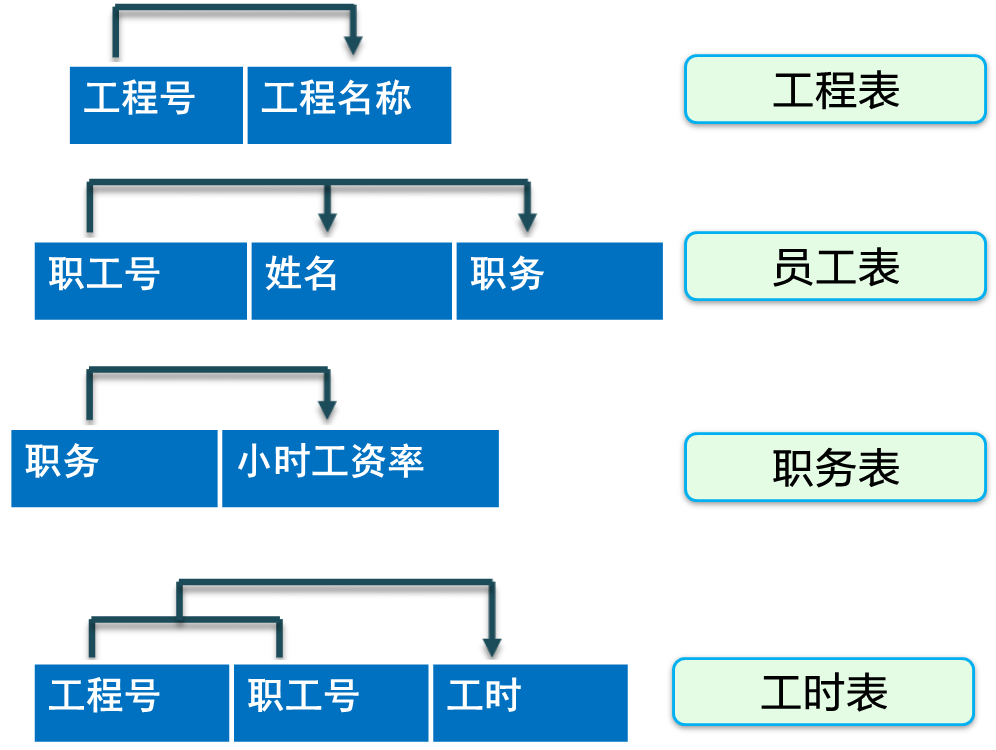
# 关卡3

## 训练案例：工程建筑公司的数据管理

### 训练描述

假设某建筑公司要设计一个数据库。公司的业务规则概括说明如下：

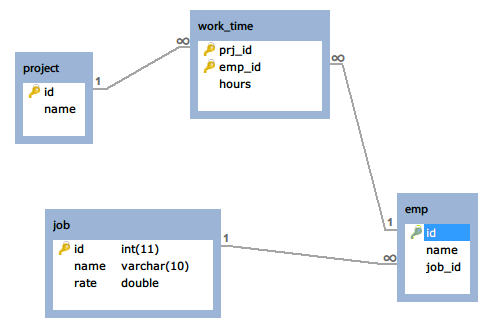
* 公司承担多个工程项目，每一项工程有：工程号、工程名称、施工人员等
* 公司有多名职工，每一名职工有：职工号、姓名、职务
* 公司按照工时和小时工资率支付工资，小时工资率由职工的职务决定（例如，技术员的小时工资率与工程师不同）

设计出四张表，如下图所示：

* 知识点：

1. 使用DDL语句创建表结构
2. 使用DML插入表的记录
3. 使用DQL进行多表连接查询

* 几张表的关系如下：



* 查询要求：

1. 查询职工号是1001的职工职务信息，要求输出职工号，姓名及其职务，并使用别名(内连接)
2. 查询职务为“技术员”的职工薪水，要求输出姓名，职务，工时率(内连接)
3. 查询所有的工程号，工程名称，职工号，工时(右连接)
4. 查询“班建斌”职工的工作情况，要求输出姓名，参与的工程名称，工时(3表连接)
5. 使用四表连接查询，查询出如下图的结果

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 工程号 | 工程名称 | 职工号 | 姓名 | 职务 | 小时工资率 | 工时 |
| A1 | 花园大厦 | 1001 | 杨国明 | 工程师 | 65 | 13 |
| A1 | 花园大厦 | 1002 | 班建斌 | 技术员 | 60 | 16 |
| A1 | 花园大厦 | 1004 | 伍岳林 | 律师 | 100 | 19 |
| A2 | 立交桥 | 1001 | 杨国明 | 工程师 | 65 | 13 |
| A2 | 立交桥 | 1003 | 鞠明亮 | 工人 | 55 | 17 |
| A3 | 临江饭店 | 1002 | 班建斌 | 技术员 | 60 | 18 |
| A3 | 临江饭店 | 1004 | 伍岳林 | 律师 | 100 | 14 |

### 操作步骤描述

1. 创建工程表： 工程号(主键,字符串),工程名称解答问题的部门，使用p标签
2. 职务表: 职务编号(主键)，职务,小时工资率
3. 员工表: 职工号(主键，从1001开始),姓名,职务编号(外键)
4. 工时表: 工时编号(主键)，工程号(外键),职工号(外键),工时
5. 插入工程数据、职务表、员工表、工时表

### 答案：

-- 1) 创建工程表： 工程号(主键,字符串),工程名称

create table project (

id char(2) primary key,

name varchar(20)

)

-- 2) 职务表: 职务编号(主键)，职务,小时工资率

create table job (

id int primary key,

name varchar(10),

rate double

)

-- 3) 员工表: 职工号(主键，从1001开始),姓名,职务编号(外键)

create table emp (

id int primary key auto\_increment,

name varchar(20),

job\_id int,

foreign key (job\_id) references job(id)

)

alter table emp auto\_increment = 1001;

-- 4) 工时表: 工程号(外键),职工号(外键),工时

create table work\_time (

prj\_id char(2),

emp\_id int,

hours int,

primary key(prj\_id, emp\_id),

foreign key(prj\_id) references project(id),

foreign key(emp\_id) references emp(id)

)

-- 插入工程数据

insert into project values ('A1','花园大厦'),('A2','立交桥'),('A3','临江饭店');

select \* from project;

-- 插入职务表

insert into job values (1,'工程师',65),(2,'技术员',60),(3,'律师',100),(4,'工人',55);

select \* from job;

-- 插入员工表

insert into emp (name,job\_id) values ('杨国明',1),('班建斌',2),('伍岳林',3),('鞠明亮',4);

select \* from emp;

-- 插入工时表

insert into work\_time values ('A1',1001,13),('A1',1002,16),('A1',1003,19),

('A2',1001,13),('A2',1004,17),('A3',1002,18),('A3',1003,14);

select \* from work\_time;

-- 查询职工号是1001的职工职务信息，要求输出职工号，姓名及其职务，并使用别名

select e.emp\_id as 职工号,e.name as 员工名,j.name as 职务 from emp e inner join job j on e.job\_id = j.job\_id where e.emp\_id=1001;

-- 查询职务为“技术员”的职工薪水，要求输出姓名，职务，工时率

select e.name as 员工名, j.name as 职务, j.rate as 工时率 from emp e inner join job j on e.job\_id = j.job\_id where j.name = '技术员'

-- 查询所有的工程号，工程名称，职工号，工时

select p.id as 工程号,p.name as 工程名称, w.emp\_id as 职工号, w.hours 工时 from project p right join work\_time w on p.id = w.id;

-- 查询“班建斌”职工的工作情况，要求输出姓名，参与的工程名称，工时

select e.name, p.name,w.hours from emp e, project p ,work\_time w where w.id=p.id

and w.emp\_id=e.emp\_id and e.name = '班建斌';

或

select e.name, p.name,w.hours from emp e inner join work\_time w inner join project p on e.emp\_id = w.emp\_id and w.id=p.id where e.name = '班建斌';

-- 查询所有的信息(工程表，员工表，职务表，工时表)

select p.id 工程号, p.name 工程名称,e.id 职工号,e.name 姓名,j.name 职务,j.rate 小时工资率,w.hours 工时 from project p inner join emp e inner join job j inner join work\_time w on p.id = w.prj\_id and w.emp\_id = e.id and j.id = e.job\_id order by p.id;