(北京)

## 中国石油大学

# CHINA UNIVERSITY OF PETROLEUM

### 数据库原理上机报告

实验二 数据库的查询和视图

完成日期 2021 年 10 月 16 日

### 《数据库原理》上机报告

报告名称	数据库的查询和视图			₩ F III F
姓名	刘康来	班级	计算机 19-3	粘贴
成绩				照片

1. 上机实践目的与准备知识(简介,300字以内)

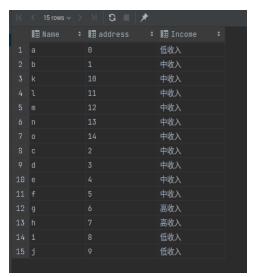
#### 1.实验目的

- (1) 掌握子查询的表示;
- (2) 掌握连接查询的表示;
- (3) 掌握 GROUP BY、ORDER BY 子句的作用和使用方法;
- (4) 掌握创建视图、查询和修改视图。

#### 2. 实验准备:

- (1) 了解子查询的表示方法; from(); where name in(), not in(), > all()
- (2) 了解连接查询的表示方法; inner join, left outer join, right outer join, full outer join + natural, on(P), using(A...)
  - (3) 了解 SELECT 语句 GROUP BY、ORDER BY 子句的作用和使用方法;
- (4) 了解创建视图和操作的方法。create view name as()
- 2. 主要实践内容与具体操作步骤(实践内容完成情况要有描述,如执行的 SQL 命令等,有运行结果截图,图大小以保证文字清晰为准)
- 1. 查询员工的姓名、住址和收入水平,3000 以下显示低收入、3000—7000 显示中等收入、7000 以上显示高收入;

```
select Name, address,
case
when Income < 3000 then '低收入'
when Income > 7000 then '高收入'
else '中收入'
end as Income
from Employees natural join Salary;
```



(2) 查询财务部年龄不低于研发部雇员年龄的雇员姓名; 改为研发部,销售部

```
select Name
from Employees natural join Departments
where DepartmentName = '研发部' and Birthday ≤ all(
    select Birthday
    from Employees natural join Departments
    where DepartmentName = '销售部'
);
```



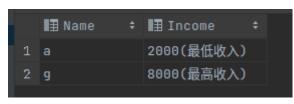
(3) 查询财务部收入在5000元以上的雇员姓名及其薪水情况;改为研发部

select Name, Income from (Employees natural join Salary) natural join Departments where DepartmentName = '研发部' and Income > 5000;



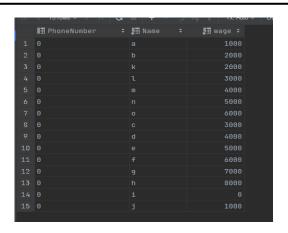
(4) 查询财务部雇员的最高和最低收入; 改为研发部

```
with FinanceIncome as (
   from (Employees natural join Salary) natural join Departments
   where DepartmentName = '研发部'
select Name,
       select max(Income)
       from FinanceIncome
   ) then concat(cast(Income as varchar(10)), '(最高收入)')
       select min(Income)
       from FinanceIncome
   ) then concat(cast(Income as varchar(10)), '(最低收入)')
   else Income
end as Income
from FinanceIncome
where Income ≥ all (
       select Income
       from FinanceIncome
   Income ≤ all (
       select Income
       from FinanceIncome
```



(5) 创建视图 Employees\_view,包含员工号码、姓名、实际收入三项,创建视图 DS\_VIEW,包含 Departments 的全部列;

```
create view Employees_view as
    select PhoneNumber, Name, (Income - Outcome) as wage
    from Employees natural join Salary;
select *
from Employees_view;
```

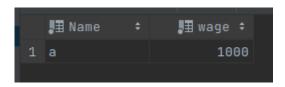


```
create view DS_VIEW as
    select *
    from Departments;
select *
from DS_VIEW;
```



(6) 从视图 Employees\_view 中查询姓名为'王琳'(也可是其他人)的员工的实际收入;

```
select Name, wage
from Employees_view
where Name = 'a';
```



(7) 可否修改视图 Employees\_view,为什么?

#### Modify the Employees\_view?

- Can't, wage 属性是表达式!
- 1. The from clause has only one database relation
- 2. The select clause contains only attribute names of the relation and does not have any

expressions, aggregates, or distinct specification.

(wage is a expression!)

3. Any attribute not listed in the select clause and be set to null; that is, it does not

have a not null constraint and is not part of a primary key.

4. The query does not have a group by or having clause.

#### 3. 总结与问题分析(100字以上)

- select 不能子查询,Scalar subquery 一般用于分组得一值;选择方法: case, if, coalesce(judge null)
- 作用域的问题?

- where 语句不能有聚合函数,而 having 用于 group by,只能 >= all (subquery)?
- 最多只能两个 with 语句?

下面语句是错的,不知原因?聚合函数的问题?

```
case
when Income = max(Income) then concat(cast(max(Income) as
varchar(10)), '(最高收入)')
when Income = min(Income) then concat(cast(min(Income) as
varchar(10)), '(最低收入)')
end as income
```

学习有大量的细节,但其实也没那么重要;而且不同的平台各有各的方案,标准难以统一。万变不离其宗,把握内在的脉络,深入了解原理,才能克敌制胜。PS:关于视图的修改,有很多的标准,但根本是要好能契合原关系的修改,当然还有很多需要注意的地方,比如插入的内容需符合 where 条件,要知道任何的规范都是为了更好的服务,按照正常的方法去思考,一定会有答案。