基于openGauss的SQL查询练习

## 登录openGauss数据库服务

登录华为云服务

<https://www.huaweicloud.com/>。

选择控制台->弹性云服务器ECS。

将自己的ECS服务器开机，并通过远程登陆登入云服务器。用户名为root，密码为自设密码（若忘记密码可在ECS开机、远程登陆界面重新设置）。

使用root登录进入系统后，使用“ **su - omm**”登录omm用户。之后使用命令“ **gs\_om -t start;**”启动数据库服务。

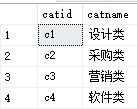
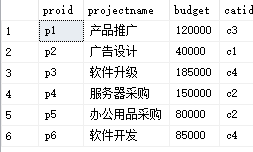
使用命令“gsql -d postgres -p 26000 -r”连接数据库。

## SQL查询练习

### 数据库模式：

Department Employee

Category Project Workson

数据库模式如下（分别为Department，Employee，Category，Project，Workson）：

部门（部门号，部门名称，位置）

员工（员工号，姓名，年龄，性别，所在部门号）

项目种类（项目种类号，项目种类名）

项目（项目号，项目名称，预算，项目种类号）

员工工作情况（员工号，项目号，职责，开始日期）

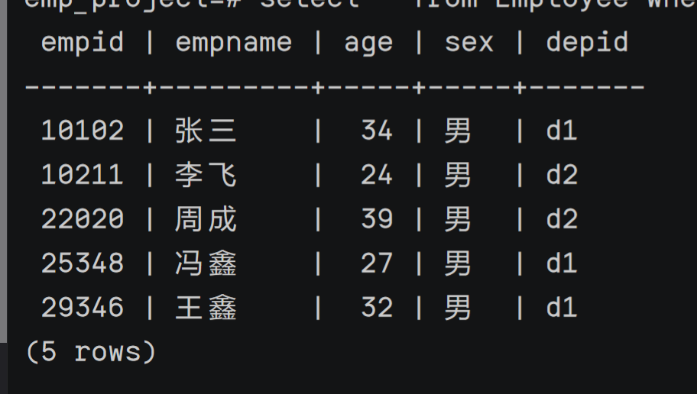
注：上面数据仅供参考，具体的SQL语句不应该和具体的数据有关。

### 创建数据库、表、插入数据

|  |
| --- |
| DROP DATABASE IF EXISTS emp\_project;  CREATE DATABASE emp\_project；  \connect emp\_project;  DROP TABLE IF EXISTS category;  CREATE TABLE category(  catid varchar(16) NOT NULL,  catname varchar(32) NOT NULL,  PRIMARY KEY (catid)  );  INSERT INTO category VALUES ('c1', '设计类');  INSERT INTO category VALUES ('c2', '财务部');  INSERT INTO category VALUES ('c3', '营销类');  INSERT INTO category VALUES ('c4', '软件类');  INSERT INTO category VALUES ('c5', '运营部');  DROP TABLE IF EXISTS department;  CREATE TABLE department (  depid varchar(16) NOT NULL ,  depname varchar(128) NOT NULL,  location varchar(128) NOT NULL,  PRIMARY KEY (depid)  ) ;  INSERT INTO department VALUES ('d1', '开发部', '天津');  INSERT INTO department VALUES ('d2', '财务部', '北京');  INSERT INTO department VALUES ('d3', '市场部', '广州');  INSERT INTO department VALUES ('d4', '人才管理部', '天津');  DROP TABLE IF EXISTS employee;  CREATE TABLE employee (  empid varchar(16) NOT NULL,  empname varchar(32) NOT NULL,  age int NOT NULL,  sex varchar(16) NOT NULL,  depid varchar(16) NOT NULL,  PRIMARY KEY (empid),  FOREIGN KEY (depid) REFERENCES department (depid) ON DELETE RESTRICT ON UPDATE RESTRICT  );  INSERT INTO employee VALUES ('10102', '张三', 34, '男', 'd1');  INSERT INTO employee VALUES ('10211', '李飞', 24, '男', 'd2');  INSERT INTO employee VALUES ('17114', '张伟', 36, '女', 'd1');  INSERT INTO employee VALUES ('18316', '王玲', 29, '女', 'd4');  INSERT INTO employee VALUES ('22020', '周成', 39, '男', 'd2');  INSERT INTO employee VALUES ('25348', '冯鑫', 27, '男', 'd1');  INSERT INTO employee VALUES ('28559', '李凤', 41, '女', 'd3');  INSERT INTO employee VALUES ('29346', '王鑫', 32, '男', 'd1');  DROP TABLE IF EXISTS project;  CREATE TABLE project (  proid varchar(16) NOT NULL,  projectname varchar(32) NOT NULL,  budget int NOT NULL,  catid varchar(16) NOT NULL,  PRIMARY KEY (proid) ,  FOREIGN KEY (catid) REFERENCES category (catid) ON DELETE RESTRICT ON UPDATE RESTRICT  );  INSERT INTO project VALUES ('p1', '产品推广', 120000, 'c3');  INSERT INTO project VALUES ('p2', '广告设计', 40000, 'c1');  INSERT INTO project VALUES ('p3', '软件升级', 185000, 'c4');  INSERT INTO project VALUES ('p4', '服务器采购', 150000, 'c2');  INSERT INTO project VALUES ('p5', '办公用品采购', 80000, 'c2');  INSERT INTO project VALUES ('p6', '软件开发', 85000, 'c4');  INSERT INTO project VALUES ('p7', '软件维护', 130000, 'c2');  INSERT INTO project VALUES ('p8', '产品售后', 56000, 'c5');  DROP TABLE IF EXISTS workson;  CREATE TABLE workson (  empid varchar(16) NOT NULL ,  proid varchar(16) NOT NULL,  job varchar(32) NULL DEFAULT NULL,  enterdate timestamp NULL DEFAULT NULL,  PRIMARY KEY (empid, proid) ,  FOREIGN KEY (empid) REFERENCES employee(empid) ON DELETE RESTRICT ON UPDATE RESTRICT,  FOREIGN KEY (proid) REFERENCES project (proid) ON DELETE RESTRICT ON UPDATE RESTRICT  );  INSERT INTO workson VALUES ('10102', 'p1', '职员', '2020-12-21 00:00:00');  INSERT INTO workson VALUES ('10102', 'p2', '职员', '2020-11-27 00:00:00');  INSERT INTO workson VALUES ('10102', 'p3', '管理员', '2020-08-05 00:00:00');  INSERT INTO workson VALUES ('10102', 'p4', '管理员', '2021-05-18 22:09:01');  INSERT INTO workson VALUES ('10102', 'p5', '管理员', '2020-12-10 00:00:00');  INSERT INTO workson VALUES ('10102', 'p6', '职员', '2020-12-22 00:00:00');  INSERT INTO workson VALUES ('10102', 'p7', NULL, '2020-12-25 00:00:00');  INSERT INTO workson VALUES ('10102', 'p8', NULL, '2020-12-01 00:00:00');  INSERT INTO workson VALUES ('10211', 'p1', '分析员', '2021-05-18 22:09:14');  INSERT INTO workson VALUES ('10211', 'p6', '分析员', '2020-06-27 00:00:00');  INSERT INTO workson VALUES ('17114', 'p4', '职员', '2020-09-01 00:00:00');  INSERT INTO workson VALUES ('18316', 'p1', '职员', '2020-06-30 00:00:00');  INSERT INTO workson VALUES ('18316', 'p4', '职员', '2020-09-01 00:00:00');  INSERT INTO workson VALUES ('18316', 'p7', NULL, '2021-05-19 10:24:26');  INSERT INTO workson VALUES ('22020', 'p2', '管理员', '2021-05-18 22:24:17');  INSERT INTO workson VALUES ('22020', 'p8', '管理员', '2020-12-01 00:00:00');  INSERT INTO workson VALUES ('25348', 'p1', NULL, '2020-10-25 00:00:00');  INSERT INTO workson VALUES ('25348', 'p2', '分析员', '2020-08-06 00:00:00');  INSERT INTO workson VALUES ('25348', 'p4', '职员', '2021-05-18 22:24:22');  INSERT INTO workson VALUES ('28559', 'p1', '职员', '2020-06-12 00:00:00');  INSERT INTO workson VALUES ('28559', 'p3', '分析员', '2021-01-01 00:00:00');  INSERT INTO workson VALUES ('28559', 'p4', '分析员', '2021-05-18 22:24:31');  INSERT INTO workson VALUES ('29346', 'p1', '分析员', '2021-05-18 22:24:26'); |

### 基于emp\_project，请完成以下SQL查询（提供参考查询结果以供核对）：

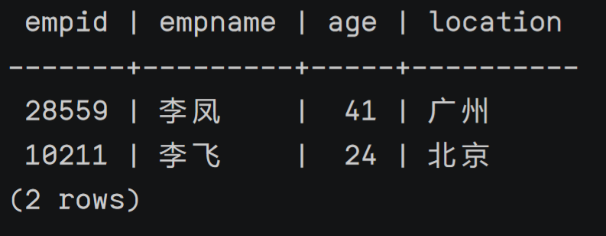
1. 给出职工中所有男性的所有信息（empid,empname,age,sex,edpid）



select\*from employee where sex = ‘男’;

select \* from employee where sex = ‘男’;

2. 统计“李”性职工信息，按年龄降序排序。（empid,empname,age,location）



select empid ,empname,age,location

from employee,department

where empname =(错)like ‘李%’ and employee.depid =department.depid

order by age desc;

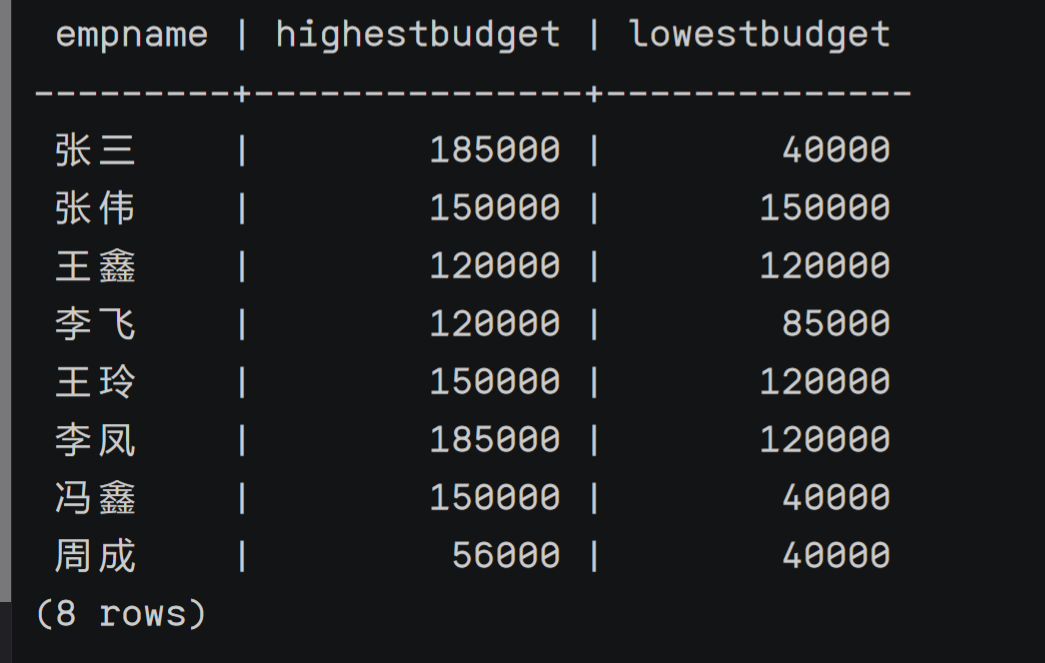
select empid,empname, age ,location

from department, employee

where employee.depid = department.depid and empname like’李%’

order by age desc;

3. 给出每位职员参与项目的最高预算和最低预算（empname，highestbudget，lowestbudget）



select empname, max(budget) as ‘highestbudget’,max(budget) as ‘lowestbudget’

from employee natural join in project,natural join in workson

where employee.empid = workson.empid and workson.proid = project.proid;

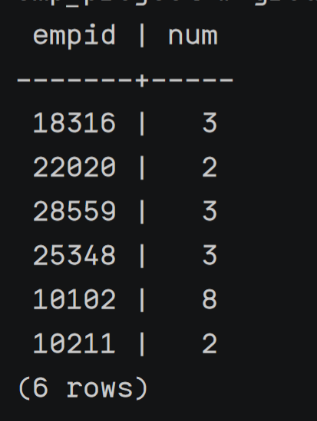
select empname,max(budget)highestbudget,min(budget)lowestbudget

from employee,project,workson

where workson.empid = employee.empid and workson.proid = project.proid

group by empname;

4. 给出所有项目超过一个的员工的id和参加的项目个数（empid, num）



select empid,count(proid)

from workson;

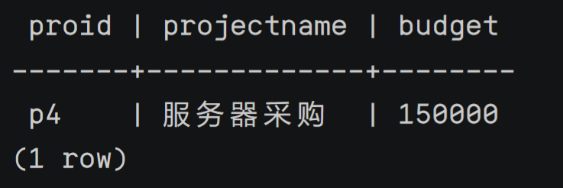
select empid ,count(empid)as num

from project,workson

where workson.proid = project.proid

group by empid having count(empid) > 1;

5. 给出项目种类号为“c2”且预算最多的项目。（proid，projectname，budget）



select proid, projectname,budget

from project

where catid = ‘c2’ and budget >= all(select budget from project);

select proid, projectname, budget

from project

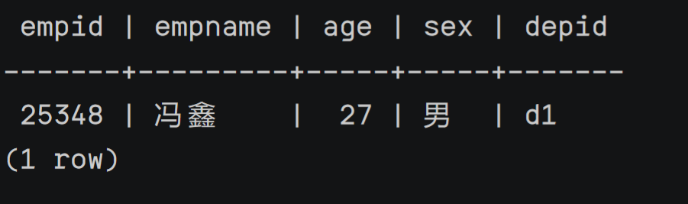
where budget =(

select budget from project

where catid = ‘c2’ oder by budget desc

limit 1);

6. 给出参加“产品推广”项目，但不担任职位的员工的信息，（empid,empname，age,sex.depid）



select employee, empname, age, sex, depid

from employee natural join workson, natrual join project

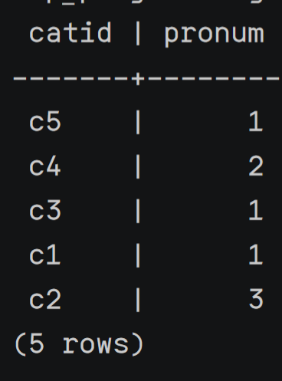
where project.proid = ‘产品推广’ and project.proid = workson.proid and workson.empid = employee.empid and workson.job <> ‘职员’;

select employee.empid,empname,age,sex,depid

from project,employee,workson

where workson.proid = project.proid and projectname = ‘产品推广’ and employee.empid = workson.empid and job is null;

7. 给出工号为“10102”的员工每类项目的参加总数，若没有参加过某类项目，则参加项目总数显示为0（catid，proNum）



select catid ,count(catid) as pronum

from project natural join workson

where workson.empid = ‘10102’ and workson.proid = project.proid;

select category.catid,count(workson.proid)as proNum

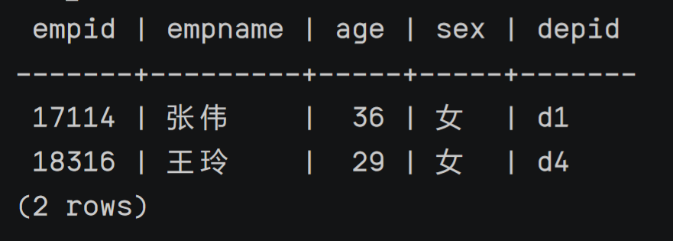
from category left outer join project on project.catid = category.catid

left outer join workson on workson.proid = project.proid

and workson.empid = ‘10102’

group by category.catid;

8. 给出没有参与“软件类”项目女性职工的信息（empid,empname, age,sex,depid)



select empid, empname,age, sex, depid(错，没有约束sex =’女’)

from employee natural join in workson natural join in project natrual join in category

where employee.empid = workson.empid and workson.proid = project.proid and project.catid = (select catid from category wherr catname = ‘软件类’);

select e.empid,e.empname,age,sex,depid

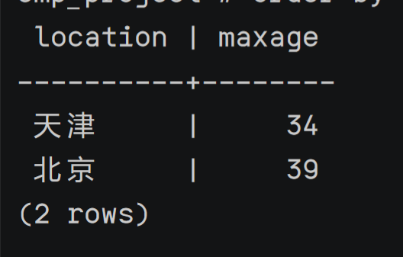
from employee e

where sex = ‘女’ and not exists(

select\* from project p,workson w,category c

where e.empid = w.empid and p.proid = w.proid and c.catid = p.catid and c.catname = ‘软件类’);

9. 给出有30岁以上男性员工的地点名称和该地男员工最大年龄，结果按最大年龄升序排序



select location, age as maxage

from department natural join in employee

where age > 30 and employee.depid = department.depid and age >= all(select age form employee e1,employee e2 where e1.depid = e2.depid)

order by maxage asc;

select location,maxAge

from(select max(age) as maxAge,d.location

from employee e,department d

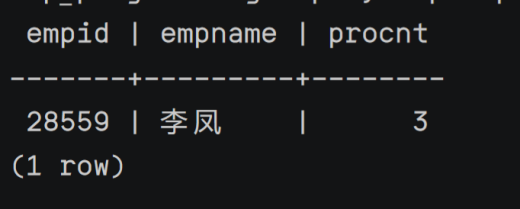
where e.depid = d.depid and e.sex = ‘男’

group by d.location)

where maxAge > 30

order by maxAge asc;

10. 给出在广州工作的、参加“'产品推广'”项目的职员id、姓名及他们参加的项目个数（empid, empname, procnt）



select empid empname,count(proid) as procnt

from workson natural join in employee natural join in project natural join in department

where employee.depid = (select depid from department where location =’广州’) and

employee.depid = workson.empid and workson.proid = (select proid from project where

projectname = ‘产品推广’) ;

selct emp.empid,emp.empname,procnt

from workson, department,project,(select employee.empid,empname,employee.depid,count(proid)as procnt

from employee,workson

where employee.empid = workson.empid

group by employee.empid,empname,employee.depid)emp

where department.location = ‘广州’ and department.depid = emp.depid and workson.proid = project.proid

and projectname = ‘产品推广’ and workson.empid = emp.empid

group by emp.empid,emp.empname,procnt;