**实验五 高级数据查询**

# 实验目的

1. 掌握连接查询的方法
2. 掌握子查询的方法
3. 熟悉如何为表和字段取别名
4. 掌握如何使用正则表达式查询

# 实验内容

1. 使用连接运算符实现多表查询，包括内连接、外连接和复合条件连接查询；
2. 在SELECT语句中嵌套子查询，常使用的操作符有：ANY(SOME)，ALL，IN，EXIST；
3. 为字段和表创建别名并在查询中使用
4. 在检索或替换某个复合要求的文本内容的语句中使用正则表达式；

# 实验作业

1. 创建数据库EmployDB，在库中创建表employee和表dept。

表employee结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段说明 | 数据类型 | 说明 |
| e\_no | 员工编号 | INT | 主键，非空 |
| e\_name | 员工姓名 | VARCHAR(20) | 非空 |
| e\_gender | 员工性别 | CHAR(20) | 非空 |
| dept\_no | 部门编号 | INT | 非空 |
| e\_job | 职位 | VARCHAR(20) | 非空 |
| e\_salary | 薪水 | SMALLINT | 非空 |
| hiredate | 入职日期 | DATE | 允许空值 |

表dept结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 字段说明 | 数据类型 | 说明 |
| d\_no | 部门编号 | INT | 主键，外键，非空 |
| d\_name | 部门名称 | VARCHAR(20) | 非空 |
| d\_location | 部门地址 | VARCHAR(20) | 非空 |

1. 向两个表中插入如下数据。

表employee中记录

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| e\_no | e\_name | e\_gender | dept\_no | e\_job | e\_salary | hiredate |
| 1001 | SMITH | m | 20 | CLERK | 800 | 2005-11-12 |
| 1002 | ALLEN | f | 30 | SALESMAN | 1600 | 2003-05-12 |
| 1003 | WARD | f | 30 | SALESMAN | 1250 | 2003-05-12 |
| 1004 | JONES | m | 20 | MANAGER | 2975 | 1998-05-18 |
| 1005 | MARTIN | m | 30 | SALESMAN | 1250 | 2001-06-12 |
| 1006 | BLAKE | f | 30 | MANAGER | 2850 | 1997-02-15 |
| 1007 | CLARK | m | 10 | MANAGER | 2450 | 2002-09-12 |
| 1008 | SCOTT | m | 20 | ANALYST | 3000 | 2003-05-12 |
| 1009 | KING | f | 10 | PRESIDENT | 5000 | 1995-01-01 |
| 1010 | TURNER | f | 30 | SALESMAN | 1500 | 1997-10-12 |
| 1011 | ADAMS | m | 20 | CLERK | 1100 | 1999-10-05 |
| 1012 | JAMES | m | 30 | CLERK | 950 | 2008-06-15 |

表dept中的记录

|  |  |  |
| --- | --- | --- |
| d\_no | d\_name | d\_location |
| 10 | ACCOUNTING | Shanghai |
| 20 | RESEARCH | Beijing |
| 30 | SALES | Shenzhen |
| 40 | OPERATIONS | Fujian |

1. 查询employee表中所有女员工（F）的薪水。
2. 查询employee表中销售人员（SALESMAN）的最低工资。
3. 查询employee表中名字以字母N或S结尾的记录。
4. 查询员工BLAKE所在部门和部门所在地。
5. 查询所有员工的部门和部门信息。
6. 查询所有2001到2005年入职的员工信息，查询部门编号为20和30的员工信息并使用UNION合并两个查询结果。
7. 使用LIKE查询员工姓名中包含字母a的记录。
8. 使用~查询员工姓名中包含T、C或者M三个字母中任意一个的记录。
9. 查询employee表中查询员工姓名以字母A或S开头的员工信息。

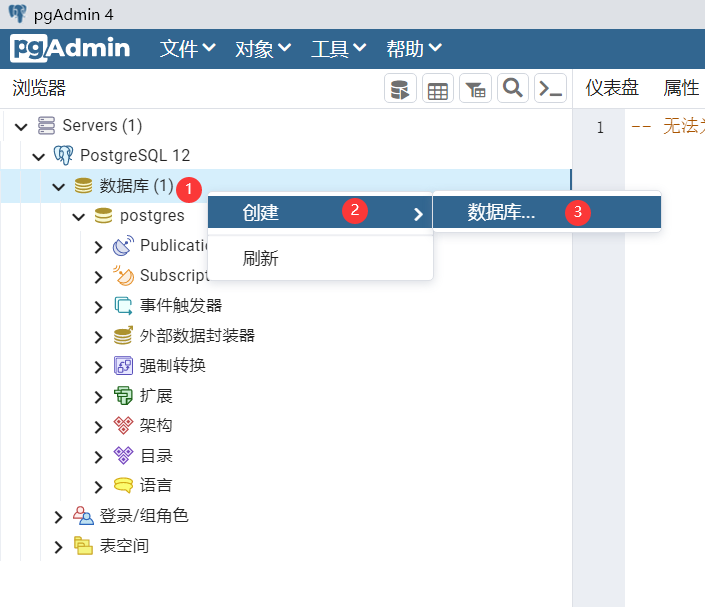
# 思考与体会

1. 在相等联接查询中，在查询某些列时是否可以忽略列前的表名呢？
2. ORDER BY可以和LIMIT混合使用吗？如果可以，顺序是怎样的？

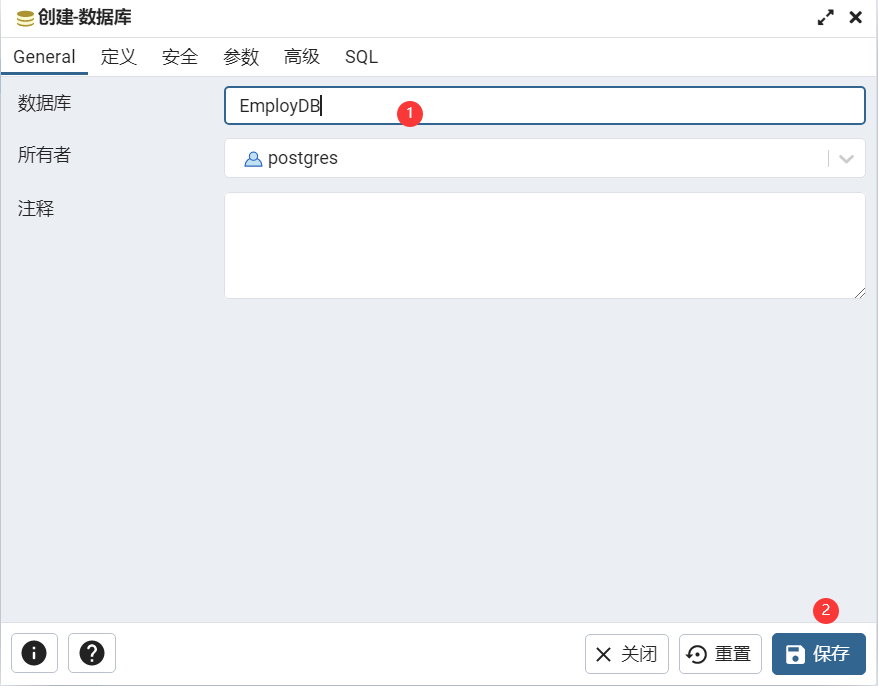
**实验报告**

1. 创建数据库EmployDB，在库中创建表employee和表dept。

Step1.右击数据库，点击创建，选择“数据库...”



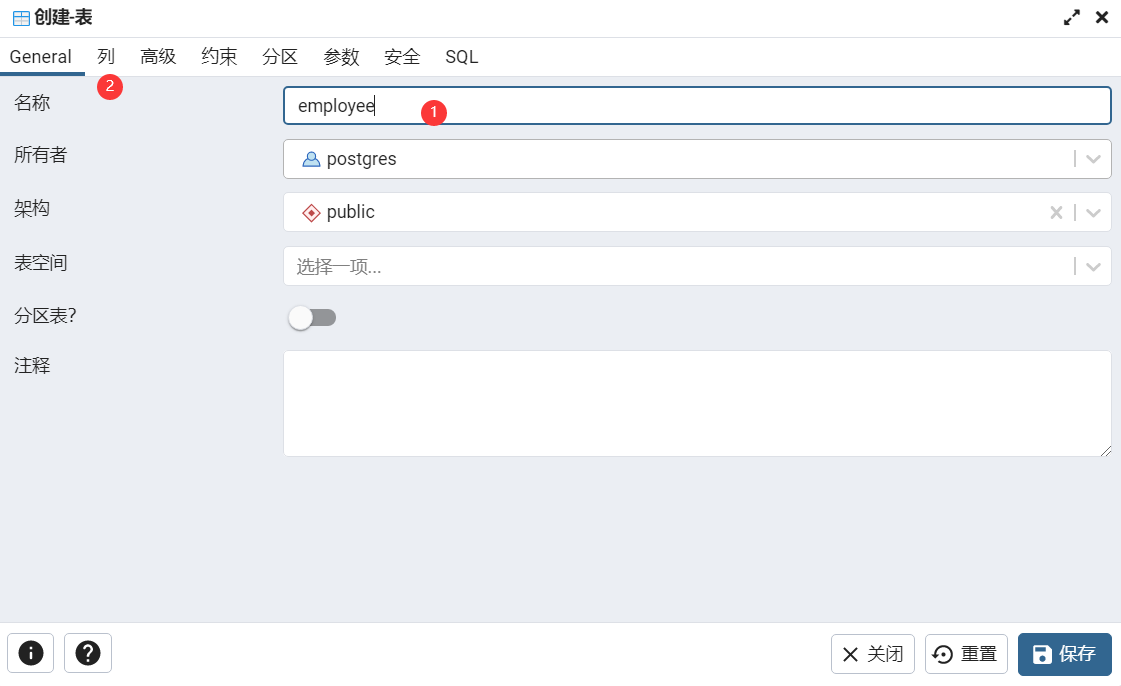
Step2.输入数据库名称，点击保存



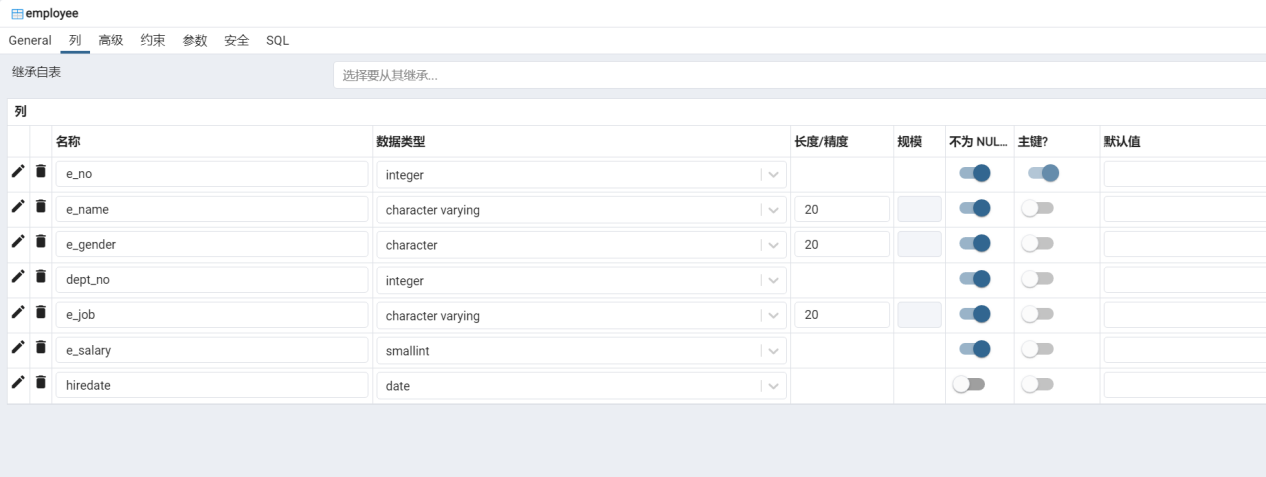
Step3.在EmployDB下选择架构，在public右击，选择创建表；



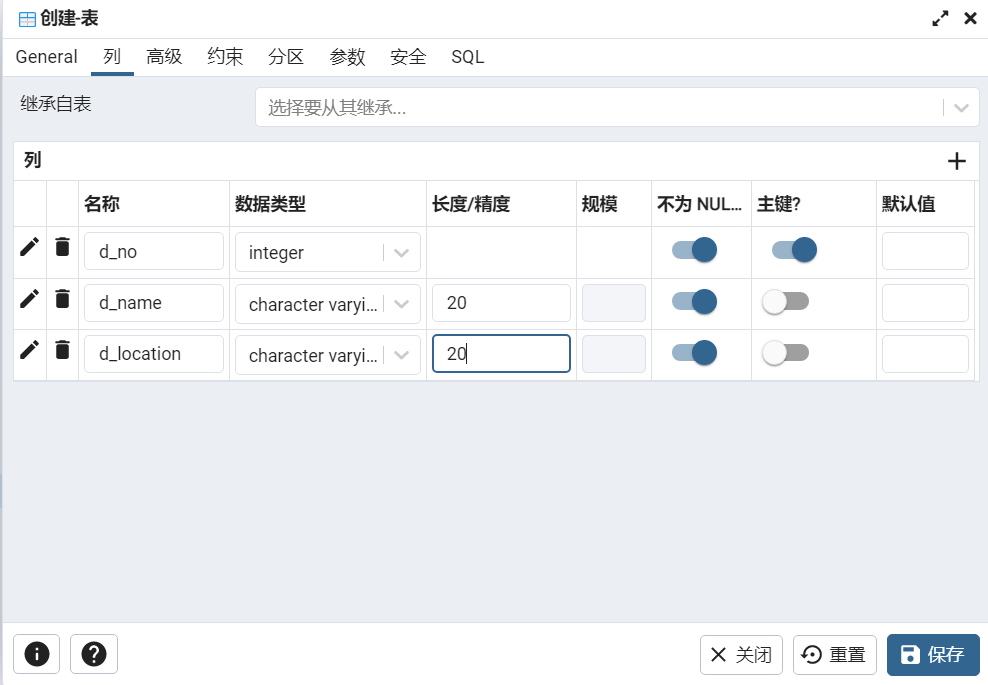
Step4.输入表名称，然后添加属性，点击保存，建立employee表；



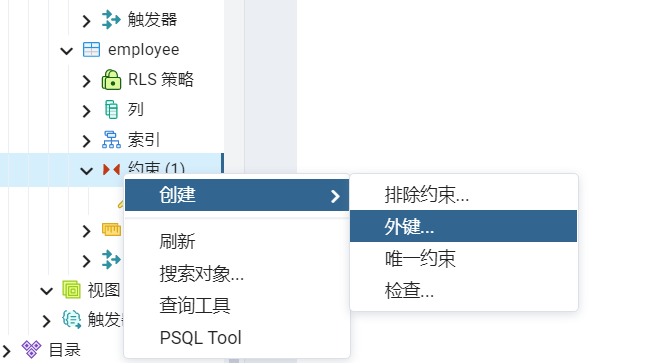




同样的步骤，建立表dept



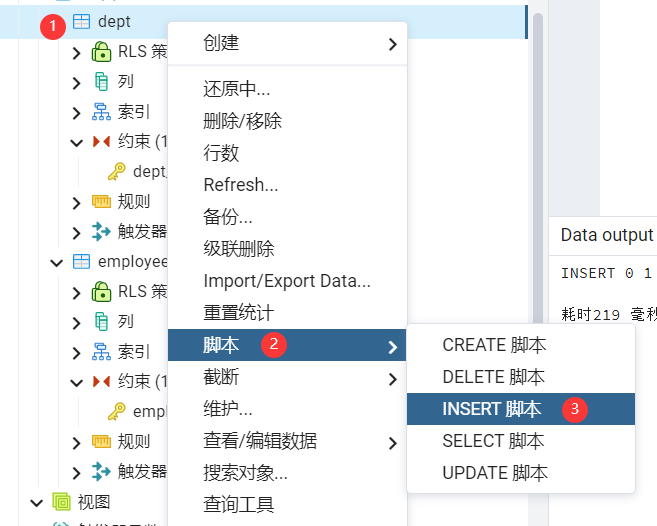
Step5.设置外键，选择employee表下的约束，创建外键



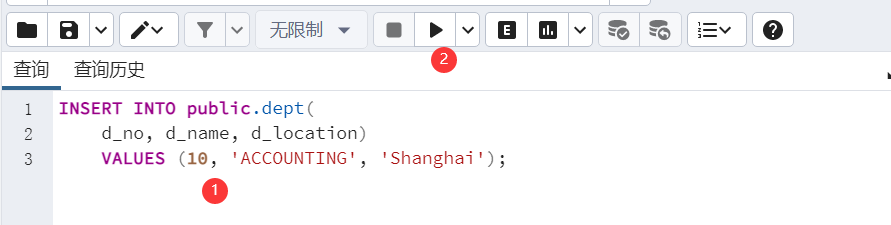


1. 向表中插入数据。

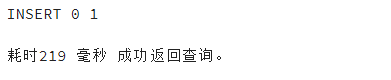
Step1.方法一，右击表，选择脚本-->INSERT脚本插入



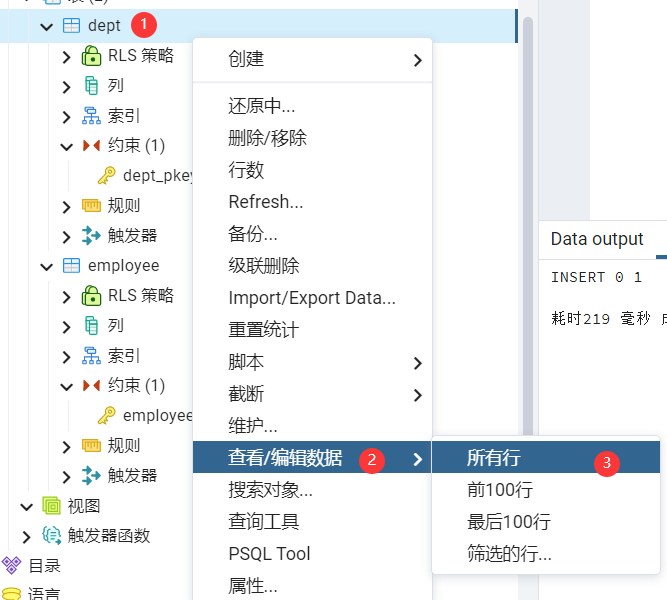
输入信息后点击运行即可



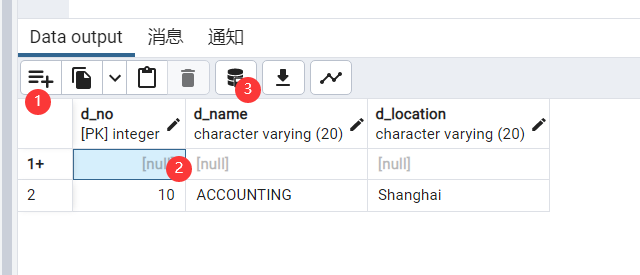
显示这个说明插入成功；

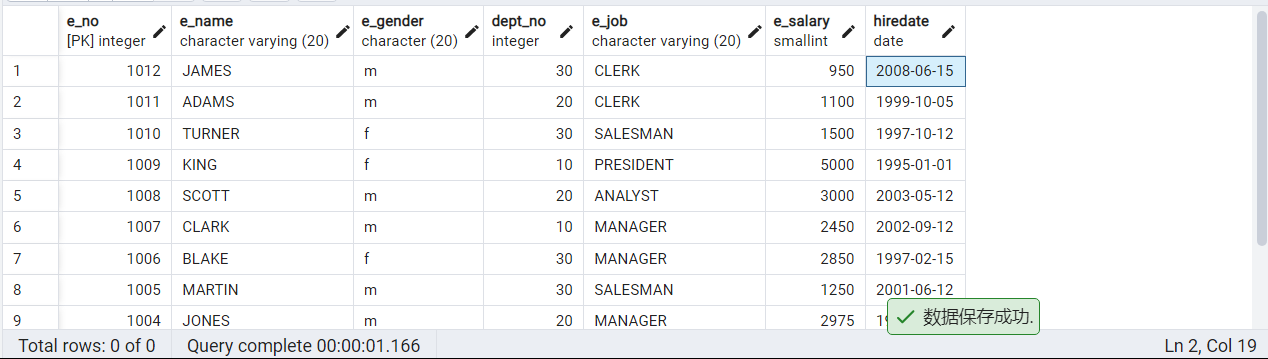
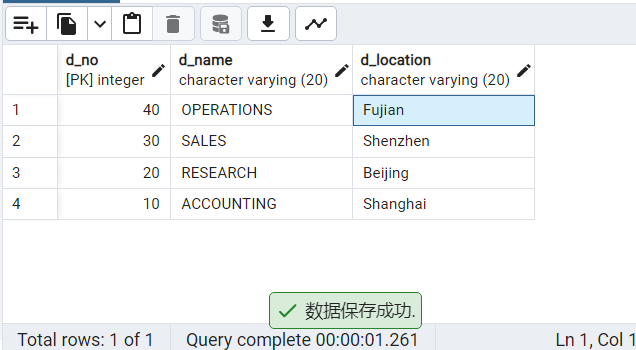


Step2.方法2：



点击新建，输入数据，保存即可；

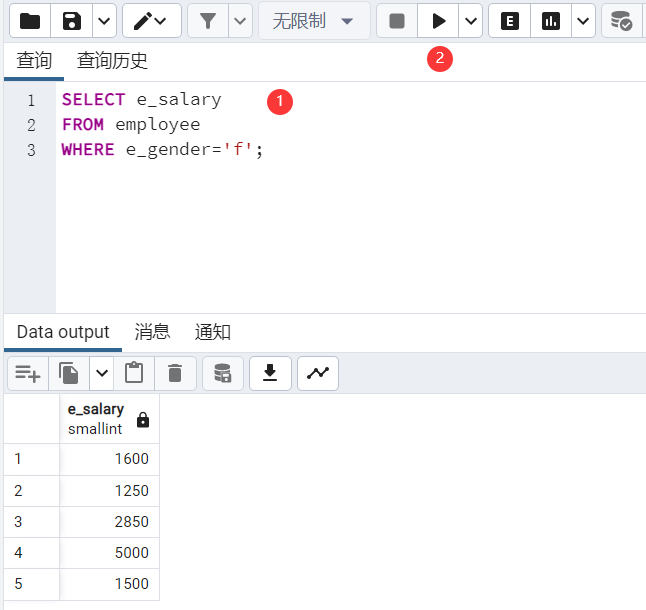




1. 查询表中所有女员工的薪水。

输入如下代码，即选择e\_salary这一列，并且选择性别为‘f’的行；

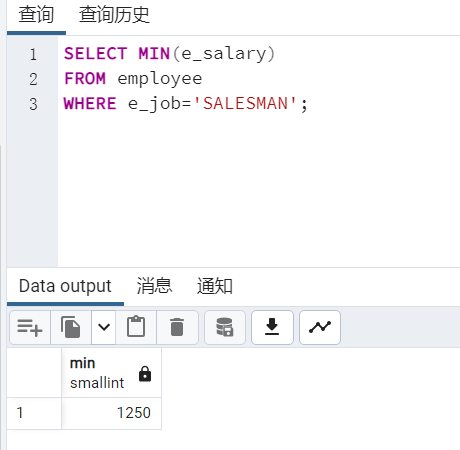
结果如图所示：



1. 查询employee表中销售人员(SALESMAN)的最低工资；

代码如下：使用了MIN()函数选择e\_salary，并指定职业为销售人员；

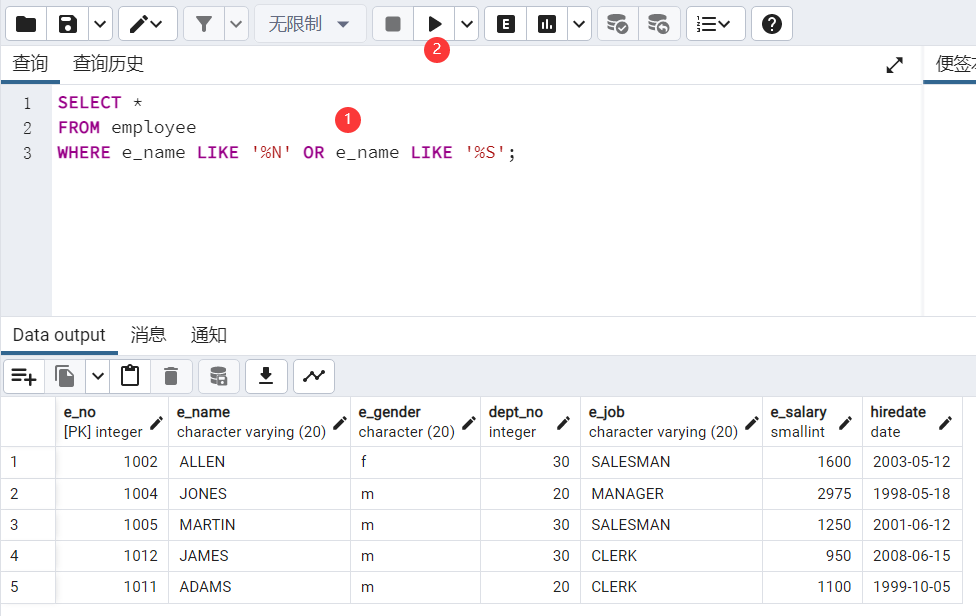
结果如下：



1. 查询employee表中名字以字母N或S结尾的记录；

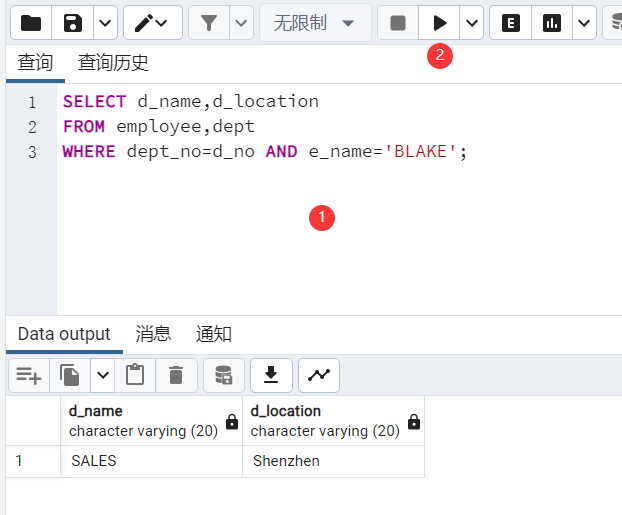
代码如下：使用LIKE语句，并使用通配符%（替代 0 个或多个字符）写出所示格式，“%N”意思就是以N结尾的串；

结果如下：



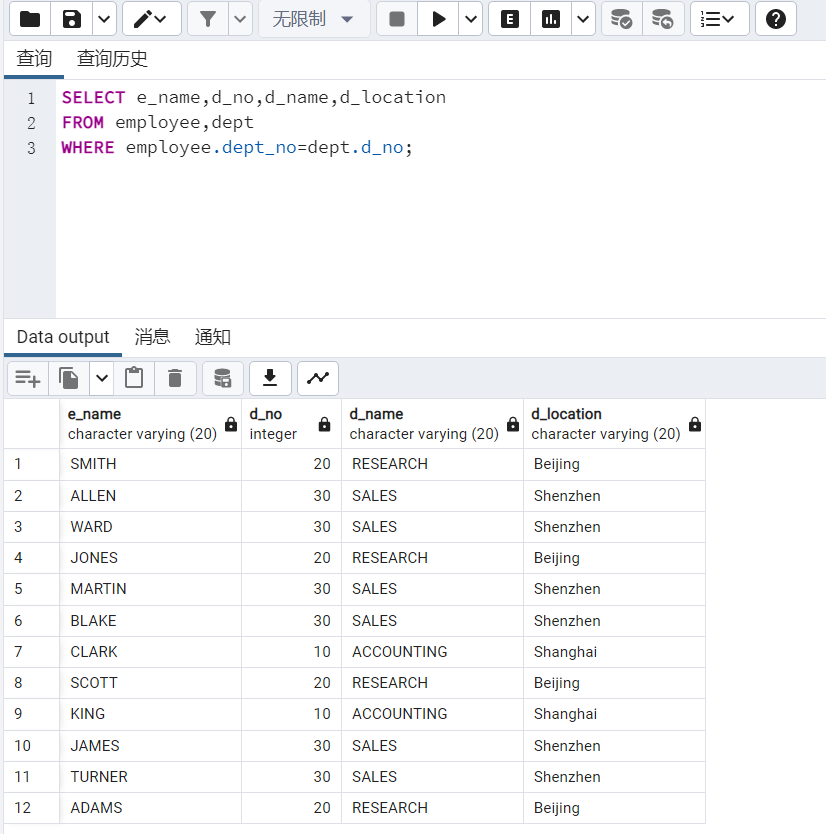
1. 查询员工BLAKE所在部门和部门所在地；

需要的查询的结果列为d\_name和d\_location，但是姓名BLAKE在employee表中，而部门信息在dept表中，因此选择对两个表进行笛卡尔积，并选择出部门编号相等的行来（去除不对的匹配），然后选择名字为BLAKE的员工，输出其d\_name和d\_location列；



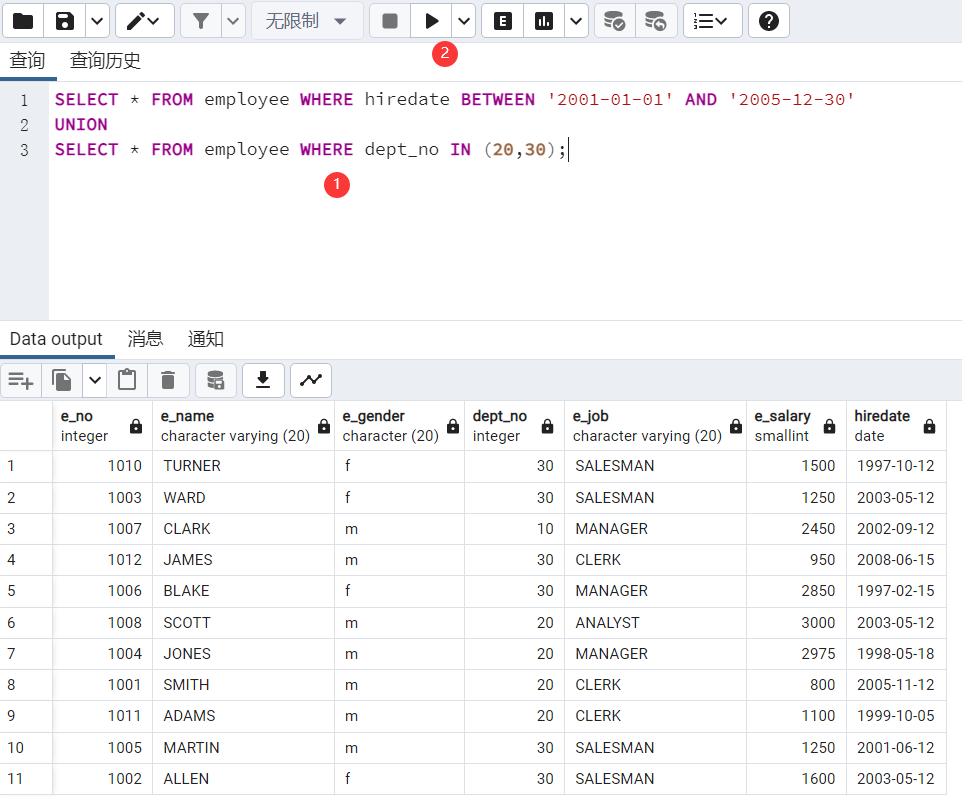
1. 查询所有员工的部门和部门信息；

部门信息在dept表中，因此需要先进行笛卡尔积运算（等值连接也可），再选择其中部门号相等的信息，选取其姓名和部门信息列输出；



1. 查询所有2001到2005年入职的员工信息，查询部门编号为20和30的员工信息并用UNION合并两个查询结果；

2001到2005年即雇用日期在2001-01-01到2005-12-30之间，dept\_no IN (20,30)即选择部门号为20或30；最后用UNION链接；

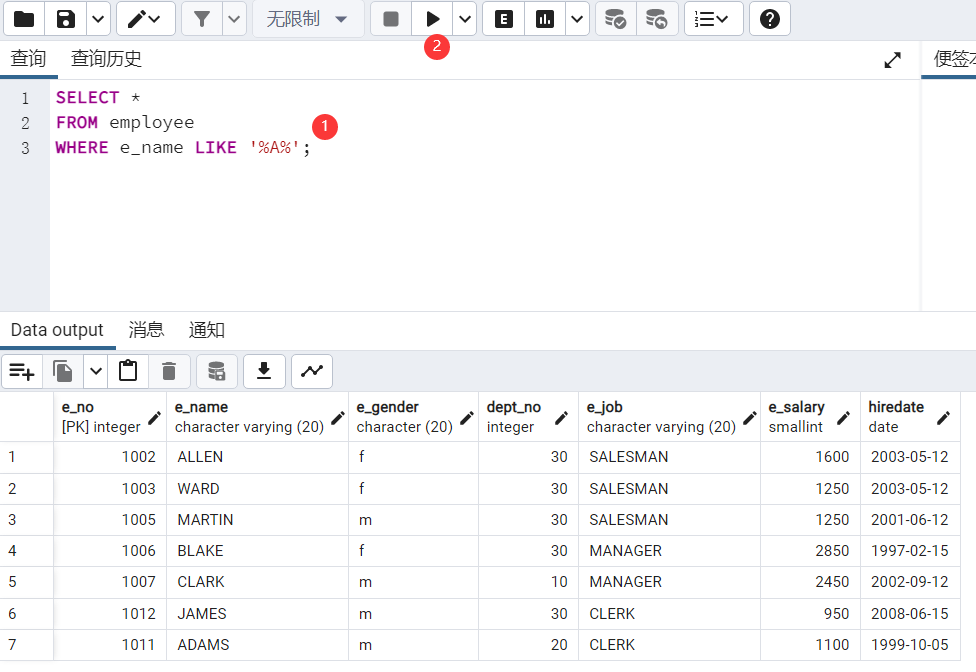


1. 使用LIKE查询员工姓名中包含字母a的记录；

由于表中员工姓名均为大写，用

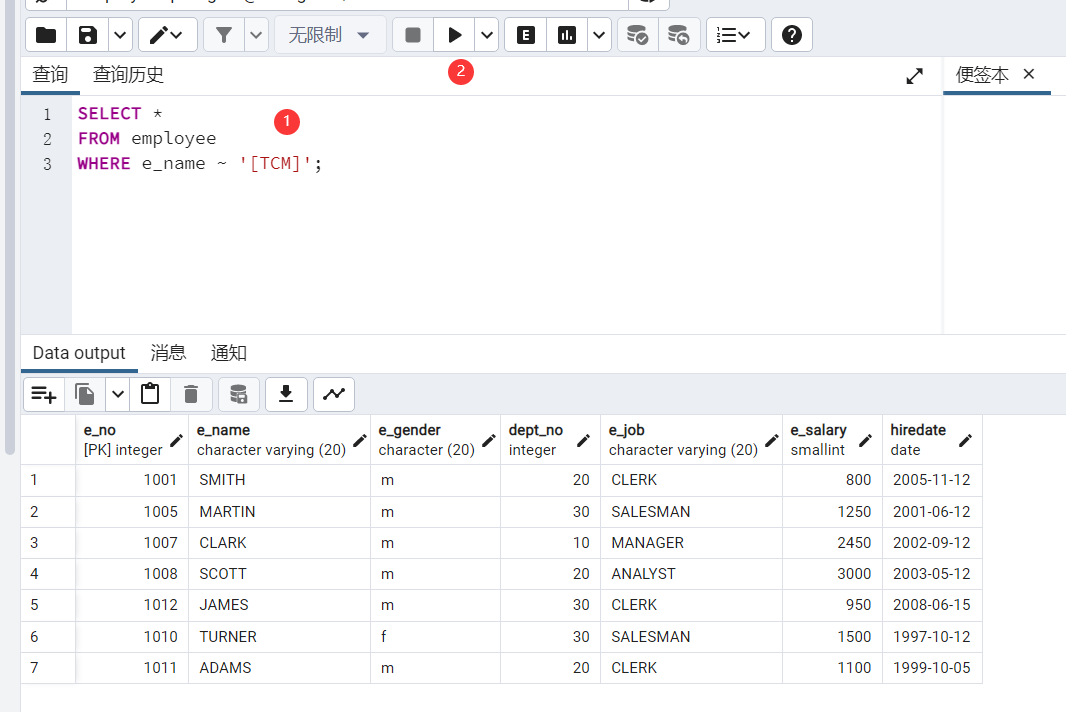
**SELECT \* FROM employee WHERE e\_name LIKE 'a%';**

查询后没有任何结果，因此选择将a改成A进行查询；包含字母A，则需要使用通配符%，包含A，即‘%A%’，查询即可；



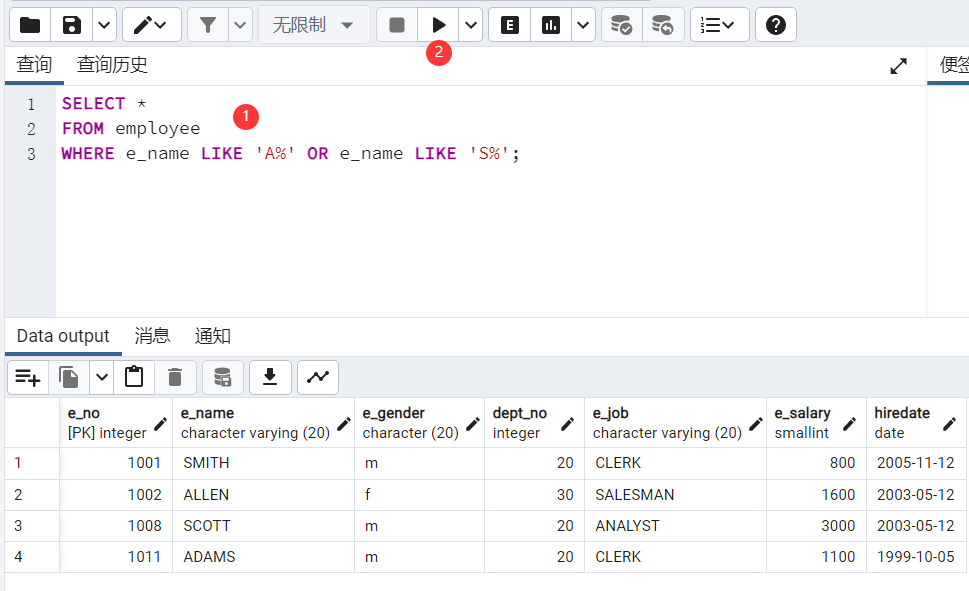
1. 使用~查询员工姓名中包含T、C或者M三个字母中任意一个的记录；

postgresql中使用正则表达式时需要使用‘~’，再在后面添加正则表达式，即通配符’[content]’，意思是在[]里出现的字母都可以识别到字符列的任何单一字符；代码和结果如下图：



1. 查询employee表中员工姓名以字母A或S开头的员工信息。

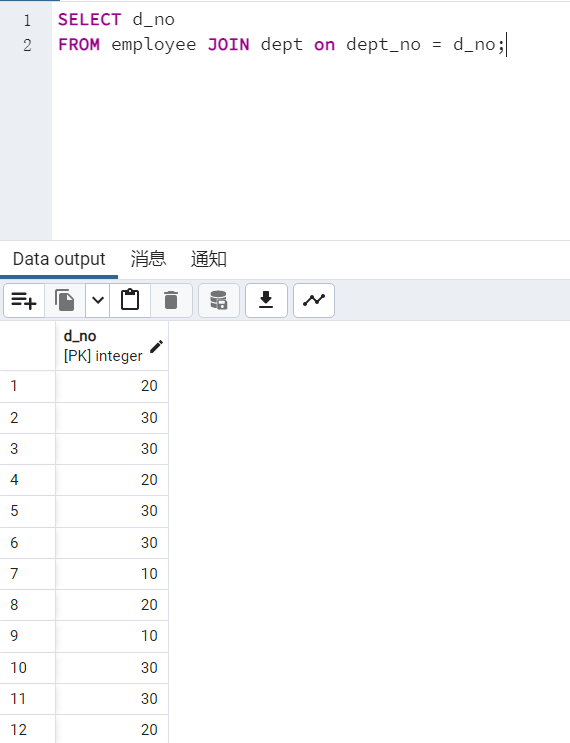
使用通配符%，以某个字母X开头的写法为’X%’，再使用或者OR联系两个条件，进行查询；



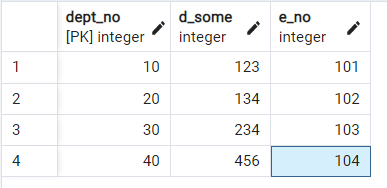
# 思考与体会

1. 在相等联接查询中，在查询某些列时是否可以忽略列前的表名呢？

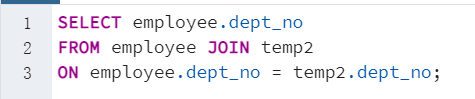
在本实验的例子中，因为不存在列名相等的情况，因此可以忽略列前的表名，查询如下；



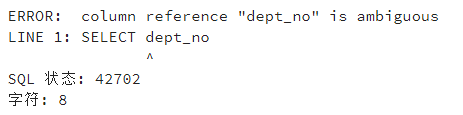
当存在同名列时，如新建的表temp2；



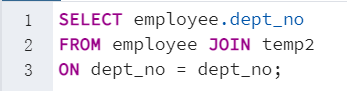
查询时写全表名，可以执行；



选择的表列没有前缀时，会出现同名列冲突的情况；



当条件不包含表名时，也会出现冲突



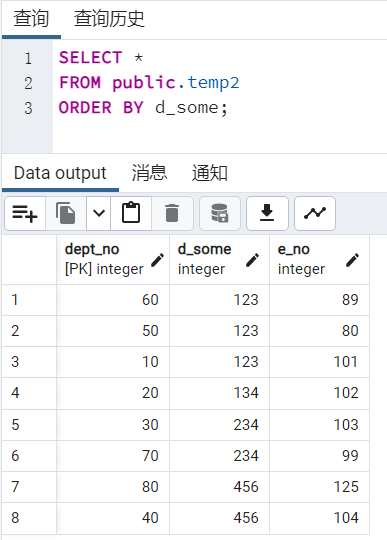
总结：当两个表之间不存在同名列时，此时每列的位置与所属数据表可知，因此可以忽略列前的表名；但查询语句中出现同名列时，必须加上表名加以区分；

1. ORDER BY可以和LIMIT混合使用吗？如果可以，顺序是怎样的？

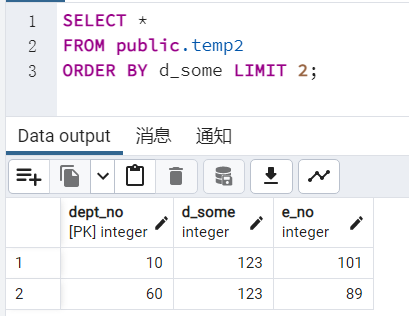
**可以**，但不建议；

当表中的行存在有相同列值时，执行ORDER BY查询和用LIMIT返回的查询结果可能会不一样；

例如，在temp2表中插入有列值相等的行，使用ORDER BY排序后，发现顺序为

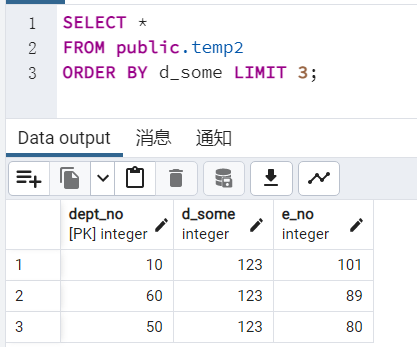


但是使用LIMIT语句后，可以成功运行，但是输出结果为：



与排序结果不同（按刚才的结果应该会输出dept\_no为60和50）

由于前三个的d\_some是一样的，这里再看LIMIT=3的情况：



发现虽然输出的是刚才排序过的前三的，但是顺序发生了变化；

因此，如果ORDER BY时列有相同的值，那么会自由地以任何顺序返回这些行，如果想让顺序相对确定下来，可以在ORDER BY时添加附加列以消除随机性。

**因此两个语句一起使用不会报错，使用顺序是先ORDER BY，再使用LIMIT；**

顺序反了会报错：

