

数据库系统课程实验报告

|  |  |
| --- | --- |
| 实验名称： | 实验四：数据高级查询 |
| 实验日期： | 2022/4/8 |
| 实验地点： | 厦门大学德旺图书馆 |
| 提交日期： | 2022/4/8 |
|  | |
| 学号： | 20420192201952 |
| 姓名： | 庾晓萍 |
| 专业年级： | 软工2020级 |
| 学年学期： | 2021-2022学年第二学期 |

1. 实验目的

* 熟练掌握设计正确的 SQL 查询语句以实现数据高级查询的方法
* 熟练掌握 openGauss 连接查询、子查询和集合查询的语法结构及使用方法

-（内）连接、（全）外连接、左外连接、右外连接

-子查询（嵌套查询）

-不相关子查询与相关子查询

-EXISTS/NOT EXISTS

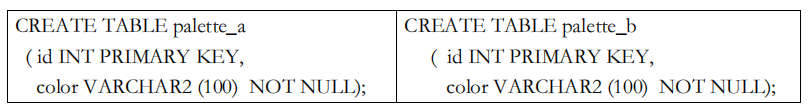
-ANY

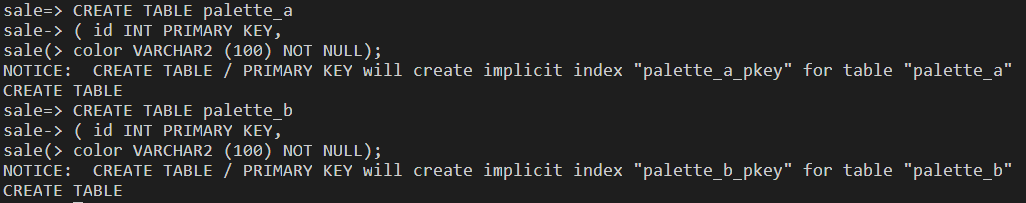
-ALL

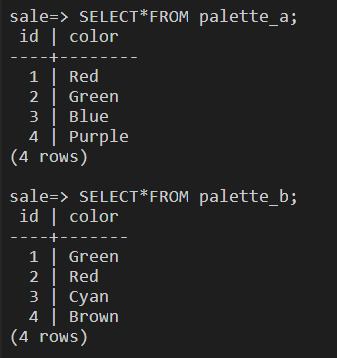
-集合运算：UNION、INSERSECT、MINUS/EXCEPT

* 理解不相关子查询与相关子查询的不同，掌握构造相应 SQL 语句的方法
* 熟练掌握基于派生表的查询方法
* 建议：对同一查询要求尽量使用不同的查询语句实现。如，所有带 IN 谓词、比较运算符、ANY 或 ALL 谓词的子查询都能用带 EXISTS 谓词的子查询等价替换。

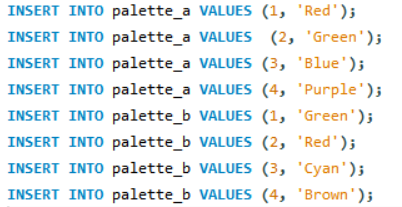
1. 实验内容和步骤
2. 创建两张表 palette\_a 和 palette\_b（结构相同，但表名不同，color 为颜色）

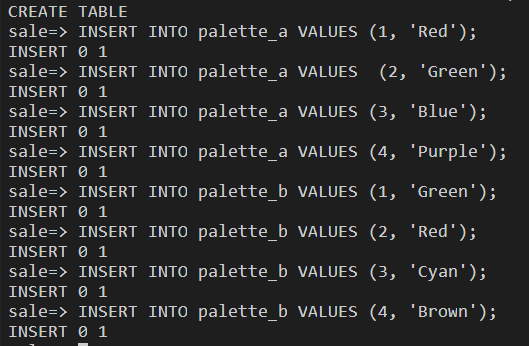




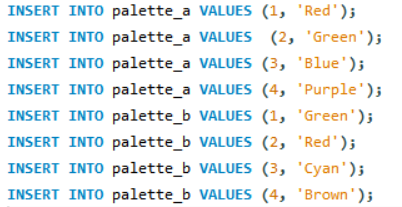


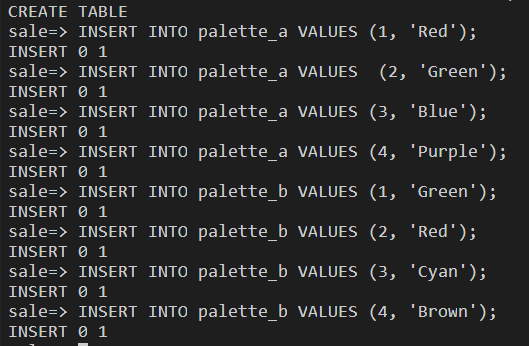
1. 为表 palette\_a 添加样例数据：{(1, 'Red'), (2, 'Green'), (3, 'Blue'), (4, 'Purple')}



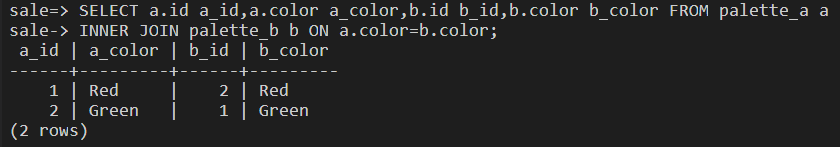


1. 为表 palette\_b 添加样例数据：{(1, 'Green'), (2, 'Red'), (3, 'Cyan'), (4, 'Brown')}

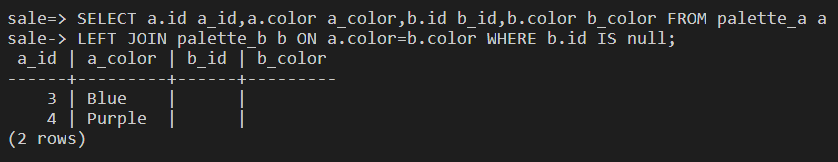




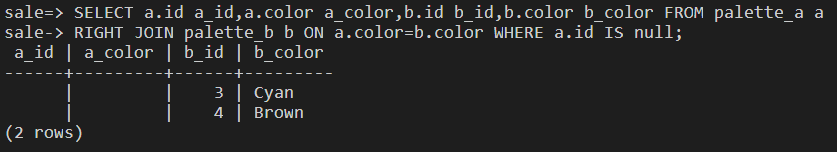
1. 查询两张表中相同颜色的所有信息



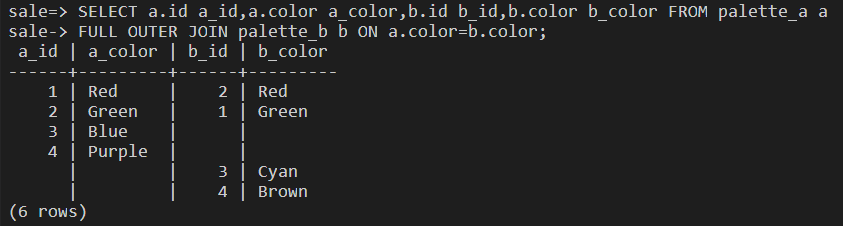
1. 查询 palette\_a 表中颜色不出现在 palette\_b 表中的两张表的 id 和颜色（用左外连接）



1. 查询 palette\_b 表中颜色不出现在 palette\_a 表中的两张表的 id 和颜色（用右外连接）

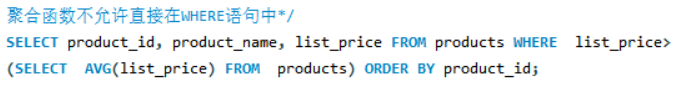


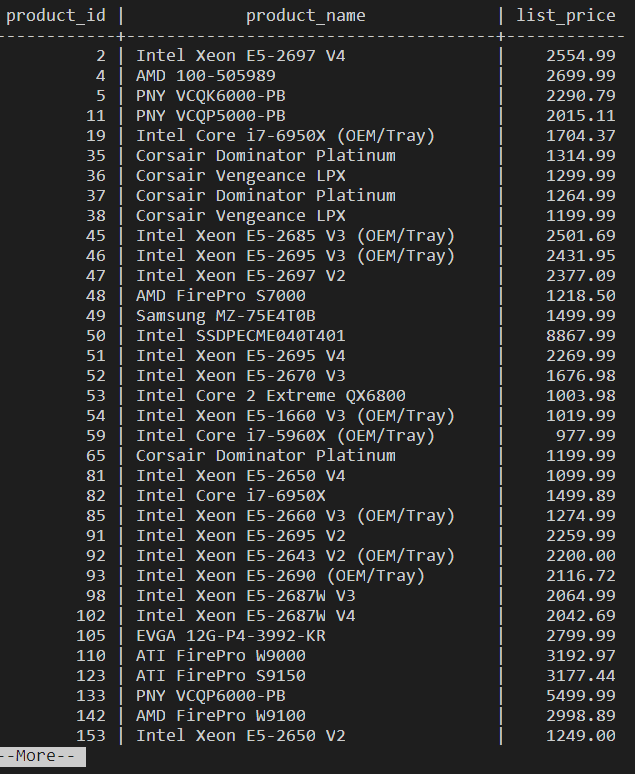
1. 查询（5）或（6）两种情况的信息（用（全）外连接）



1. 查询产品表 products 中的 product\_id, product\_name, list\_price 信息，要求产品定价 list\_price 大于其平均定价 list\_price。

思路：聚合函数不允许直接在WHERE语句中，故首先子查询返回所有产品的平均标价。其次外部查询获取标价大于子查询返回的平均标价的产品。



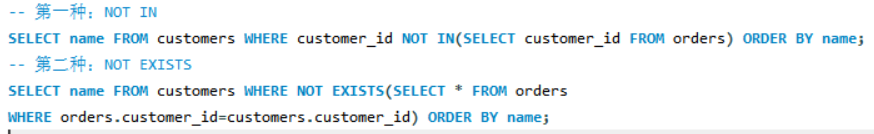


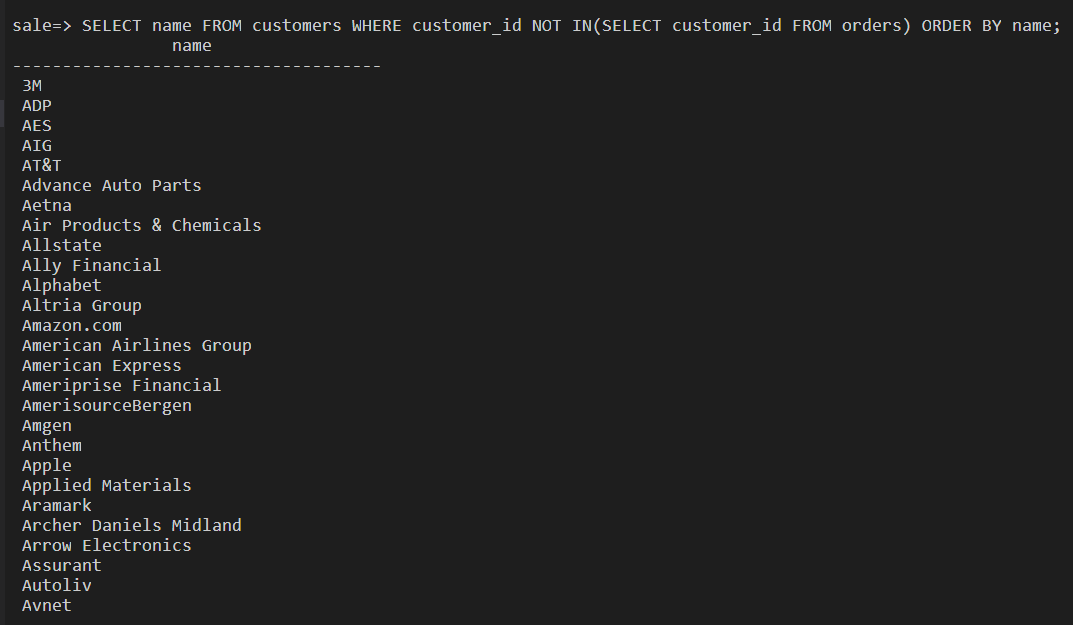
1. 查询没有一个订单的顾客姓名（实现要求：NOT IN（必须）+其它查询方法（如果找到））

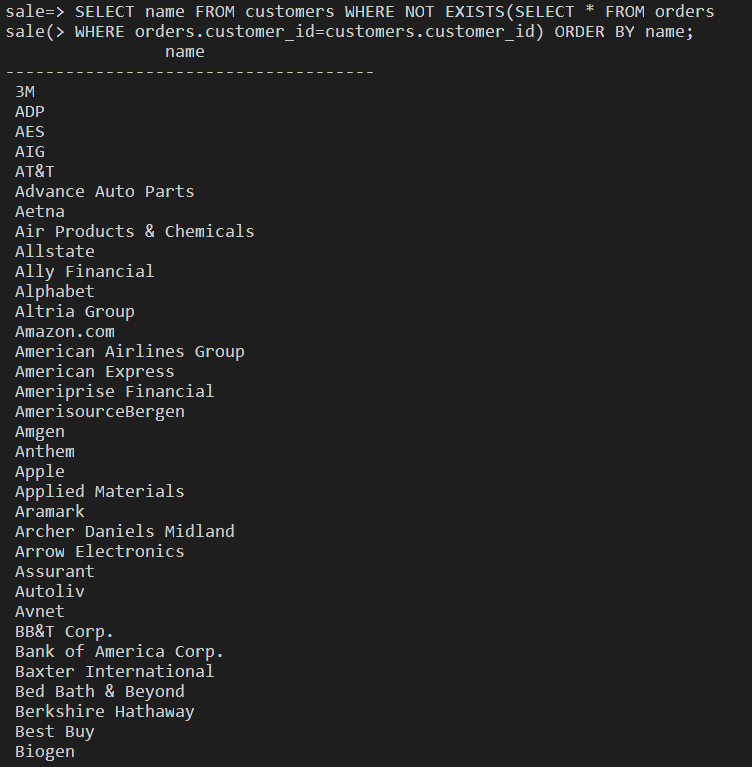
第一种可以使用NOT IN，SELECT name FROM customers WHERE customer\_id NOT IN(SELECT customer\_id FROM orders) ORDER BY name; 第二种可以使用NOT EXISTS，SELECT name FROM customers WHERE NOT EXISTS(SELECT \* FROM orders

WHERE orders.customer\_id=customers.customer\_id) ORDER BY name;

输出的结果相同。





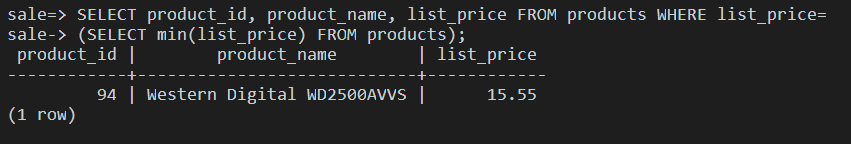


1. 查询产品表 products 中最便宜产品的 product\_id, product\_name, list\_price。

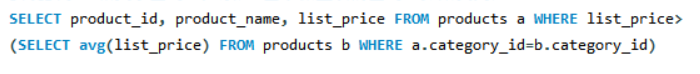
思路：首先独立执行子查询。其次Oracle 只对子查询求值一次。子查询返回结果集后，外部查询使用它们。换句话说，外部查询依赖于子查询。但是，子查询是独立的，不依赖于外部查询的值。

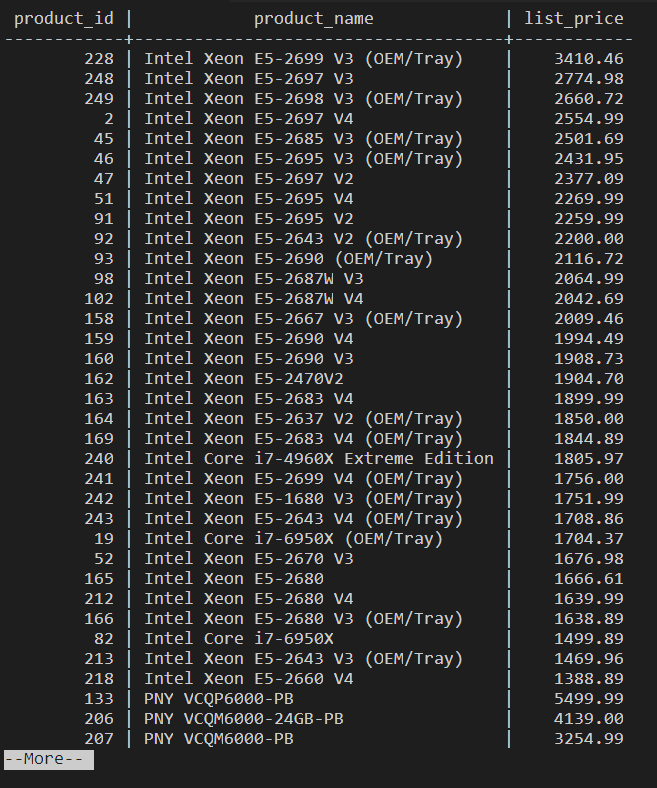
相关子查询是使用来自外部查询的值的子查询。因此，使用相关子查询的查询可能会很慢。





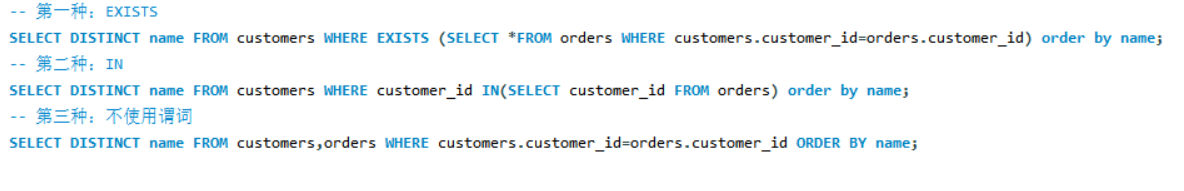
1. 查询产品表 products 中产品的 product\_id, product\_name, list\_price，要求产品定价 list\_price 大于其同类产品（可由 category\_id 表达）的平均定价。实现要求：相关子查询（必须）+基于派生表的查询（如果找到）

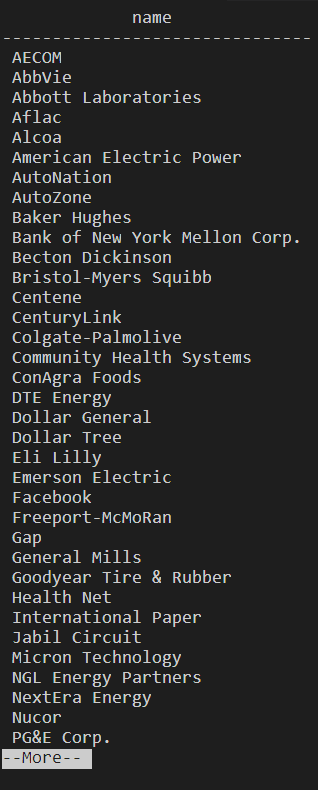
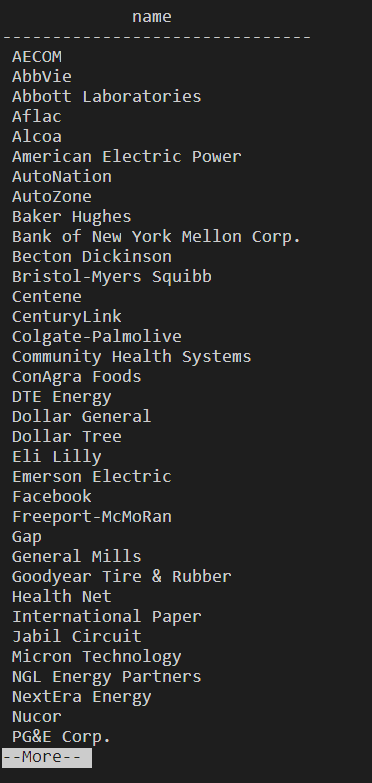
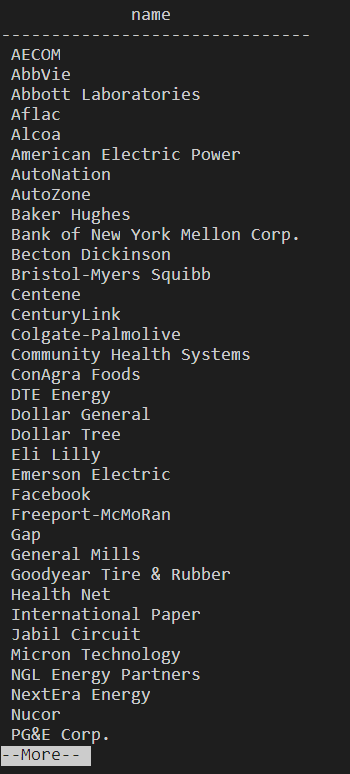




1. 查询有订单 order 的所有顾客 customer 姓名（查询涉及 customers 表和 orders 表）实现要求：使用 EXISTS（必须）+其它查询方法（如果找到）

可以使用EXISTS、IN或者不用谓词，结果输出相同。



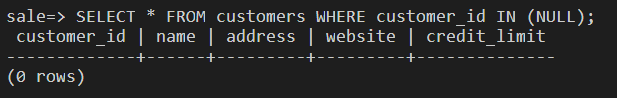


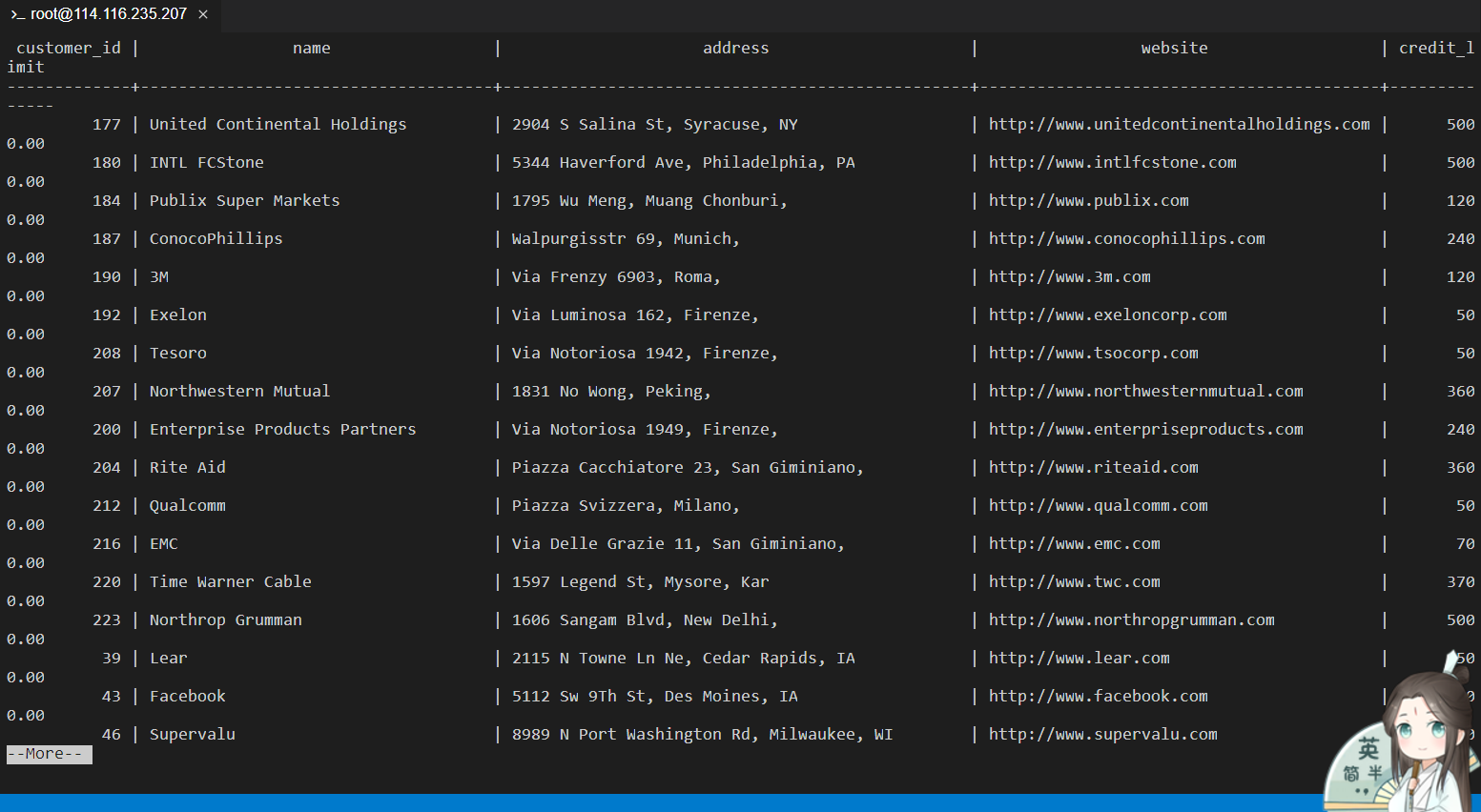
1. 执行以下两条语句，观察有何不同，能否得出某些初步结论？

结论：一旦子查询返回第一行，EXISTS运算符就会停止扫描行，因为它可以确定结果，而IN运算符必须扫描子查询返回的所有行才能得出结果。此外，IN子句不能将任何内容与NULL 值进行比较，但该EXISTS子句可以将所有内容与 NULL 值进行比较。例如，第一个语句不返回任何行，而第二个语句返回customers表中的所有行。通常，当子查询的结果集很大时，EXISTS运算符比IN运算符快。相比之下，当子查询的结果集较小时，IN运算符比EXISTS运算符快。

（注释：openguass不需要FROM dual，可以将FROM dual直接删去）

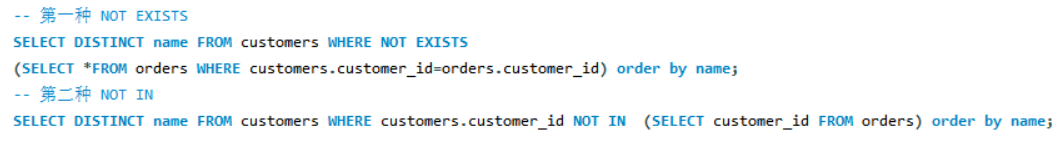


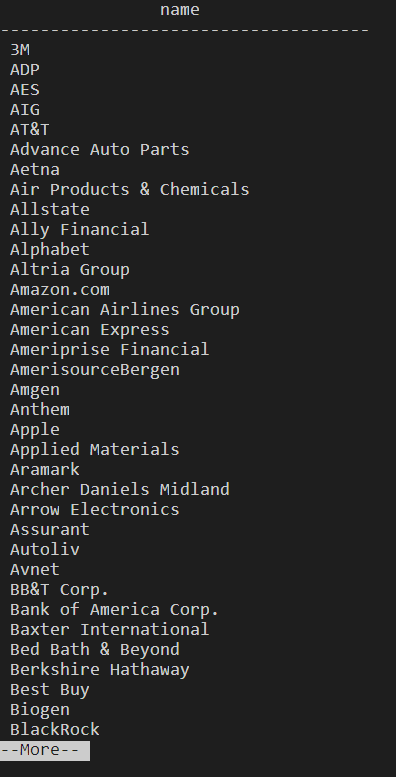




1. 找出所有没有订单的顾客姓名（查询涉及 customers 表和 orders 表）实现要求：使用 NOT EXISTS（必须）+其它查询方法（如果找到）

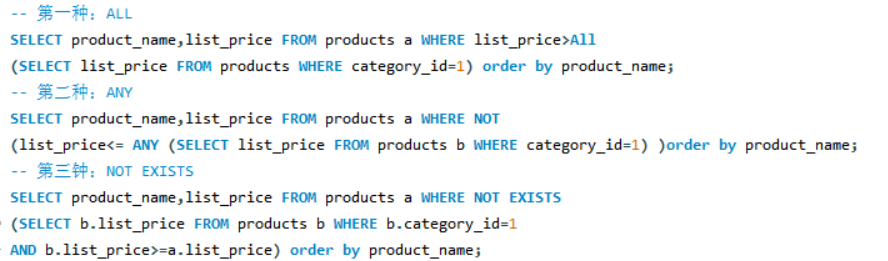
思路：可以使用NOT IN和NOT EXISTS两种方法。

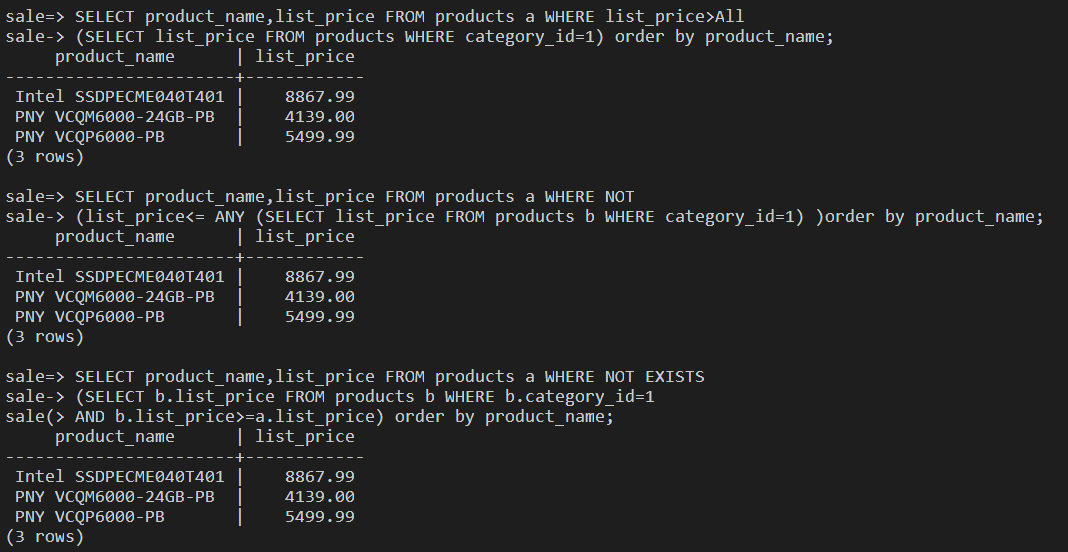




1. 查询产品表 products 中的产品名 product\_name 和定价 list\_price，要求其定价高于产品种类 1 中的任何产品定价。实现要求：ANY（必须）+其它查询方法（如果找到）

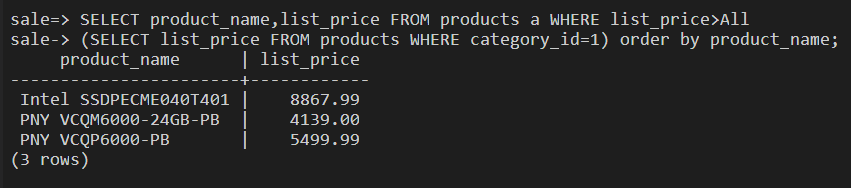
思路：可以使用三种方法：ALL，ANY，NOT EXISTS。



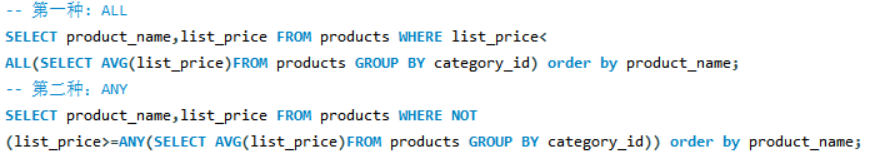


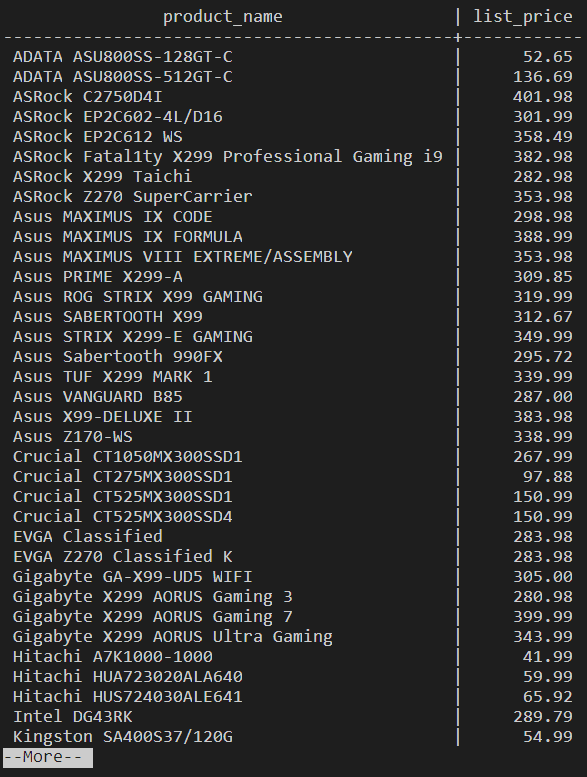
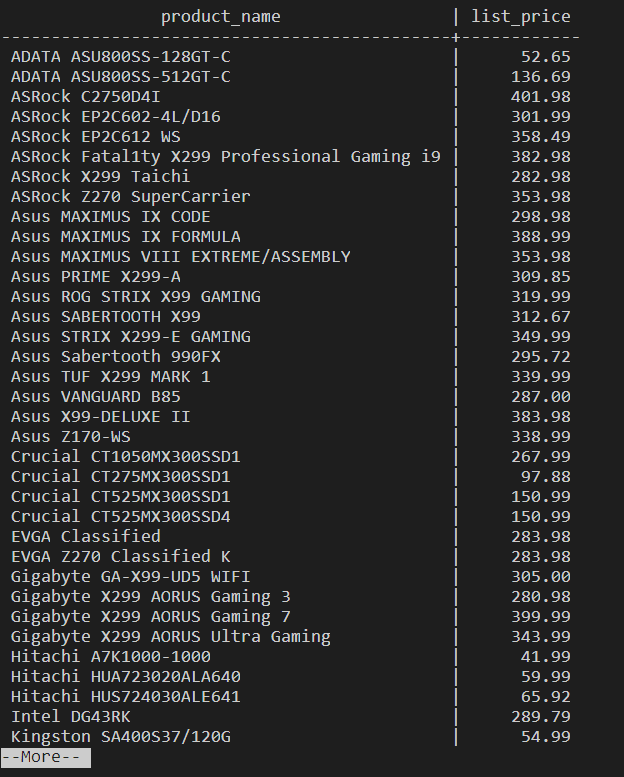
1. 查询产品表 products 中的产品名 product\_name 和定价 list\_price，要求其定价高于产品种类 1 中的所有定价。



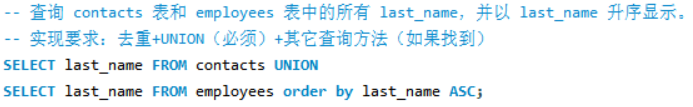
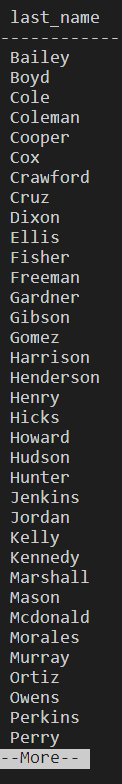


1. 查询产品表 products 中的产品名 product\_name 和定价 list\_price，要求其定价低于产品种类的所有平均定价。实现要求：ALL（必须）+其它查询方法（如果找到）

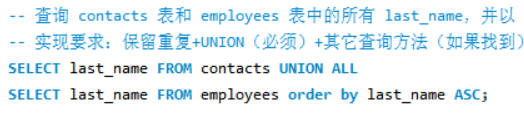
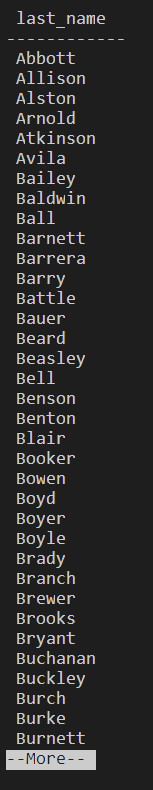




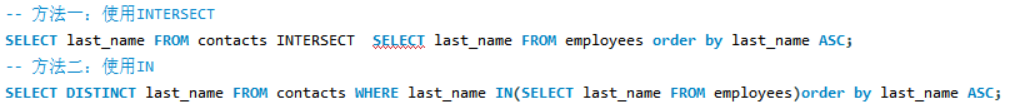
1. 查询 contacts 表和 employees 表中的所有 last\_name，并以 last\_name 升序显示。实现要求：**去重**+UNION（必须）+其它查询方法（如果找到）

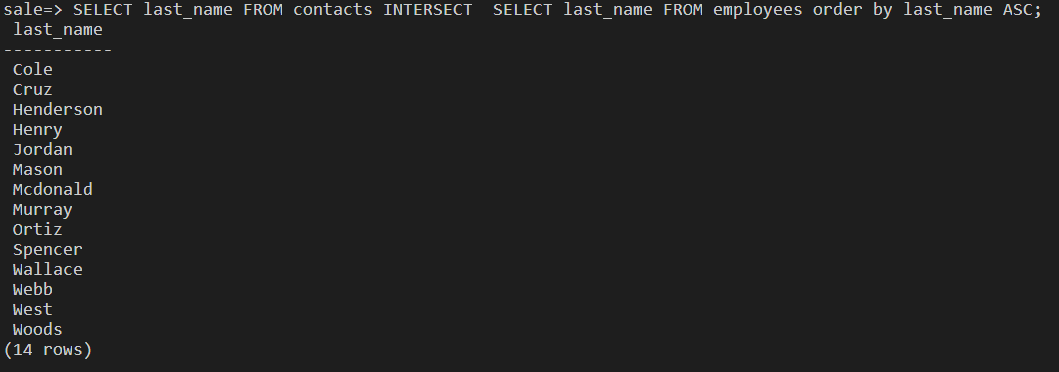


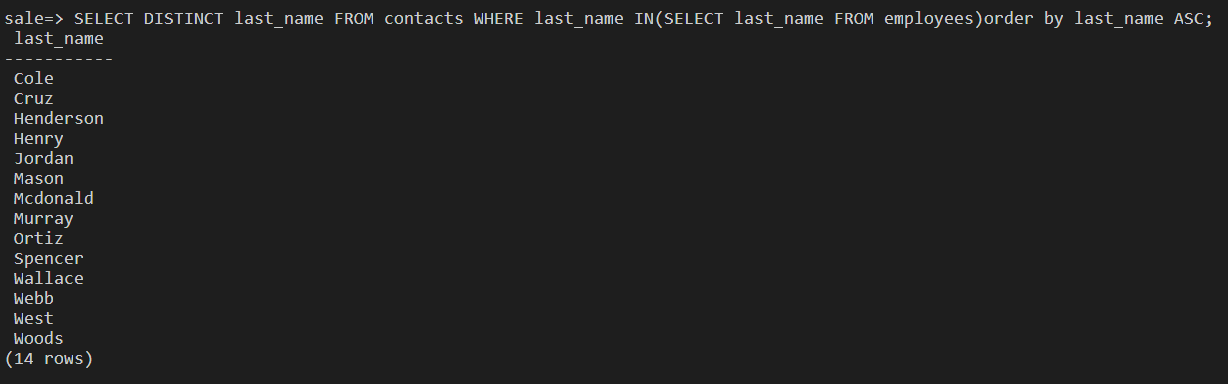
1. 查询 contacts 表和 employees 表中的所有 last\_name，并以 last\_name 升序显示。实现要求：**保留重复**+UNION ALL（必须）+其它查询方法（如果找到）



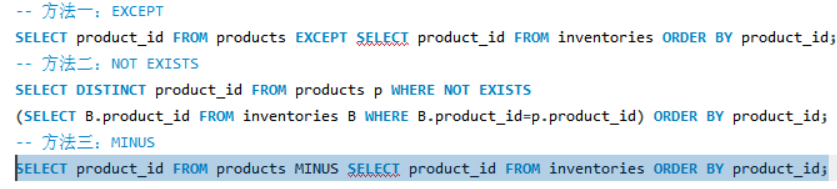
1. 查询同时出现在 contacts 表和 employees 表中的所有 last\_name。实现要求：INTERSECT（必须）+其它查询方法（如果找到）

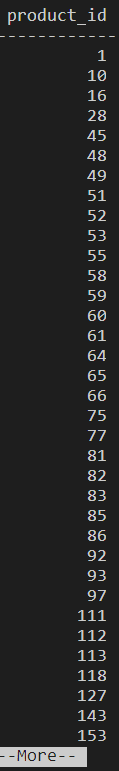
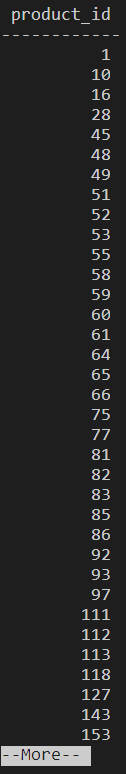
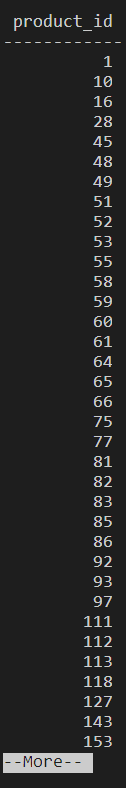






1. 查询在产品表 products 中而不在库存表 inventories 中的产品号 product\_id。实现要求：MINUS/EXCEPT（必须）+其它查询方法（如果找到）





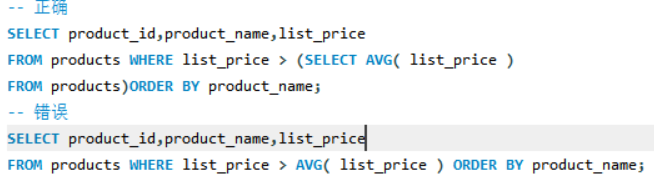
1. 实验总结

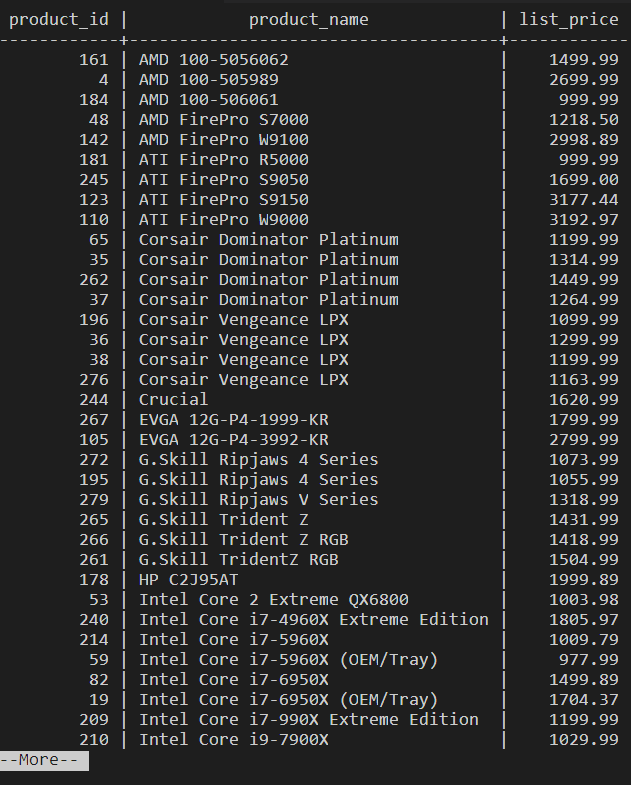
3.1 实验思考

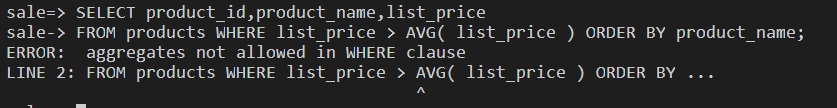
· 什么类型的查询只能用子查询实现？试举例说明。

答：Oracle 子查询可以帮助构建更具可读性的查询，并允许在不使用复杂联接或联合的情况下编写查询，允许以可以隔离每个部分的方式构建复杂的查询。什么类型的查询只能用子查询实现：

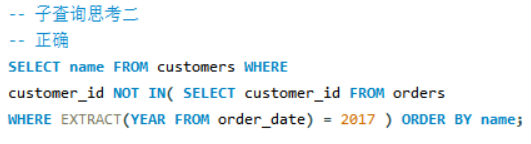
① 使用比较运算符的子查询，例如 >、>=、<、<=、<>、= 通常包括聚合函数，因为聚合函数返回单个值，可用于WHERE外部子句中的比较询问。比如下面的代码，如果直接在WHERE中使用AVG会出现aggregates not allowed in WHERE clause的错误，而通过子查询就可以完美地解决这个问题。

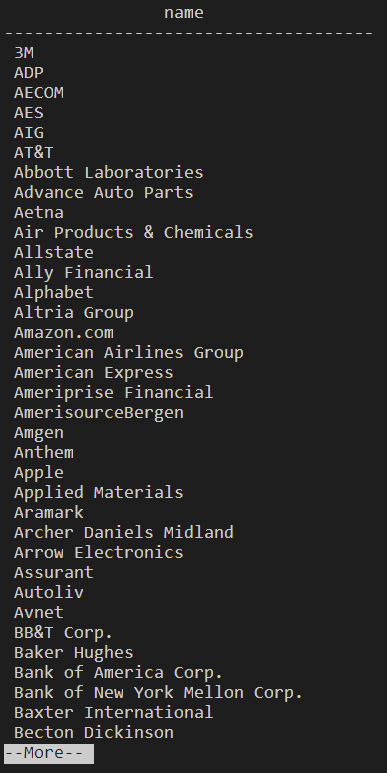






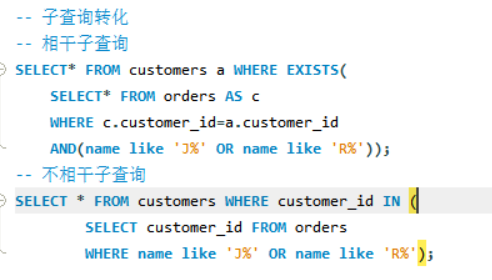
② 还有NOT IN等情况，比如下面的代码，用于查找 2017 年尚未下订单的所有客户。使用 IN 运算符的子查询通常返回零个或多个值的列表。子查询返回结果集后，外部查询使用它们。

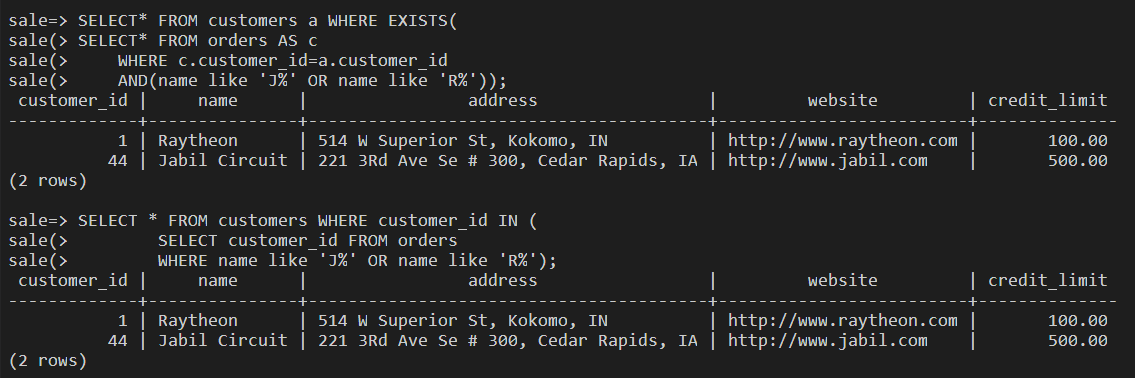




· 相关子查询与不相关子查询的区别？什么情形下使用相关子查询？如何将相关子查询转换成一般查询？（说明：一般查询指不一定必须使用子查询）

答：相关子查询是一个子查询，其某些子句引用外部查询中的列表达式。不相关子查询返回结果集后，外部查询使用它们。换句话说，外部查询依赖于子查询。但是，子查询是独立的，不依赖于外部查询的值。与不相关子查询不同，相关子查询是使用来自外部查询的值的子查询。可以为外部查询选择的每一行评估一次相关子查询。因此，使用相关子查询的查询可能会很慢。以下面代码为例，可将相干子查询转换成不相干子查询。





3.2 对实验的认识

通过实验我对openGauss中的一些语句更熟悉了。如

SET SEARCH\_PATH TO icebear, public;可以将搜索路径设置为icebear、public，首先搜索icebear。如SELECT \* FROM customer\_t1;可以用来查询表customer\_t1的所有数据。gsql -d sale -p 26000 -U yuxiaoping -W yuxiaoping@123 -r可以用来将新用户连接到数据库。可以使用gsql -d postgres -p 26000 -r 连接到postgres。gs\_om -t start可以开启数据库。

3.3 遇到的困难及解决方法

要更改当前会话的默认Schema，请使用SET命令。执行如下命令

SET SEARCH\_PATH To icebear,public;将搜索路径设置为myschema、public，首先搜索myschema。



高斯默认有session超时时间，若想要session一直保持，需要修改配置项：ALTER DATABASE sale SET session\_timeout TO 0;

