



数据库系统课程实验报告

实验名称:	数据高级查询
实验日期:	2022.11.11
实验地点:	四号楼
提交日期:	2022.11.14

学号:	22920202202877
姓名:	陈鑫蕾
专业年级:	数媒 2020 级
学年学期:	2022-2023 学年第一学期

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、INTERSECT、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

2. 实验内容和步骤

（1）连接到数据库

步骤如下：

1.在数据库主节点服务器上，切换至 omm 操作系统用户环境。

```
-bash: gsqt: command not found  
[root@ecs-7cda ~]# su - omm  
Last login: Tue Oct 18 10:19:04 CST 2022 on pts/0
```

2.查看服务是否启动。

```
[omm@ecs-7cda ~]$ gs_om -t status
```

```
-----  
-----  
cluster_name      : dbCluster  
cluster_state     : Normal  
redistributing    : No  
-----  
-----
```

3.启动数据库服务

```
[omm@ecs-7cda ~]$ gs_om -t start
```

```
Starting cluster.
```

```
=====  
[SUCCESS] ecs-7cda:  
[2022-10-18 12:26:33.753][14929][][gs_ctl]: gs_ctl started,data  
dir is /gaussdb/data/db1  
[2022-10-18 12:26:33.756][14929][][gs_ctl]: another server mig  
ht be running; Please use the restart command  
=====  
Successfully started.
```

4.连接数据库

```
[omm@ecs-7cda ~]$ gsql -d postgres -p 26000 -r  
gsql ((openGauss 2.0.0 build 78689da9) compiled at 2021-03-31 2  
1:03:52 commit 0 last mr )  
Non-SSL connection (SSL connection is recommended when requirin  
g high-security)  
Type "help" for help.
```

5. 使用自己创建的用户连接到此数据库

```
[omm@ecs-7cda ~]$ gsql -d sales -p 26000 -U chenxinlei -r  
Password for user chenxinlei:
```

(2) 查询

1. 创建两张表 palette_a 和 palette_b(结构相同,但表名不同,color 为颜色)

```

sales=> CREATE TABLE palette_a
sales-> ( id INT PRIMARY KEY,
sales(> color VARCHAR2 (100) NOT NULL);
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index
"palette_a_pkey" for table "palette_a"
CREATE TABLE

```

```

sales=> CREATE TABLE palette_b
sales-> ( id INT PRIMARY KEY,
sales(> color VARCHAR2 (100) NOT NULL);
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index
"palette_b_pkey" for table "palette_b"
CREATE TABLE

```

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

INSERT INTO palette_a

VALUES(1, 'Red'), (2, 'Green'), (3, 'Blue'), (4, 'Purple');

```

sales=> INSERT INTO palette_a
sales-> VALUES(1, 'Red'), (2, 'Green'), (3, 'Blue'), (4, 'Purple');

```

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

INSERT INTO palette_b

VALUES(1, 'Green'), (2, 'Red'), (3, 'Cyan'), (4, 'Brown');

```

sales=> INSERT INTO palette_b
sales-> VALUES(1, 'Green'), (2, 'Red'), (3, 'Cyan'), (4, 'Brown');
INSERT 0 4

```

4. 查询两张表中相同颜色的所有信息。

SELECT *

FROM palette_a,palette_b

WHERE palette_a.color=palette_b.color;

```

sales=> SELECT *
sales-> FROM palette_a,palette_b
sales-> WHERE palette_a.color=palette_b.color;
  id | color | id | color
-----+-----+-----+-----
   1 | Red   |  2 | Red
   2 | Green |  1 | Green
(2 rows)

```

5. 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）。

```

SELECT *
FROM palette_a
LEFT OUTER JOIN palette_b
ON palette_a.color=palette_b.color
WHERE palette_b.id IS NULL;

```

```

sales=> SELECT *
sales-> FROM palette_a
sales-> LEFT OUTER JOIN palette_b
sales-> ON palette_a.color=palette_b.color
sales-> WHERE palette_b.id IS NULL;
  id | color | id | color
-----+-----+-----+-----
   3 | Blue  |    | 
   4 | Purple|    | 
(2 rows)

```

6. 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）

```

SELECT *
FROM palette_a
RIGHT OUTER JOIN palette_b
ON palette_a.color=palette_b.color

```

WHERE palette_a.id IS NULL;

```
sales=> SELECT *
sales-> FROM palette_a
sales-> RIGHT OUTER JOIN palette_b
sales-> ON palette_a.color=palette_b.color
sales-> WHERE palette_a.id IS NULL;
 id | color | id | color
----+-----+----+-----
    |      | 3  | Cyan
    |      | 4  | Brown
(2 rows)
```

7. 查询 (5) 或 (6) 两种情况的信息 (用 (全) 外连接)。

FROM palette_a

FULL OUTER JOIN palette_b

ON palette_a.color=palette_b.color

WHERE palette_b.id IS NULL OR palette_a.id IS NULL;

```
sales=> SELECT *
sales-> FROM palette_a
sales-> FULL OUTER JOIN palette_b
sales-> ON palette_a.color=palette_b.color
sales-> WHERE palette_b.id IS NULL OR palette_a.id IS NULL;
 id | color | id | color
----+-----+----+-----
 3  | Blue  |    |
 4  | Purple|    |
    |      | 3  | Cyan
    |      | 4  | Brown
(4 rows)
```

8. 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

SELECT product_id, product_name, list_price

FROM products

WHERE list_price > (

SELECT AVG(list_price)

FROM products

);

```
sales=> SELECT product_id,product_name,list_price
sales-> FROM products
sales-> WHERE list_price >(
sales(>          SELECT AVG(list_price)
sales(>          FROM products
sales(>          );
```

product_id	product_name	list_price
228	Intel Xeon E5-2699 V3 (OEM/Tray)	3410.46
248	Intel Xeon E5-2697 V3	2774.98
249	Intel Xeon E5-2698 V3 (OEM/Tray)	2660.72
2	Intel Xeon E5-2697 V4	2554.99
45	Intel Xeon E5-2685 V3 (OEM/Tray)	2501.69
46	Intel Xeon E5-2695 V3 (OEM/Tray)	2431.95
47	Intel Xeon E5-2697 V2	2377.09
51	Intel Xeon E5-2695 V4	2269.99
91	Intel Xeon E5-2695 V2	2259.99
92	Intel Xeon E5-2643 V2 (OEM/Tray)	2200.00
93	Intel Xeon E5-2690 (OEM/Tray)	2116.72
98	Intel Xeon E5-2687W V3	2064.99
102	Intel Xeon E5-2687W V4	2042.69
158	Intel Xeon E5-2667 V3 (OEM/Tray)	2009.46
159	Intel Xeon E5-2690 V4	1994.49
160	Intel Xeon E5-2690 V3	1908.73
162	Intel Xeon E5-2470V2	1904.70
163	Intel Xeon E5-2683 V4	1899.99
164	Intel Xeon E5-2637 V2 (OEM/Tray)	1850.00
169	Intel Xeon E5-2683 V4 (OEM/Tray)	1844.89
240	Intel Core i7-4960X Extreme Edition	1805.97
241	Intel Xeon E5-2699 V4 (OEM/Tray)	1756.00
242	Intel Xeon E5-1680 V3 (OEM/Tray)	1751.99

--More--


```

184 | AMD 100-506061 | 999.99
181 | ATI FirePro R5000 | 999.99
182 | PNY VCQK4200-PB | 949.89
244 | Crucial | 1620.99
261 | G.Skill TridentZ RGB | 1504.99
262 | Corsair Dominator Platinum | 1449.99
265 | G.Skill Trident Z | 1431.99
266 | G.Skill Trident Z RGB | 1418.99
279 | G.Skill Ripjaws V Series | 1318.99
35 | Corsair Dominator Platinum | 1314.99
36 | Corsair Vengeance LPX | 1299.99
37 | Corsair Dominator Platinum | 1264.99
38 | Corsair Vengeance LPX | 1199.99
65 | Corsair Dominator Platinum | 1199.99
276 | Corsair Vengeance LPX | 1163.99
196 | Corsair Vengeance LPX | 1099.99
272 | G.Skill Ripjaws 4 Series | 1073.99
195 | G.Skill Ripjaws 4 Series | 1055.99
190 | Supermicro X10SDV-8C-TLN4F | 948.99
49 | Samsung MZ-75E4T0B | 1499.99
50 | Intel SSDPECM040T401 | 8867.99
208 | Samsung MZ-V6P2T0BW | 1199.99
(85 rows)

```

9. 查询产品表 products 中最便宜产品的 product_id, product_name, list_price

```
SELECT product_id, product_name, list_price
```

```
FROM products
```

```
WHERE list_price = (
```

```
SELECT MIN(list_price)
```

```
FROM products
```

```
);
```

```

sales=> SELECT product_id, product_name, list_price
sales-> FROM products
sales-> WHERE list_price = (
sales(> SELECT MIN(list_price)
sales(> FROM products
sales(> );
product_id | product_name | list_price
-----+-----+-----
94 | Western Digital WD2500AVVS | 15.55
(1 row)

```

10. 查询没有一个订单的顾客姓名。

```
SELECT name
```


FROM customers

WHERE customer_id NOT IN (

SELECT customer_id

FROM orders

);

```
sales=> SELECT name
sales-> FROM customers
sales-> WHERE customer_id NOT IN (
sales(>          SELECT customer_id
sales(>          FROM orders
sales(>          );
```

```
          name
-----
United Continental Holdings
INTL FCStone
Publix Super Markets
ConocoPhillips
3M
Exelon
Tesoro
Northwestern Mutual
Enterprise Products Partners
Rite Aid
Qualcomm
EMC
Time Warner Cable
Northrop Grumman
Lear
Genuine Parts
Omnicom Group
Monsanto
National Oilwell Varco
Marriott International
Kinder Morgan
Molina Healthcare
Lincoln National
```

```
Xcel Energy
State Farm Insurance Cos.
Johnson & Johnson
Marathon Petroleum
Aetna
AIG
Intel
Pfizer
FedEx
New York Life Insurance
Ingram Micro
American Airlines Group
Johnson Controls
Goldman Sachs Group
Oracle
(272 rows)
```

11. 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。

```
SELECT product_id, product_name, list_price
FROM products x
WHERE list_price > (
    SELECT AVG(list_price)
    FROM products y
    WHERE x.category_id=y.category_id
);
```

```

sales=> SELECT product_id,product_name,list_price
sales-> FROM products x
sales-> WHERE list_price > (
sales(>          SELECT AVG(list_price)
sales(>          FROM products y
sales(>          WHERE x.category_id=y.category_id
sales(>          );

```

product_id	product_name	list_price
228	Intel Xeon E5-2699 V3 (OEM/Tray)	3410.46
248	Intel Xeon E5-2697 V3	2774.98
249	Intel Xeon E5-2698 V3 (OEM/Tray)	2660.72
2	Intel Xeon E5-2697 V4	2554.99
45	Intel Xeon E5-2685 V3 (OEM/Tray)	2501.69
46	Intel Xeon E5-2695 V3 (OEM/Tray)	2431.95
47	Intel Xeon E5-2697 V2	2377.09
51	Intel Xeon E5-2695 V4	2269.99
91	Intel Xeon E5-2695 V2	2259.99
92	Intel Xeon E5-2643 V2 (OEM/Tray)	2200.00
93	Intel Xeon E5-2690 (OEM/Tray)	2116.72

195	Asus Z10PE-D16 WS	501.99
271	MSI X99A GODLIKE GAMING CARBON	549.59
64	Supermicro H8DG6-F	525.99
192	Asus Rampage V Edition 10	519.99
225	Gigabyte GA-Z270X-Gaming 9	503.98
252	Gigabyte X299 AORUS Gaming 9	499.99
274	ASRock E3C224D4M-16RE	499.99
61	Asus PRIME X299-DELUXE	487.30
188	Asus X99-E WS/USB 3.1	482.49
28	Supermicro X9SRH-7TF	479.99
137	Asus Z10PE-D16 WS	469.99
32	ASRock X99 Extreme11	469.99
31	Supermicro MBD-X10DAX	443.72
84	Asus X99-DELUXE/U3.1	440.30
141	Supermicro X11SSL-CF	419.99
34	Asus KGPE-D16	417.98
143	Asus Z10PE-D16	402.99
49	Samsung MZ-75E4T0B	1499.99
50	Intel SSDPECM040T401	8867.99
208	Samsung MZ-V6P2T0BW	1199.99

(132 rows)

12. 查询有订单 order 的所有顾客 customer 姓名（查询涉及 customers 表和 orders 表）

SELECT name

FROM customers

WHERE EXISTS (

SELECT *

FROM orders

WHERE customers.customer_id=orders.customer_id

);

```
sales=> SELECT name
sales-> FROM customers
sales-> WHERE EXISTS (
sales(>         SELECT *
sales(>         FROM orders
sales(>         WHERE customers.customer_id=orders.customer_id
sales(>         );
          name
```

```
-----
Facebook
Supervalu
NextEra Energy
PG&E Corp.
Goodyear Tire & Rubber
Micron Technology
ConAgra Foods
Bank of New York Mellon Corp.
Sempra Energy
Raytheon
Plains GP Holdings
US Foods Holding
AbbVie
Centene
```

```
AECOM
Jabil Circuit
CenturyLink
General Mills
Southern
Thermo Fisher Scientific
American Electric Power
NGL Energy Partners
Bristol-Myers Squibb
Nucor
PNC Financial Services Group
Health Net
Colgate-Palmolive
Freeport-McMoRan
Gap
Baker Hughes
Dollar Tree
Whole Foods Market
PPG Industries
DTE Energy
Becton Dickinson
AutoZone
(47 rows)
```

13. 执行以下两条语句，观察有何不同，能否得出某些初步结论

```

sales=> SELECT *
sales-> FROM customers
sales-> WHERE customer_id IN (NULL);
  customer_id | name | address | website | credit_limit
-----+-----+-----+-----+-----
(0 rows)

```

```

sales=> SELECT *
sales-> FROM customers
sales-> WHERE EXISTS (SELECT NULL FROM customers);
  customer_id | name | address | website | credit_limit
-----+-----+-----+-----+-----
177 | United Continental Holdings | 2984 S Salina St, Syracuse, NY | http://www.unitedcontinentalholdings.com | 5000.00
188 | INTL FCStone | 5344 Haverford Ave, Philadelphia, PA | http://www.intlfcstone.com | 5000.00
184 | Publix Super Markets | 1795 Wu Meng, Muang Chonburi, | http://www.publix.com | 1200.00
187 | ConocoPhillips | Walpurgisstr 69, Munich, | http://www.conocophillips.com | 2400.00
198 | 3M | Via Frenzy 6983, Roma, | http://www.3m.com |

```

第一条查询的是 id 为 null 的情况，但是 id 为主码，不可能为空，所以返回 0 条结果；第二条是判断 exists 后的情况，SELECT NULL FROM table 返回值为 1；

14. 找出所有没有订单的顾客姓名（查询涉及 customers 表和 orders 表）

SELECT name

FROM customer

WHERE NOT EXISTS(

SELECT *

FROM orders

WHERE

customers.customer_id=orders.customer_id

);

```

sales=> SELECT name
sales-> FROM customers
sales-> WHERE NOT EXISTS(
sales(>          SELECT *
sales(>          FROM orders
sales(>          WHERE customers.customer_id=orders.customer_id
sales(>          );
          name
-----
United Continental Holdings
INTL FCStone
Publix Super Markets
ConocoPhillips
3M
Exelon
Tesoro
Northwestern Mutual
Enterprise Products Partners
Rite Aid
Qualcomm
EMC
Time Warner Cable
Northrop Grumman
Lear
Genuine Parts
Omnicom Group

```

```

MGM Resorts International
Fidelity National Financial
Republic Services
Peter Kiewit Sons?
Mosaic
Core-Mark Holding
Cameron International
Xcel Energy
State Farm Insurance Cos.
Johnson & Johnson
Marathon Petroleum
Aetna
AIG
Intel
Pfizer
FedEx
New York Life Insurance
Ingram Micro
American Airlines Group
Johnson Controls
Goldman Sachs Group
Oracle
(272 rows)

```

15. 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。

SELECT product_name,list_price

FROM products

WHERE list_price > ANY (

SELECT list_price

FROM products

WHERE category_id=1

);

```
sales=> SELECT product_name,list_price
sales-> FROM products
sales-> WHERE list_price > ANY (
sales(>          SELECT list_price
sales(>          FROM products
sales(>          WHERE category_id=1
sales(>          );
```

product_name	list_price
Intel Xeon E5-2699 V3 (OEM/Tray)	3410.46
Intel Xeon E5-2697 V3	2774.98
Intel Xeon E5-2698 V3 (OEM/Tray)	2660.72
Intel Xeon E5-2697 V4	2554.99
Intel Xeon E5-2685 V3 (OEM/Tray)	2501.69
Intel Xeon E5-2695 V3 (OEM/Tray)	2431.95
Intel Xeon E5-2697 V2	2377.09
Intel Xeon E5-2695 V4	2269.99
Intel Xeon E5-2695 V2	2259.99
Intel Xeon E5-2643 V2 (OEM/Tray)	2200.00
Intel Xeon E5-2690 (OEM/Tray)	2116.72
Intel Xeon E5-2687W V3	2064.99
Intel Xeon E5-2687W V4	2042.69
Intel Xeon E5-2667 V3 (OEM/Tray)	2009.46
Intel Xeon E5-2690 V4	1994.49
Intel Xeon E5-2690 V3	1908.73
Intel Xeon E5-2470V2	1904.70

Kingston		644.00
G.Skill Ripjaws V Series		640.99
G.Skill Ripjaws V Series		640.99
Kingston HyperX Predator		635.99
Supermicro X10SDV-8C-TLN4F		948.99
Intel DP35DPM		789.79
Asus X99-E-10G WS		649.00
Asus ROG MAXIMUS IX EXTREME		573.99
Asus RAMPAGE V EXTREME		572.96
Asus Z10PE-D8 WS		561.59
Samsung MZ-75E4T0B		1499.99
Intel SSDPECM040T401		8867.99
Samsung MZ-V6P2T0BW		1199.99
Samsung MZ-V6P1T0BW		579.99

(188 rows)

16. 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

```
SELECT product_name,list_price
```

```
FROM products
```

```
WHERE list_price > ALL(
```

```
    SELECT list_price
```

```
    FROM products
```

```
    WHERE category_id=1
```

```
);
```

```

sales=> SELECT product_name,list_price
sales-> FROM products
sales-> WHERE list_price > ALL(
sales(>         SELECT list_price
sales(>         FROM products
sales(>         WHERE category_id=1
sales(>         );

```

product_name	list_price
PNY VCQP6000-PB	5499.99
PNY VCQM6000-24GB-PB	4139.00
Intel SSDPECM040T401	8867.99

(3 rows)

17. 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。

```

SELECT product_name,list_price
FROM products
WHERE list_price < ALL(
        SELECT AVG(list_price)
        FROM products
        GROUP BY category_id
);

```

```

sales=> SELECT product_name,list_price
sales-> FROM products
sales-> WHERE list_price < ALL(
sales(>          SELECT AVG(list_price)
sales(>          FROM products
sales(>          GROUP BY category_id
sales(>          );

```

product_name	list_price
ASRock C2750D4I	401.98
Gigabyte X299 AORUS Gaming 7	399.99
MSI X99A GODLIKE GAMING	399.99
MSI X299 GAMING M7 ACK	397.42
Supermicro MBD-X10DRI-O	394.99
Asus MAXIMUS IX FORMULA	388.99
Asus X99-DELUXE II	383.98
ASRock Fatal1ty X299 Professional Gaming i9	382.98
ASRock EP2C612 WS	358.49
ASRock Z270 SuperCarrier	353.98
Asus MAXIMUS VIII EXTREME/ASSEMBLY	353.98
Asus STRIX X299-E GAMING	349.99
Gigabyte X299 AORUS Ultra Gaming	343.99
Asus TUF X299 MARK 1	339.99
Asus Z170-WS	338.99
MSI X299 GAMING PRO CARBON AC	337.81
MSI X99A XPOWER GAMING TITANIUM	329.99
Seagate ST2000DX002	90.99
Western Digital WD5000AACS	26.99
SanDisk SDSSDHII-240G-G25	84.95
PNY SSD7CS1311-120-RB	57.98
PNY SSD9SC240GMDA-RB	80.72
Crucial CT1050MX300SSD1	267.99
Western Digital WDS250G1B0B	89.99
Samsung MZ-75E120B/AM	88.99
Western Digital WDS500G1B0B	149.88
Hitachi HUA723020ALA640	59.99
Western Digital WD2500AVVS	15.55
Seagate ST1000DX002	68.06
Crucial CT275MX300SSD1	97.88
Seagate ST31000340NS - FFP	43.99
Crucial CT525MX300SSD4	150.99
Western Digital WD2500AAJS	16.99
Western Digital WD20EZRZ	67.34
Western Digital WDS256G1X0C	109.99
SanDisk SDSSDHII-480G-G25	141.56
Kingston SV300S37A/120G	59.87

(81 rows)

18. 查询 contacts 表和 employees 表中的所有 last_name, 并以 last_name 升序显示。

```
SELECT last_name  
  
FROM(  
  
    SELECT last_name FROM contacts  
  
    UNION  
  
    SELECT last_name FROM employees  
  
)  
  
ORDER BY last_name ASC;
```

```
sales=> SELECT last_name  
sales-> FROM(  
sales(>    SELECT last_name FROM contacts  
sales(>    UNION  
sales(>    SELECT last_name FROM employees  
sales(> )  
sales-> ORDER BY last_name ASC;  
last_name  
-----  
Abbott  
Alexander  
Allison  
Alston  
Arnold  
Atkinson  
Avila  
Bailey  
Baldwin  
Ball  
Barnes
```

```
Wall  
Wallace  
Walters  
Walton  
Ward  
Warren  
Washington  
Watson  
Webb  
Weber  
Wells  
West  
Whitfield  
Wilkerson  
Wilkins  
Wilkinson  
Williamson  
Wilson  
Wolf  
Wong  
Wood  
Woods  
(357 rows)
```

19. 查询 contacts 表和 employees 表中的所有 last_name, 并以 last_name 升序显示

```
SELECT last_name  
  
FROM(  
  
    SELECT last_name FROM contacts  
  
    UNION ALL  
  
    SELECT last_name FROM employees  
  
)  
  
ORDER BY last_name ASC;
```

```

sales=> SELECT last_name
sales-> FROM(
sales(>    SELECT last_name FROM contacts
sales(>    UNION ALL
sales(>    SELECT last_name FROM employees
sales(> )
sales-> ORDER BY last_name ASC;
  last_name
-----
  Abbott
  Alexander
  Allison
  Alston
  Arnold
  Atkinson
  Atkinson
  Avila
  Bailey
  Baldwin
  Ball

```

```

  Washington
  Watson
  Webb
  Webb
  Weber
  Wells
  West
  West
  West
  Whitfield
  Wilkerson
  Wilkins
  Wilkinson
  Wilkinson
  Williamson
  Williamson
  Wilson
  Wolf
  Wong
  Wood
  Woods
  Woods
(426 rows)

```

20. 查询同时出现在 contacts 表和 employees 表中的所有 last_name。

SELECT last_name

FROM contacts

INTERSECT

SELECT last_name

FROM employees;

```
sales=> SELECT last_name
sales-> FROM contacts
sales-> INTERSECT
sales-> SELECT last_name
sales-> FROM employees;
  last_name
-----
  Stone
  Webb
  Henry
  Brooks
  Flores
  Cruz
  Mason
  Simmons
  Cole
  Murray
  Wallace
```



```
Robertson  
Ferguson  
Grant  
Henderson  
West  
Rose  
Ortiz  
Hayes  
Jordan  
Nichols  
Bryant  
Spencer  
Mcdonald  
(29 rows)
```

21. 查询在产品表 `products` 中而不在库存表 `inventories` 中的产品号 `product_id`。

```
SELECT product_id  
  
FROM products  
  
EXCEPT  
  
SELECT product_id  
  
FROM inventories;
```

```

sales=> SELECT product_id
sales-> FROM products
sales-> EXCEPT
sales-> SELECT product_id
sales-> FROM inventories;
 product_id
-----
        158
         86
         77
         97
         16
        118
        178
        153
        111

```

```

        249
        171
        169
         82
        226
        208
         55
         53
        177
          1
        215
        127
         45
        266
        250
(80 rows)

```

(3) 思考：

- 什么类型的查询只能用子查询实现？试举例说明。

当查询的数据不能直接获取时，比如查询没有一个订单的顾客姓名

- 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的查询）

不相关子查询中子查询的条件不依赖于父查询；相关子查询中的子查询依赖父查询；

相关子查询中的每查询一条记录，需要重新做一次子查询；非相关子查询中子查询只需要进行一次；

相关子查询可以嵌套多层，但是嵌套层越多，效率越低；非相关子查询不可以嵌套，效率较高；

当子查询需要用到父查询的信息的时候就要用到相关自查询；

3. 实验总结

3.1 完成的工作

打开数据库

输入 SQL 语句

3.2 对实验的认识

掌握了复杂的查询，懂得了连接查询和子查询

3.3 遇到的困难及解决方法

无