数据库 查询与修改实验 实验报告

陈俊卉 2020212256 2020219111班

刘 帅 2020212267 2020219111班

李沅昕 2020212295 2020219112班

一、实验环境

- 数据库：openGauss1.1.0/openGauss2.0.0 数据库和华为云GaussDB(openGauss)数据库
- 实验数据：电商数据库的八张表

二、实验目的

- 对前面实验建立的电商数据库关系表进行各种类型的查询操作和修改操作，加深对 SQL 语言中DML的了解，掌握相关查询语句和数据修改语句的使用方法。

三、实验内容

1. 单表简单查询，包括复合选择条件、结果排序、结果去重、结果重命名查询；
2. 多表查询，包括等值连接、自然连接、元组变量查询；
3. 统计查询，包括带有分组、聚集函数的查询；
4. 嵌套查询，包括带有 **in/some/all、 exists、unique** 的嵌套查询，from 中子查询；
5. with 临时视图查询；
6. 键/函数依赖分析；
7. 表的插入、删除、更新；

四、实验步骤与过程、结果

查询1

```sql
select o_orderkey,o_custkey,o_orderstatus,o_totalprice,o_orderdate as o_date,o_orderpriority ,o_shippriority  from orders
where o_totalprice BETWEEN 5000 and 10000
and o_clerk = 'Clerk#000000951'
and o_orderdate between '2015-12-02 00:00:00'::date and '2019-12-02 00:00:00'::date
and o_orderstatus IS NOT NULL
ORDER BY o_orderpriority desc, o_shippriority desc
```

SQL执行记录 消息 结果集1

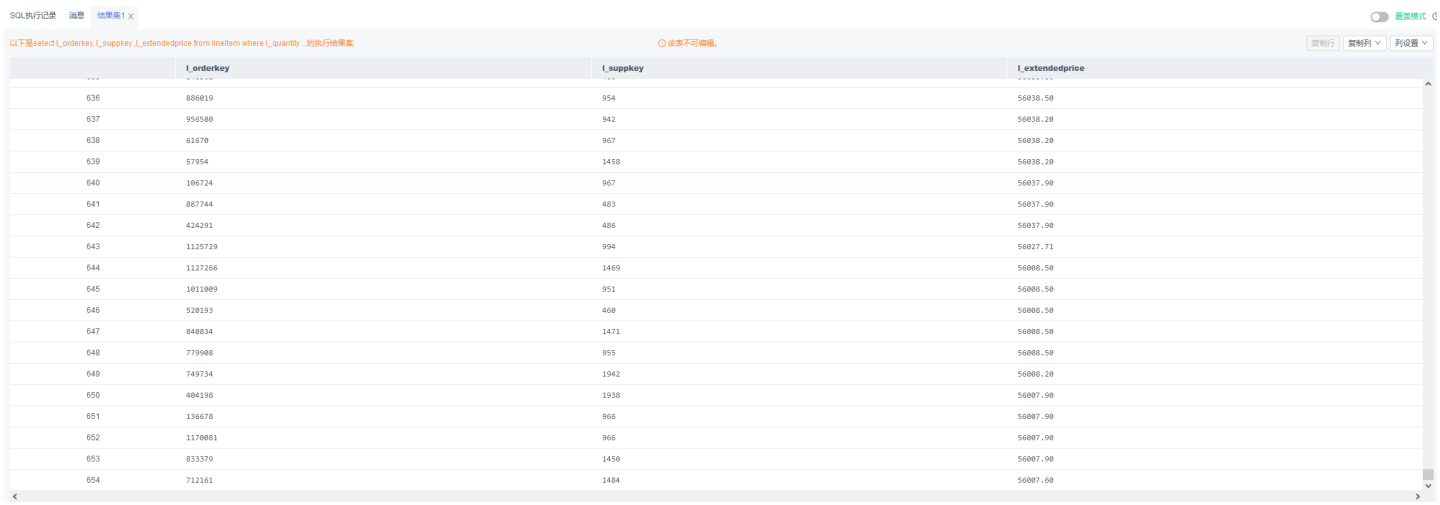
执行时间	SQL语句	消耗时间	执行结果
2022-11-13 18:35:51	select o_orderkey,o_custkey,o_orderstatus,o_totalprice,o_orderdate as...	134 ms	执行成功

查询2

不去重：

SQL执行记录 消息 结果集1

执行时间	SQL语句	消耗时间	执行结果
2022-11-13 18:27:18	select l_orderkey, l_suppkey ,l_extendedprice from lineitem where l_quantity...	523 ms	执行成功



l_orderkey	l_suppkey	l_extendedprice
636	886019	56038.50
637	956580	56038.20
638	61670	56038.20
639	57954	56038.20
640	106724	56037.90
641	887744	56037.90
642	424291	56037.90
643	1125729	56027.71
644	1127266	56008.50
645	1011009	56008.50
646	520193	56008.50
647	840834	56008.50
648	779908	56008.50
649	749734	56008.20
650	404198	56007.90
651	136678	56007.90
652	1170081	56007.90
653	833379	56007.90
654	712161	56007.60

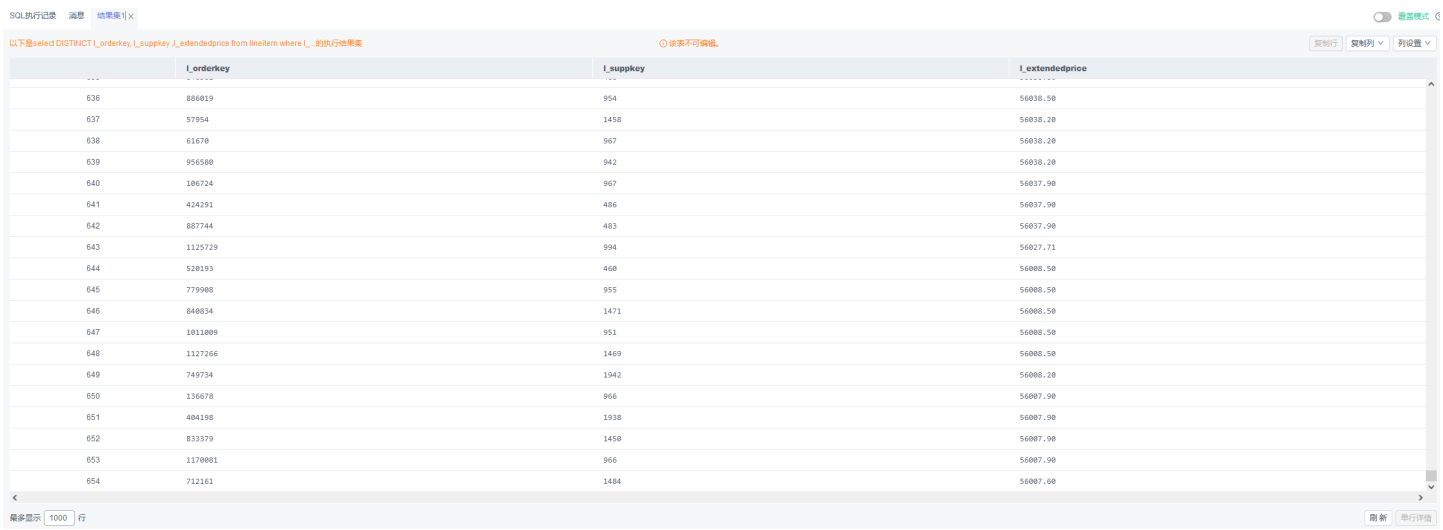
```sql
select l_orderkey, l_suppkey ,l_extendedprice from lineitem
where l_quantity BETWEEN 10 and 30
and l_returnflag = 'N'
and l_extendedprice >= 56000
```

```
5 ORDER BY l_extendedprice desc
```

去重：没有重复的 所以没有变化

SQL执行记录 消息 结果集1

执行时间	SQL语句	消耗时间	执行结果
2022-11-13 18:27:52	select DISTINCT l_orderkey, l_suppkey ,l_extendedprice from lineitem where...	540 ms	执行成功

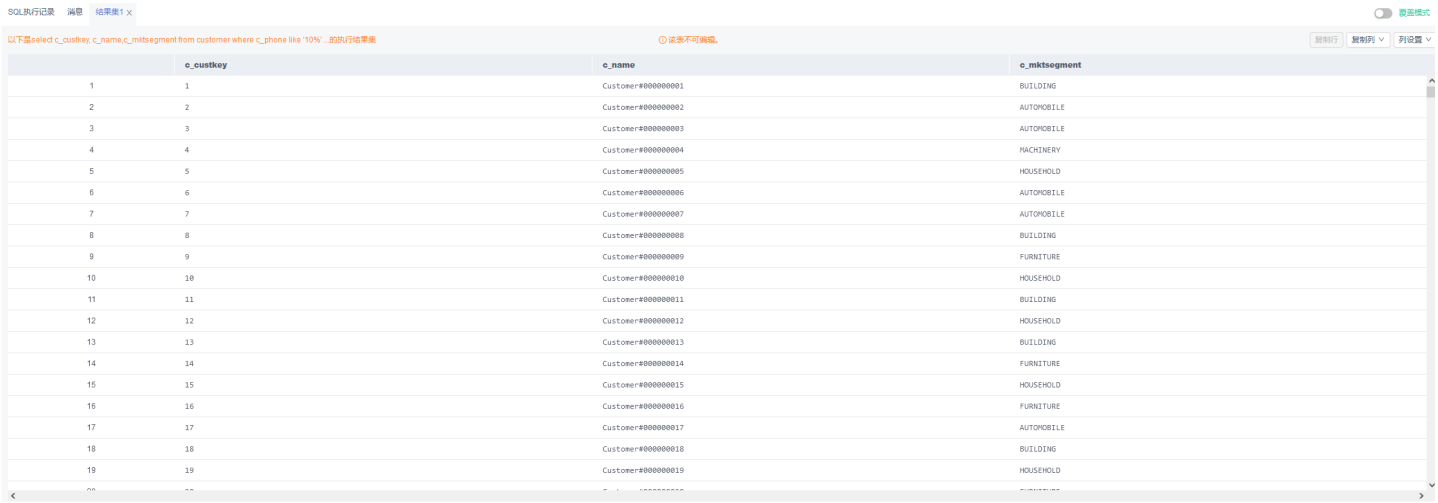


l_orderkey	l_suppkey	l_extendedprice
636	886019	954	56038.50
637	57954	1458	56038.20
638	61670	967	56038.20
639	956580	942	56038.20
640	106724	967	56037.90
641	424291	486	56037.90
642	887744	483	56037.90
643	1125729	994	56027.71
644	520193	460	56008.50
645	779908	955	56008.50
646	840834	1471	56008.50
647	1011009	951	56008.50
648	1127266	1469	56008.50
649	749734	1942	56008.20
650	136678	966	56007.90
651	404198	1938	56007.90
652	833379	1450	56007.90
653	1170081	966	56007.90
654	712161	1484	56007.60

```
1 select distinct l_orderkey, l_suppkey ,l_extendedprice from lineitem
2 where l_quantity BETWEEN 10 and 70
3 and l_returnflag = 'N'
4 and l_extendedprice >= 10000
5 ORDER BY l_extendedprice desc
```

查询3

```
1 select c_custkey, c_name,c_mktsegment from customer
2 where c_phone like '10%'
3 or c_mktsegment  like 'BUILDING'
4 and c_phone NOT LIKE '%8'
```



	c_custkey	c_name	c_mktsegment
1	1	Customer#000000001	BUILDING
2	2	Customer#000000002	AUTOMOBILE
3	3	Customer#000000003	AUTOMOBILE
4	4	Customer#000000004	MACHINERY
5	5	Customer#000000005	HOUSEHOLD
6	6	Customer#000000006	AUTOMOBILE
7	7	Customer#000000007	AUTOMOBILE
8	8	Customer#000000008	BUILDING
9	9	Customer#000000009	FURNITURE
10	10	Customer#000000010	HOUSEHOLD
11	11	Customer#000000011	BUILDING
12	12	Customer#000000012	HOUSEHOLD
13	13	Customer#000000013	BUILDING
14	14	Customer#000000014	FURNITURE
15	15	Customer#000000015	HOUSEHOLD
16	16	Customer#000000016	FURNITURE
17	17	Customer#000000017	AUTOMOBILE
18	18	Customer#000000018	BUILDING
19	19	Customer#000000019	HOUSEHOLD

SQL执行记录 消息 结果集1 ×

执行时间	SQL语句	消耗时间	执行结果
2022-11-13 18:37:22	select c_custkey, c_name,c_mktsegment from customer where c_phone like '10%' ...	30 ms	执行成功

查询4

```
select c_name from customer
where c_custkey like '__'
and length(c_address) >= 18
```

SQL执行记录 消息 结果集1 ×

以下是select c_name from customer where c_custkey like '__' and length(c_address) >...的执行结果集

	c_name
1	Customer#000000010
2	Customer#000000012
3	Customer#000000013
4	Customer#000000014
5	Customer#000000015
6	Customer#000000017
7	Customer#000000018
8	Customer#000000019
9	Customer#000000020
10	Customer#000000022
11	Customer#000000023
12	Customer#000000024
13	Customer#000000025
14	Customer#000000027
15	Customer#000000028
16	Customer#000000030
17	Customer#000000031
18	Customer#000000033
19	Customer#000000035

SQL执行记录 消息 结果集1 ×

执行时间	SQL语句	消耗时间	执行结果
2022-11-13 18:45:06	select c_name from customer where c_custkey like '__' and length(c_address) >= 18	41 ms	执行成功

查询5

Union all:

```
1 select l_orderkey FROM lineitem
2 WHERE l_shipdate < '2016-01-01'::date
3 union all
4 select l_orderkey FROM lineitem
5 where l_quantity > 100
```

SQL执行记录 消息 结果集1 ×

以下是select l_orderkey FROM lineitem WHERE l_shipdate < '2016-01-01'::date union a...的执行结果集

	l_orderkey
1	6
2	37
3	37
4	129
5	129
6	129
7	129
8	130
9	130
10	130
11	130
12	130
13	134
14	134
15	134
16	134
17	134
18	134
19	164

129
复制

SQL执行记录 消息 结果集1 ×

执行时间	SQL语句	消耗时间	执行结果
2022-11-13 18:51:17	select l_orderkey FROM lineitem WHERE l_shipdate < '2016-01-01'::date union all...	287 ms	执行成功

Union:

```
select l_orderkey FROM lineitem
WHERE l_shipdate < '2016-01-01'::date
union
select l_orderkey FROM lineitem
where l_quantity > 100
```

以下是select l_orderkey FROM lineitem WHERE l_shipdate < '2016-01-01'::date union s...的执行结果集

	l_orderkey
1	865029
2	370596
3	310753
4	363111
5	1096454
6	728802
7	586087
8	186371
9	1190499
10	615072
11	1196007
12	701091
13	277216
14	134595
15	50468
16	868391
17	592389
18	470022
19	614980

SQL执行记录 消息 结果集1 ×

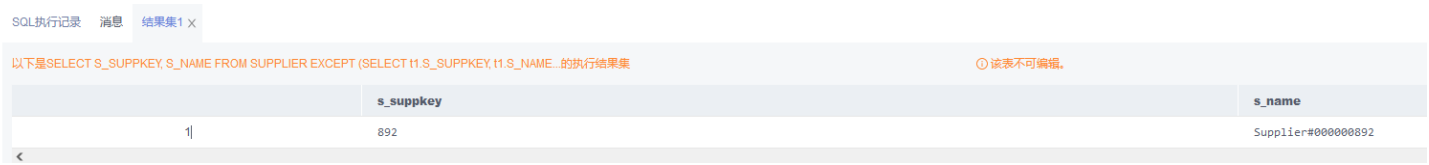
执行时间	SQL语句	消耗时间	执行结果
2022-11-13 18:53:15	select l_orderkey FROM lineitem WHERE l_shipdate < '2016-01-01'::date union sel...	895 ms	执行成功

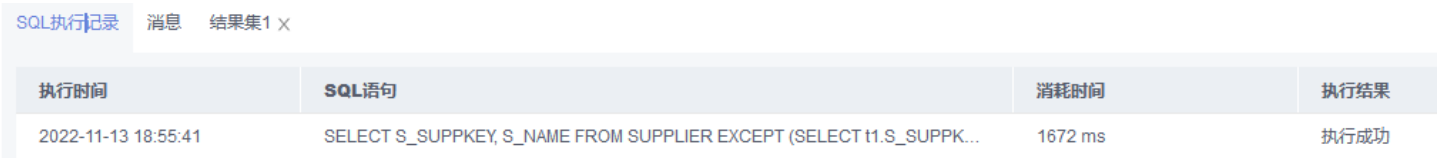
查询6

Except

```
1 SELECT S_SUPPKEY, S_NAME
2 FROM SUPPLIER
3 EXCEPT
4 (SELECT t1.S_SUPPKEY, t1.S_NAME
```

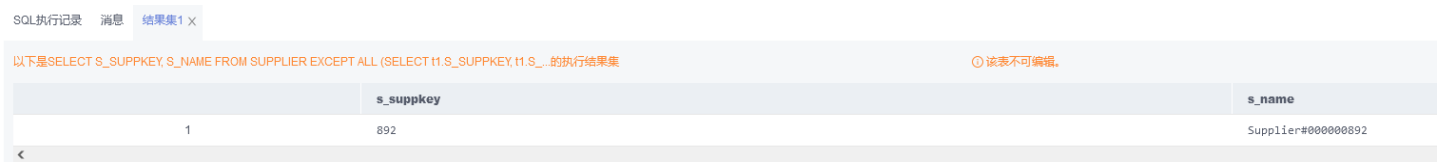
```
5 FROM SUPPLIER t1, SUPPLIER t2
6 WHERE t1.S_ACCTBAL < t2.S_ACCTBAL);
```

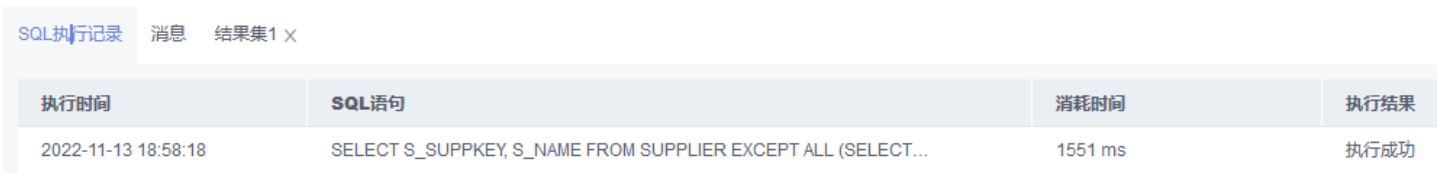




Except all （计算前不删除重复）

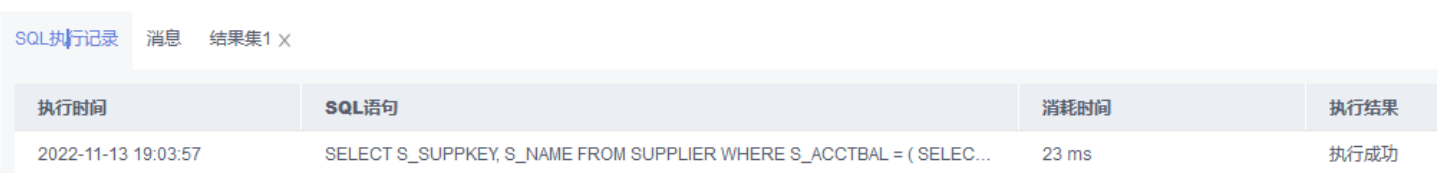
```
1 SELECT S_SUPPKEY, S_NAME
2 FROM SUPPLIER
3 EXCEPT ALL
4 (SELECT t1.S_SUPPKEY, t1.S_NAME
5 FROM SUPPLIER t1, SUPPLIER t2
6 WHERE t1.S_ACCTBAL < t2.S_ACCTBAL);
```





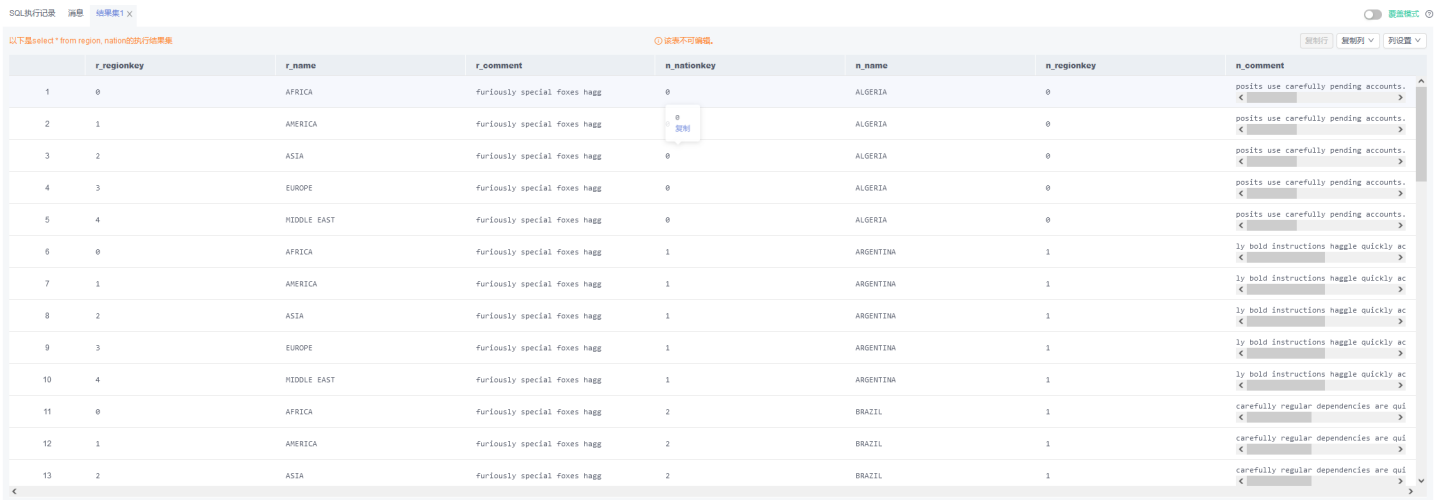
Max

```
1
```

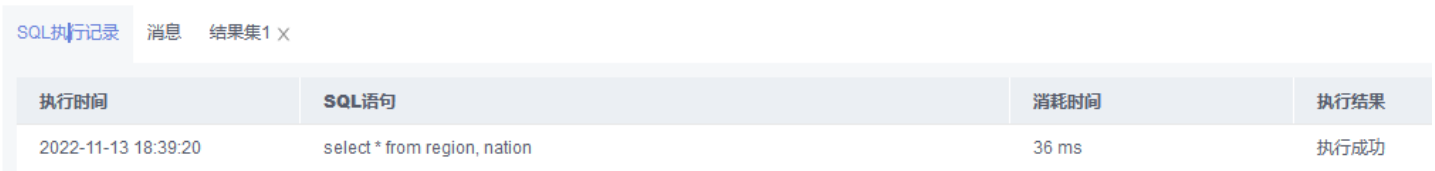


查询7

```
select *
from region, nation
```



r_regionkey	r_name	r_comment	n_nationkey	n_name	n_regionkey	n_comment
0	AFRICA	furiously special foxes hagg	0	ALGERIA	0	posits use carefully pending accounts.
1	AMERICA	furiously special foxes hagg	0	ALGERIA	0	posits use carefully pending accounts.
2	ASIA	furiously special foxes hagg	0	ALGERIA	0	posits use carefully pending accounts.
3	EUROPE	furiously special foxes hagg	0	ALGERIA	0	posits use carefully pending accounts.
4	MIDDLE EAST	furiously special foxes hagg	0	ALGERIA	0	posits use carefully pending accounts.
0	AFRICA	furiously special foxes hagg	1	ARGENTINA	1	ly bold instructions haggle quickly ac
1	AMERICA	furiously special foxes hagg	1	ARGENTINA	1	ly bold instructions haggle quickly ac
2	ASIA	furiously special foxes hagg	1	ARGENTINA	1	ly bold instructions haggle quickly ac
3	EUROPE	furiously special foxes hagg	1	ARGENTINA	1	ly bold instructions haggle quickly ac
4	MIDDLE EAST	furiously special foxes hagg	1	ARGENTINA	1	ly bold instructions haggle quickly ac
0	AFRICA	furiously special foxes hagg	2	BRAZIL	1	carefully regular dependencies are qui
1	AMERICA	furiously special foxes hagg	2	BRAZIL	1	carefully regular dependencies are qui
2	ASIA	furiously special foxes hagg	2	BRAZIL	1	carefully regular dependencies are qui

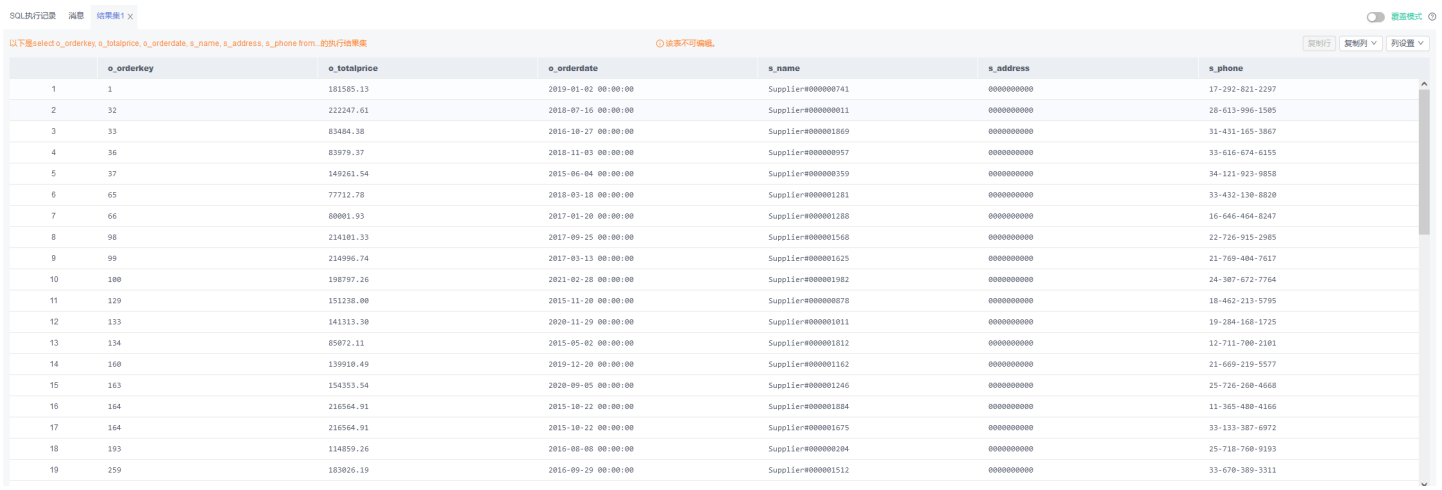


SQL执行记录 消息 结果集1

执行时间	SQL语句	消耗时间	执行结果
2022-11-13 18:39:20	select * from region, nation	36 ms	执行成功

查询8

```
select o_orderkey, o_totalprice, o_orderdate, s_name, s_address, s_phone
from lineitem join orders on l_orderkey = o_orderkey , supplier
WHERE supplier.s_suppkey = lineitem.l_suppkey
and lineitem.l_receiptdate < lineitem.l_commitdate
```



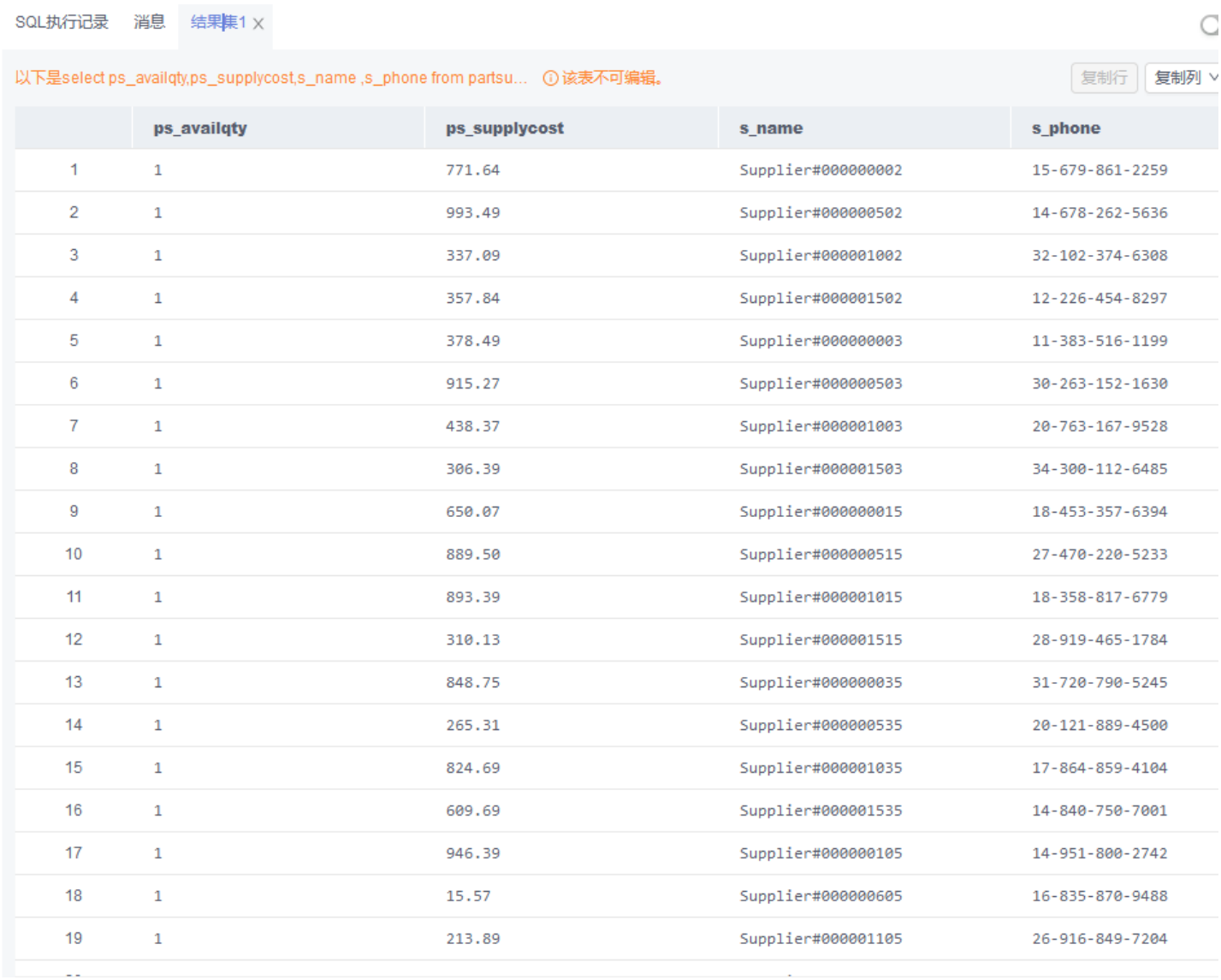
	o_orderkey	o_totalprice	o_orderdate	s_name	s_address	s_phone
1	1	181585.13	2019-01-02 00:00:00	Supplier#000000741	0000000000	17-292-821-2297
2	32	222247.61	2018-07-16 00:00:00	Supplier#000000011	0000000000	28-613-996-1505
3	33	83484.38	2016-10-27 00:00:00	Supplier#000001869	0000000000	31-431-165-3867
4	36	83979.37	2018-11-03 00:00:00	Supplier#000000957	0000000000	33-616-674-6155
5	37	149261.54	2015-06-04 00:00:00	Supplier#000000359	0000000000	34-121-923-9858
6	65	77712.78	2018-03-18 00:00:00	Supplier#000001281	0000000000	33-432-130-8820
7	66	80001.93	2017-01-20 00:00:00	Supplier#000001288	0000000000	16-646-464-8247
8	98	214101.33	2017-09-25 00:00:00	Supplier#000001568	0000000000	22-726-915-2985
9	99	214996.74	2017-03-13 00:00:00	Supplier#000001625	0000000000	21-769-404-7617
10	100	198797.26	2021-02-28 00:00:00	Supplier#000001982	0000000000	24-307-672-7764
11	129	151238.00	2015-11-20 00:00:00	Supplier#000000878	0000000000	18-462-213-5795
12	133	141313.30	2020-11-29 00:00:00	Supplier#000001011	0000000000	19-284-168-1725
13	134	85072.11	2015-05-02 00:00:00	Supplier#000001812	0000000000	12-711-700-2101
14	160	139910.49	2019-12-20 00:00:00	Supplier#000001162	0000000000	21-669-219-5577
15	163	154353.54	2020-09-05 00:00:00	Supplier#000001246	0000000000	25-726-260-4668
16	164	216564.91	2015-10-22 00:00:00	Supplier#000001884	0000000000	11-365-480-4166
17	164	216564.91	2015-10-22 00:00:00	Supplier#000001675	0000000000	33-133-387-6972
18	193	114859.26	2016-08-08 00:00:00	Supplier#000000204	0000000000	25-718-760-9193
19	259	183026.19	2016-09-29 00:00:00	Supplier#000001512	0000000000	33-670-389-3311

SQL执行记录　消息　结果集1 ×

执行时间	SQL语句	消耗时间	执行结果
2022-11-13 21:00:48	select o_orderkey, o_totalprice, o_orderdate, s_name, s_address, s_phone from...	1887 ms	执行成功

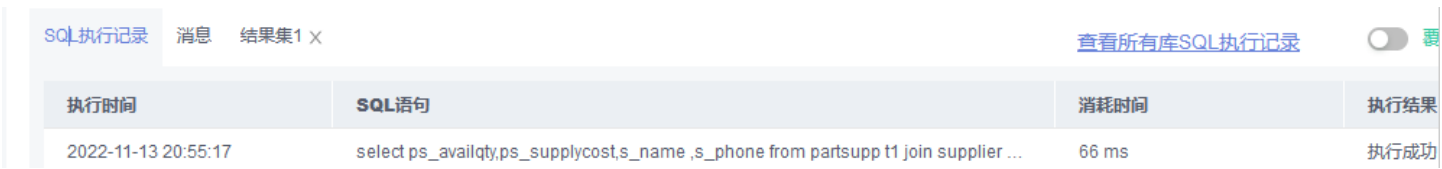
查询9

```
select ps_availqty,ps_supplycost,s_name ,s_phone
from partsupp t1 join supplier  on t1.ps_suppkey = supplier.s_suppkey, part
where t1.ps_partkey = p_partkey
and part.p_brand = 'Brand#13'
```


SQL执行记录 消息 结果集1 ×

以下是select ps_availqty,ps_supplycost,s_name ,s_phone from partsu... ⓘ该表不可编辑。 复制行 复制列

	ps_availqty	ps_supplycost	s_name	s_phone
1	1	771.64	Supplier#000000002	15-679-861-2259
2	1	993.49	Supplier#000000502	14-678-262-5636
3	1	337.09	Supplier#000001002	32-102-374-6308
4	1	357.84	Supplier#000001502	12-226-454-8297
5	1	378.49	Supplier#000000003	11-383-516-1199
6	1	915.27	Supplier#000000503	30-263-152-1630
7	1	438.37	Supplier#000001003	20-763-167-9528
8	1	306.39	Supplier#000001503	34-300-112-6485
9	1	650.07	Supplier#000000015	18-453-357-6394
10	1	889.50	Supplier#000000515	27-470-220-5233
11	1	893.39	Supplier#000001015	18-358-817-6779
12	1	310.13	Supplier#000001515	28-919-465-1784
13	1	848.75	Supplier#000000035	31-720-790-5245
14	1	265.31	Supplier#000000535	20-121-889-4500
15	1	824.69	Supplier#000001035	17-864-859-4104
16	1	609.69	Supplier#000001535	14-840-750-7001
17	1	946.39	Supplier#000000105	14-951-800-2742
18	1	15.57	Supplier#000000605	16-835-870-9488
19	1	213.89	Supplier#000001105	26-916-849-7204


SQL执行记录 消息 结果集1 × 查看所有库SQL执行记录

执行时间	SQL语句	消耗时间	执行结果
2022-11-13 20:55:17	select ps_availqty,ps_supplycost,s_name ,s_phone from partsupp t1 join supplier ...	66 ms	执行成功

查询10

```
select l_orderkey, l_linenumber as number, l_partkey, l_discount
from lineitem
where l_discount > (
  select l_discount
  from lineitem
  where l_linenumber = '1'
  and l_orderkey = '1'
  )
order by l_discount desc
```

SQL执行记录 消息 结果集1 ×

以下是select l_orderkey, l_linenumber as number, l_partkey, l_discount... ⓘ该表不可编辑。 复制行 复制列

	l_orderkey	number	l_partkey	l_discount
1	85762	2	5380	.10
2	85732	4	27569	.10
3	42624	5	359	.10
4	42624	6	19238	.10
5	42624	7	8771	.10
6	170368	1	39971	.10
7	6852	1	27242	.10
8	141893	1	13941	.10
9	128996	6	34897	.10
10	226758	4	32582	.10
11	159554	5	25772	.10
12	42626	5	2184	.10
13	42626	6	16965	.10
14	85731	4	14283	.10
15	239649	4	20964	.10
16	197250	7	1564	.10
17	226757	1	4463	.10
18	85731	2	6707	.10
19	122471	3	22488	.10

SQL执行记录 消息 结果集1 × 查看所有库SQL执行记录

执行时间	SQL语句	消耗时间	执行结果
2022-11-13 21:43:45	select l_orderkey, l_linenumber as number, l_partkey, l_discount from lineitem wher...	1336 ms	执行成功

查询11

(1)

```sql
select max(l_quantity), min(l_quantity), avg(l_quantity)
from lineitem, customer, nation, orders
where lineitem.l_orderkey = orders.o_orderkey
and customer.c_nationkey = nation.n_nationkey
and orders.o_custkey = customer.c_custkey
and nation.n_name = 'ALGERIA'
and orders.o_orderdate between '2015-01-01'::date and '2015-02-02'::date
```

	max	min	avg
1	50.00	1.00	25.4716969806110811

SQL执行记录 消息 结果集1 ×

查看所有库

执行时间	SQL语句	消耗时间
2022-11-13 22:05:18	select max(l_quantity), min(l_quantity), avg(l_quantity) from lineitem, customer, natio...	459 ms

(2)

```
select l_quantity, l_shipdate, o_orderdate
from lineitem , customer, nation, orders
where lineitem.l_orderkey = orders.o_orderkey
and customer.c_nationkey = nation.n_nationkey
and orders.o_custkey = customer.c_custkey
and nation.n_name = 'ALGERIA'
and orders.o_orderdate between '2015-01-01'::date and '2015-02-02'::date
order by l_quantity desc limit 1
```

	l_quantity	l_shipdate	o_orderdate
1	50.00	2015-01-20 00:00:00	2015-01-14 00:00:00

SQL执行记录 消息 结果集1 ×

查看所有库SQL执行记录

执行时间	SQL语句	消耗时间	执行结果
2022-11-13 22:10:08	select l_quantity, l_shipdate, o_orderdate from lineitem , customer, nation, orders...	708 ms	执行成功

查询12

```
select p_mfgr, p_type, p_retailprice, count(s_suppkey) as supplier_num
from part, partsupp, supplier
where part.p_partkey = partsupp.ps_partkey
and supplier.s_suppkey = partsupp.ps_suppkey
and part.p_brand = 'Brand#13'
group by p_mfgr, p_type, p_retailprice
order by p_retailprice desc
```

	p_mfgr	p_type	p_retailprice	supplier_num
1	Manufacturer#1	STANDARD ANODIZED TIN	1932.99	8
2	Manufacturer#1	STANDARD ANODIZED TIN	1930.99	4
3	Manufacturer#1	STANDARD ANODIZED TIN	1929.99	4
4	Manufacturer#1	STANDARD ANODIZED TIN	1923.99	8
5	Manufacturer#1	STANDARD ANODIZED TIN	1922.99	4
6	Manufacturer#1	STANDARD ANODIZED TIN	1922.98	4
7	Manufacturer#1	STANDARD ANODIZED TIN	1921.99	4
8	Manufacturer#1	STANDARD ANODIZED TIN	1921.98	4
9	Manufacturer#1	STANDARD ANODIZED TIN	1920.99	4
10	Manufacturer#1	STANDARD ANODIZED TIN	1920.98	4
11	Manufacturer#1	STANDARD ANODIZED TIN	1917.98	4
12	Manufacturer#1	STANDARD ANODIZED TIN	1916.98	4
13	Manufacturer#1	STANDARD ANODIZED TIN	1916.97	4
14	Manufacturer#1	STANDARD ANODIZED TIN	1915.99	4
15	Manufacturer#1	STANDARD ANODIZED TIN	1914.98	4
16	Manufacturer#1	STANDARD ANODIZED TIN	1913.98	4
17	Manufacturer#1	STANDARD ANODIZED TIN	1911.98	4
18	Manufacturer#1	STANDARD ANODIZED TIN	1908.97	4
19	Manufacturer#1	STANDARD ANODIZED TIN	1905.99	4

SQL执行记录 消息 结果集1 ×

查看所有历

执行时间	SQL语句	消耗时间
2022-11-13 22:33:05	select p_mfgr, p_type, p_retailprice, count(s_suppkey) as supplier_num from part,...	141 ms

查询13

```sql
select p_partkey, ps_supplycost, avg(p_retailprice) as avg_retail_price
from part,partsupp
where part.p_partkey = partsupp.ps_partkey
and part.p_size between 7 and 14
group by  p_partkey, ps_supplycost
order by avg_retail_price desc
```

SQL执行记录 消息 结果集1 ×

以下是select p_partkey, ps_supplycost, avg(p_retailprice) as avg_retail... ⓘ该表不可编辑。 复制行 复制

	p_partkey	ps_supplycost	avg_retail_price
1	39998	254.68	1937.9900000000000000
2	39998	711.23	1937.9900000000000000
3	39998	755.19	1937.9900000000000000
4	39998	880.04	1937.9900000000000000
5	35999	215.76	1934.9900000000000000
6	35999	154.96	1934.9900000000000000
7	35999	562.69	1934.9900000000000000
8	35999	288.24	1934.9900000000000000
9	33999	829.38	1932.9900000000000000
10	36996	203.32	1932.9900000000000000
11	33999	492.21	1932.9900000000000000
12	36996	774.61	1932.9900000000000000
13	33999	846.45	1932.9900000000000000
14	36996	109.99	1932.9900000000000000
15	33999	713.85	1932.9900000000000000
16	36996	750.02	1932.9900000000000000
17	35996	623.43	1931.9900000000000000
18	35996	576.17	1931.9900000000000000
19	35996	835.87	1931.9900000000000000

SQL执行记录 消息 结果集1 × 查看所有库S

执行时间	SQL语句	消耗时间
2022-11-13 22:37:07	select p_partkey, ps_supplycost, avg(p_retailprice) as avg_retail_price from...	168 ms

查询14

(1)嵌套

```
select o_orderkey, c_name
from orders, customer
where o_custkey = c_custkey
and o_orderkey in (
  select l_orderkey
  from lineitem
  where l_discount < 0.01)
```

SQL执行记录 消息 结果集1 ×

以下是select o_orderkey, c_name from orders, customer where o_cust... ⓘ该表不可编辑。

	o_orderkey	c_name
1	2	Customer#000015601
2	34	Customer#000012202
3	65	Customer#000003251
4	66	Customer#000025840
5	71	Customer#000000676
6	97	Customer#000004213
7	98	Customer#000020896
8	100	Customer#000029401
9	101	Customer#000005600
10	129	Customer#000014227
11	132	Customer#000005279
12	133	Customer#000008800
13	134	Customer#000001240
14	135	Customer#000012097
15	160	Customer#000016499
16	163	Customer#000017552
17	165	Customer#000005449
18	192	Customer#000016514
19	194	Customer#000012346

SQL执行记录 消息 结果集1 ×

查看所有库

执行时间	SQL语句	消耗时间
2022-11-13 22:42:17	select o_orderkey, c_name from orders, customer where o_custkey = c_custkey an...	880 ms

(2)多表

```
select distinct l_orderkey, c_name
from lineitem, orders, customer
where o_custkey = c_custkey
and l_orderkey = o_orderkey
and l_discount < 0.01
```

SQL执行记录 消息 结果集1 ×

以下是select distinct l_orderkey, c_name from lineitem, orders, custom... ⓘ该表不可编辑。

	l_orderkey	c_name
1	674469	Customer#000002546
2	835143	Customer#000003364
3	964549	Customer#000026950
4	1049190	Customer#000003877
5	223141	Customer#000025421
6	1124481	Customer#000003056
7	168228	Customer#000016942
8	187460	Customer#000005974
9	429351	Customer#000004274
10	25766	Customer#000011998
11	309155	Customer#000021184
12	1011879	Customer#000006592
13	171812	Customer#000025333
14	303138	Customer#000015259
15	209732	Customer#000008218
16	338528	Customer#000028264
17	1127303	Customer#000029626
18	571138	Customer#000002926
19	233511	Customer#000019312

SQL执行记录 消息 结果集1 ×

查看所有库

执行时间	SQL语句	消耗时间
2022-11-13 22:49:09	select distinct l_orderkey, c_name from lineitem, orders, customer where o_custkey...	983 ms

查询15-1

```
select l_linenumber, l_orderkey, l_tax
from lineitem
where l_quantity > some(
  select l_quantity
  from lineitem
```

```
6   where l_shipdate > '2015-01-01'::date
7   and l_shipdate < '2015-02-02'::date
8   );
```

SQL执行记录 消息 结果集1 ×

以下是select l_linenumber, l_orderkey, l_tax from lineitem where l_qua... ⓘ该表不可编辑。

	l_linenumber	l_orderkey	l_tax
1	1	1	.02
2	2	1	.06
3	3	1	.02
4	4	1	.06
5	5	1	.04
6	6	1	.02
7	1	2	.08
8	1	3	.05
9	2	3	.01
10	3	3	.03
11	4	3	.05
12	5	3	.01
13	6	3	.05
14	1	4	.07
15	1	5	0.00
16	2	5	0.00
17	3	5	.07
18	1	6	.06
19	1	7	0.00

SQL执行记录 消息 结果集1 ×

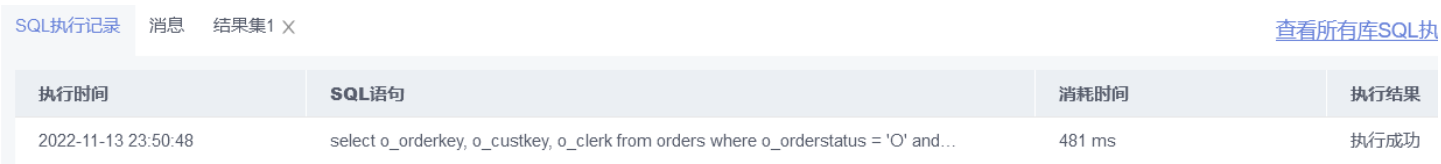
查看所有库

执行时间	SQL语句	消耗时间
2022-11-13 22:55:04	select l_linenumber, l_orderkey, l_tax from lineitem where l_quantity > some(select...	1539 ms

查询15-2

```
1 select o_orderkey, o_custkey, o_clerk
```

```
from orders
where o_orderstatus = 'O'
and o_totalprice > some(
  select o_totalprice
  from orders
  where o_orderdate >= '2020-01-01'::date
  )
```



查询16-1

```
select l_suppkey
from lineitem
where l_shipdate >= '2019-01-01'::date
and l_shipdate <= '2019-12-30'::date
group by l_suppkey
having sum(l_quantity) >= all(
  select sum(l_quantity)
  from lineitem
  where l_shipdate >= '2019-01-01'::date
  and l_shipdate <= '2019-12-30'::date
  group by l_suppkey
)
```

SQL执行记录 消息 结果集1 ×

以下是select l_suppkey from lineitem where l_shipdate >= '2019-01-01'::date and l_s...的执... ⓘ该表不可编辑。

	l_suppkey
1	370

SQL执行记录 消息 结果集1 × 查看所有库SQL执

执行时间	SQL语句	消耗时间	执行结果
2022-11-14 00:03:48	select l_suppkey from lineitem where l_shipdate >= '2019-01-01'::date and l_shipdate <...	1075 ms	执行成功

查询16-2

```
1 select s_name, s_suppkey, s_phone
2 from supplier
3 where s_acctbal >= all(
4   select s_acctbal
5   from supplier
6   )
```

SQL执行记录 消息 结果集1 ×

以下是select s_name, s_suppkey, s_phone from supplier where s_acctbal >= all(selec...的... ⓘ该表不可编辑。

	s_name	s_suppkey	s_phone
1	Supplier#000000892	892	18-893-665-3629

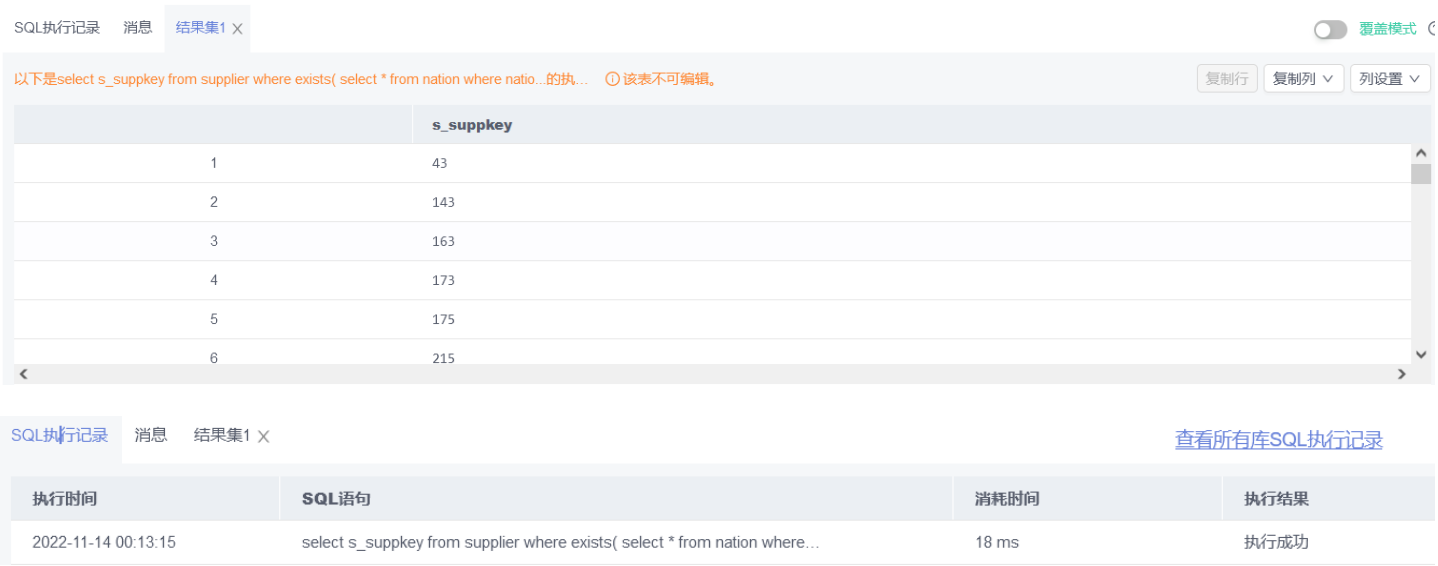
SQL执行记录 消息 结果集1 × 查看所有库SQL执行记录

执行时间	SQL语句	消耗时间	执行结果
2022-11-14 00:08:30	select s_name, s_suppkey, s_phone from supplier where s_acctbal >= all(select...	8 ms	执行成功

查询17-1

```
1 select s_suppkey
2 from supplier
3 where exists(
```

```
    select * from nation
    where nation.n_nationkey = supplier.s_nationkey
    and nation.n_name = 'JAPAN'
    and supplier.s_acctbal > 5000
    )
```



查询17-2

笔者感觉无法使用not exists except解决该问题：

- not exists(A except B)表示的是“A为B的子集”，**但不是真子集，所以当供应商刚好能够供应所有文件时，A=B，但此时A-B仍然为空，所以not exists为真，即把刚好能供应的也算作不能供应的了。**

查询18

Q：group by 为什么不用写n_name?

answer：因为n_name本身也是一个候选码，聚集n_nationkey后在每个分组中是唯一的，而每个分组最终只能输出一个元组，所以没有问题。若n_name不是候选码，每个分组输出到唯一元组时会出问题。

```
select n_name, n_nationkey
from nation, customer
where nation.n_nationkey = customer.c_nationkey
group by n_nationkey, n_name
having count(c_custkey) >= 3
```

查询19

```sql
select ps_partkey
from (
  select ps_partkey, p_size
  from part, partsupp
  where part.p_partkey = partsupp.ps_partkey
  group by ps_partkey, p_size
  having count(ps_suppkey) > 2
  )
where p_size > 20
```

SQL执行记录 消息 结果集1 ×

以下是select ps_partkey from (select ps_partkey, p_size from part, part... ⓘ该表不可编辑。

	ps_partkey
1	23031
2	22061
3	7044
4	13215
5	27572
6	22504
7	36160
8	9615
9	23212
10	30529
11	8857
12	18313
13	6206
14	14776
15	7016
16	10466
17	21177
18	14440
19	14197

SQL执行记录 消息 结果集1 × 查看所有库SQL执行记录

执行时间	SQL语句	消耗时间	执行结果
2022-11-14 14:24:22	select ps_partkey from (select ps_partkey, p_size from part, partsupp where...	192 ms	执行成功

查询20

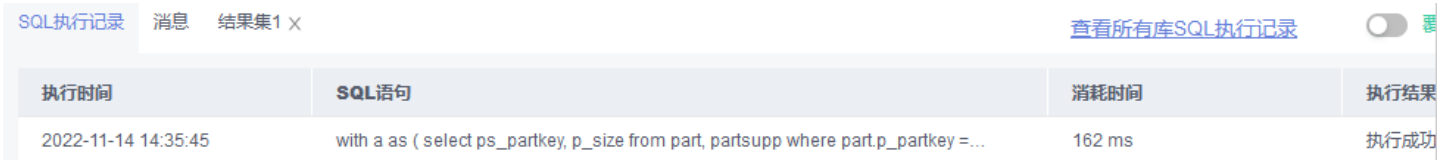
```
1 with a as (
2   select ps_partkey, p_size
```

```
  from part, partsupp
  where part.p_partkey = partsupp.ps_partkey
  group by ps_partkey, p_size
  having count(ps_suppkey) > 2
  )
select ps_partkey
from a
where p_size > 20
```

SQL执行记录　消息　结果集1 ×

以下是with a as (select ps_partkey, p_size from part, partsupp where p... ⓘ该表不可编辑。

	ps_partkey
1	23031
2	22061
3	7044
4	13215
5	27572
6	22504
7	36160
8	9615
9	23212
10	30529
11	8857
12	18313
13	6206
14	14776
15	7016
16	10466
17	21177
18	14440
19	14197



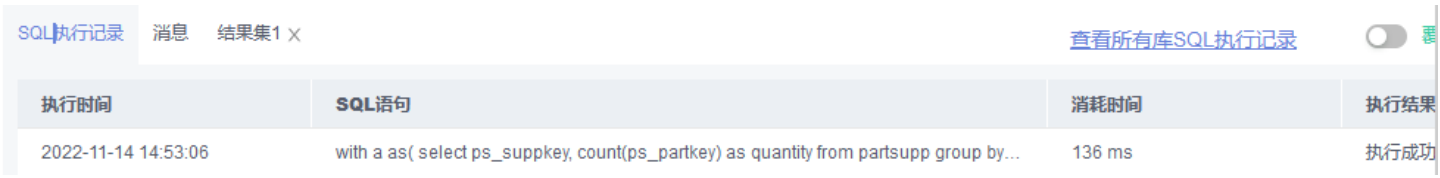
查询21

```
with a as(
  select ps_suppkey, count(ps_partkey) as quantity
  from partsupp
  group by ps_suppkey
  )
select ps_suppkey, quantity
from a
where quantity = (
  select max(quantity)
  from a
  )
```

SQL执行记录 消息 结果集1 ×

以下是with a as(select ps_suppkey, count(ps_partkey) as quantity fro... ⓘ该表不可编辑。

	ps_suppkey	quantity
1	1920	80
2	1092	80
3	1986	80
4	492	80
5	1088	80
6	1679	80
7	862	80
8	686	80
9	721	80
10	343	80
11	1364	80
12	882	80
13	350	80
14	1726	80
15	1574	80
16	1624	80
17	517	80
18	303	80
19	429	80

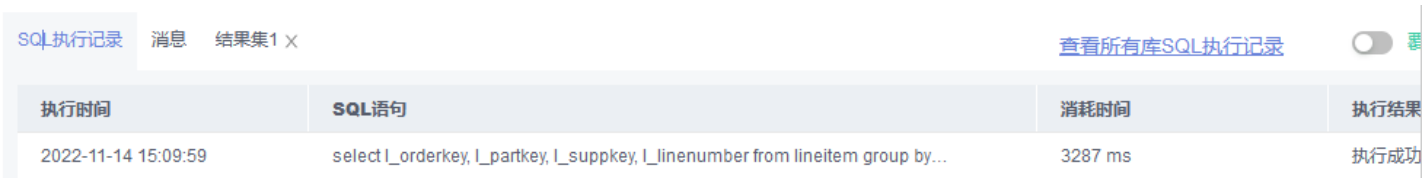


查询22

是超键

```
1 select l_orderkey, l_partkey, l_suppkey, l_linenumber
```

```
2 from lineitem
3 group by l_orderkey, l_partkey, l_suppkey, l_linenumber
4 having count(*) > 1
```



查询23

不成立。**说明厂家对每个零件的价格都有不同的定价。**

```
1 select l_partkey
2 from lineitem
3 group by l_partkey
4 having count(distinct l_extendedprice) > 1
```

SQL执行记录 消息 结果集1 ×

以下是select l_partkey from lineitem group by l_partkey having count(di... ⓘ该表不可编辑。

	l_partkey
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19

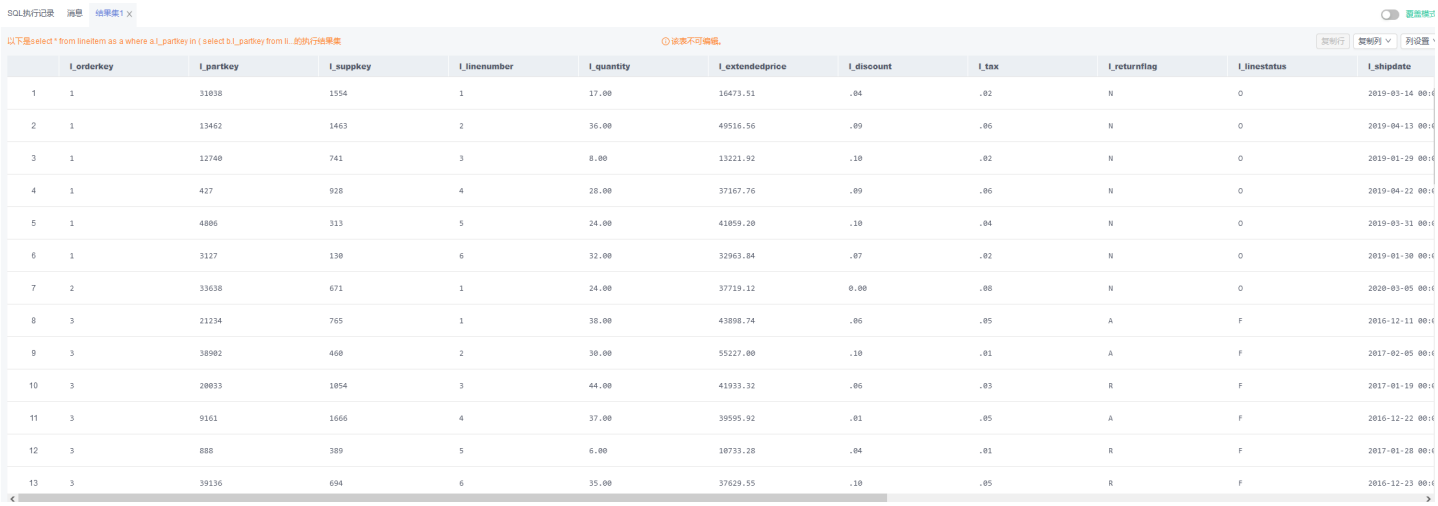
SQL执行记录 消息 结果集1 ×

执行时间	SQL语句	消耗时间	执行结果
2022-11-14 15:44:36	select l_partkey from lineitem group by l_partkey having count(distinct...	2868 ms	执行成功

找出导致不成立的元组：

```
select *
from lineitem as a
where a.l_partkey in (
```

```
4    select b.l_partkey
5    from lineitem b
6    group by b.l_partkey
7    having count (distinct l_extendedprice) > 1
8    )
9
```



SQL执行记录 消息 结果集1 ×

执行时间	SQL语句	消耗时间	执行结果
2022-11-14 15:43:25	select * from lineitem as a where a.l_partkey in (select b.l_partkey from lineitem b...	3376 ms	执行成功

查询24

```
1 insert into orders
2 values('1200001', '20045', 'F', 61365.24, '2017-03-19'::date, '2-HIGH', 'Clerk#0
```

```
---------------开始执行---------------
【拆分SQL完成】：将执行SQL语句数量：（1条）
【执行SQL：(1)】
insert into orders
values('1200001', '20045', 'F', 61365.24, '2017-03-19'::date, '2-HIGH', 'Clerk#000000098', 0, 'furiously special f')
执行成功，耗时：[69ms.]
```

查询25

```
1 insert into partsupp(
2    select '20', ps_suppkey, ps_availqty, ps_supplycost, ps_comment
```

```
  from partsupp
  where ps_partkey = '32'
  /*避免零件20的经销商重复*/
  and ps_suppkey not in (
    select ps_suppkey
    from partsupp
    where ps_partkey = '20'
    )
  )
```

```
---------------开始执行---------------

【拆分SQL完成】：将执行SQL语句数量：（1条）

【执行SQL：(1)】
insert into partsupp(
  select '20', ps_suppkey, ps_availqty, ps_supplycost, ps_comment
  from partsupp
  where ps_partkey = '32'
  and ps_suppkey not in (
    select ps_suppkey
    from partsupp
    where ps_partkey = '20'
    )
  )
执行成功，耗时：[92ms.]
```

查询26

```
delete from lineitem
where l_returnflag  = 'R'
```

```
---------------开始执行---------------

【拆分SQL完成】：将执行SQL语句数量：（1条）

【执行SQL：(1)】
delete from lineitem
where l_returnflag  = 'R'
执行成功，耗时：[816ms.]
```

查询27

```
update lineitem
set l_receiptdate = l_commitdate
where l_orderkey in (
  select o_orderkey
  from orders
  where o_orderdate >= '2020-01-01'::date
  )
```

SQL执行记录　消息

```
--------------开始执行--------------

【拆分SQL完成】：将执行SQL语句数量：（1条）

【执行SQL：(1)】
update lineitem
set l_receiptdate = l_commitdate
where l_orderkey in (
  select o_orderkey
  from orders
  where o_orderdate >= '2020-01-01'::date
  )
执行成功，耗时：[3913ms.]
```

查询28

```
update lineitem
set l_commitdate =
case when l_orderkey in (
  select o_orderkey
  from orders
  /*where o_orderpriority = '3-MEDIUM'*/
  where SUBSTRING(o_orderpriority,1,1) + 0 > 3
  )
  then l_commitdate + '2 day'
  else l_commitdate + '1 day'
end
```

```
---------------开始执行---------------
【拆分SQL完成】：将执行SQL语句数量：（1条）
【执行SQL：(1)】
update lineitem
set l_commitdate =
case when l_orderkey in (
  select o_orderkey
  from orders
  /*where o_orderpriority = '3-MEDIUM'*/
  where SUBSTRING(o_orderpriority,1,1) + 0 > 3
  )
  then l_commitdate + '2 day'
  else l_commitdate + '1 day'
end
执行成功，耗时：[11066ms.]
```

注意：下式得到的为优先级为4、5的表：

```sql
select o_orderkey, o_orderpriority
from orders
/*where o_orderpriority = '3-MEDIUM'*/
where SUBSTRING(o_orderpriority,1,1) + 0 > 3
```

SQL执行记录 消息 结果集1 ×

以下是select o_orderkey, o_orderpriority from orders /*where o... ⓘ该表不可编辑。

	o_orderkey	o_orderpriority
1	1	5-LOW
2	3	5-LOW
3	4	5-LOW
4	5	5-LOW
5	6	4-NOT SPECIFIED
6	35	4-NOT SPECIFIED
7	38	4-NOT SPECIFIED
8	66	5-LOW
9	67	4-NOT SPECIFIED
10	69	4-NOT SPECIFIED
11	70	5-LOW
12	71	4-NOT SPECIFIED
13	99	4-NOT SPECIFIED
14	100	4-NOT SPECIFIED
15	103	4-NOT SPECIFIED
16	129	5-LOW
17	134	4-NOT SPECIFIED
18	135	4-NOT SPECIFIED
19	160	4-NOT SPECIFIED

SQL执行记录 消息 结果集1 ×

查看所有库SQL执行记

执行时间	SQL语句	消耗时间
2022-11-14 16:42:27	select o_orderkey, o_orderpriority from orders /*where o_orderpriority = '3-MEDIUM'*...	311 ms

查询29

```
1 select o_orderkey, o_totalprice, rank() over(order by o_totalprice desc)as Rank_
```

```
2 from orders
```

SQL执行记录 消息 结果集1 ×

以下是select o_orderkey, o_totalprice, rank() over(order by o_t... ⓘ该表不可编辑。

	o_orderkey	o_totalprice	rank_
1	209028	505770.15	1
2	528388	497758.84	2
3	993697	487758.42	3
4	1111238	485577.76	4
5	489319	484671.66	5
6	366692	483521.14	6
7	546785	481047.81	7
8	326117	473020.26	8
9	149509	471154.02	9
10	185124	460604.60	10
11	227297	457002.50	11
12	800518	455125.90	12
13	135808	450689.29	13
14	1005734	450359.31	14
15	580609	448411.89	15
16	12070	448042.34	16
17	123907	445855.74	17
18	901892	444778.64	18
19	173796	444258.68	19

SQL执行记录 消息 结果集1 ×

查看所有库SQL执行记录

执行时间	SQL语句	消耗时间
2022-11-14 16:46:30	select o_orderkey, o_totalprice, rank() over(order by o_totalprice desc)as Rank_ fro...	479 ms

五、总结

通过本次实验，我们对单表简单查询、多表查询、统计查询、嵌套查询、with 临时视图查询、键/函数依赖分析与表的插入、删除、更新有了更加深刻的印象，加深了我们对SQL语言中DML的理解，熟练掌握了各种语句的使用方法。

虽然实验指导手册下面有可供参考的实验示例，但实际上，它是存在错误的。我们对每一个查询进行仔细研读并誊写，最终与实验示例进行比较，并对其可能有的错误进行了分析比较与查证，最终得到了上述的正确答案。这对我们的数据库语句编写能力的提升以及debug能力的提升有很大的帮助。