

Minrui Li, ml7136

CS-GY 6083 B

11/19/2021

Problem1

I.

```
select b.dname "DEPARTMENT", sum(sal) "SALARY"
from ml_emp a join ml_dept b on a.deptno=b.deptno
group by b.dname
having sum(sal) > (select sum(sal) from ml_emp a join ml_dept b on a.deptno=b.deptno
where b.dname='SALES' group by a.deptno)
```

The screenshot shows the Live SQL interface. On the left is a sidebar with navigation options: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains the following SQL query:

```
1 select b.dname "DEPARTMENT", sum(sal) "SALARY"
2 from ml_emp a join ml_dept b on a.deptno=b.deptno
3 group by b.dname
4 having sum(sal) > (select sum(sal) from ml_emp a join ml_dept b on a.deptno=b.deptno where b.dname='SALES' group by a.deptno)
5
```

Below the query editor, the results are displayed in a table:

DEPARTMENT	SALARY
RESEARCH	32625

Below the table is a 'Download CSV' link. At the bottom of the interface, there is a footer with Oracle branding and copyright information: © 2021 Oracle Corporation · Privacy · Terms of Use. It also mentions 'Oracle Learning Library · Ask Tom · Dev Gym · Database Documentation · Follow on Twitter' and 'Live SQL 21.4.1, running Oracle Database 19c Enterprise Edition - 19.8.0.0.0'. A note at the bottom states 'Built with ❤️ using Oracle APEX running on Oracle Cloud Infrastructure and Oracle Kubernetes Engine'.

II.

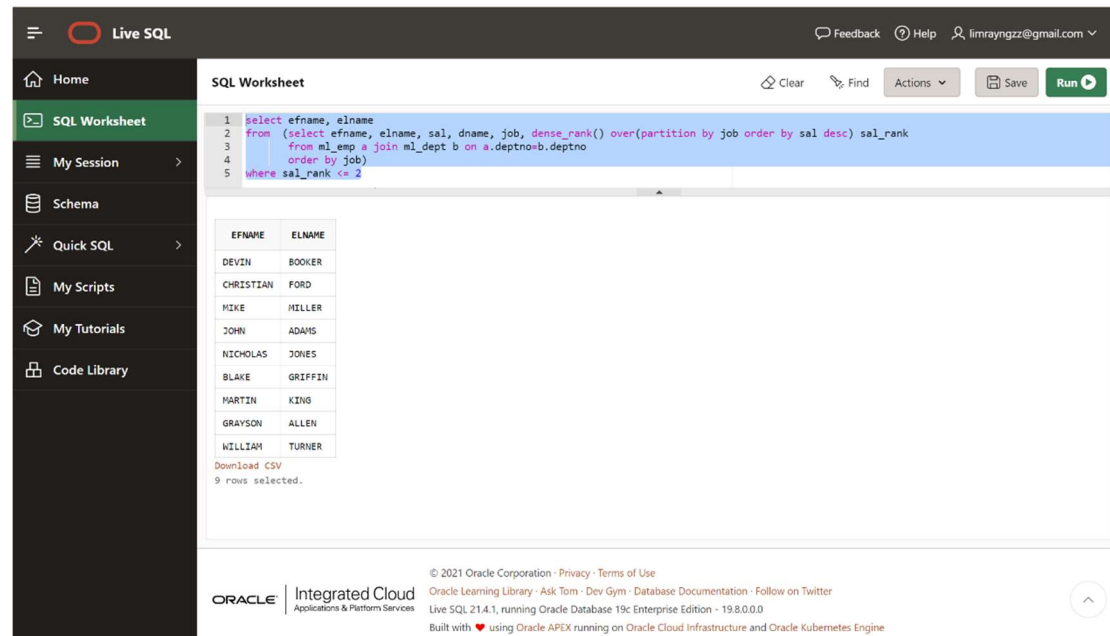
```
select a.ename "FIRST NAME", a.lname "LAST NAME", a.hiredate, b.ename
"MGR FIRST NAME", b.lname "MGR LAST NAME", b.hiredate "MGR
HIREDATE"
from ml_emp a join ml_emp b on a.mgr=b.empno
where a.hiredate<b.hiredate;
```



```

select efname, elname
from (select efname, elname, sal, dname, job, dense_rank() over(partition by job
order by sal desc) sal_rank
      from ml_emp a join ml_dept b on a.deptno=b.deptno
      order by job)
where sal_rank <= 2

```



The screenshot shows the 'Live SQL' web interface. On the left is a navigation menu with options: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains a text editor with the following SQL query:

```

1 select efname, elname
2 from (select efname, elname, sal, dname, job, dense_rank() over(partition by job order by sal desc) sal_rank
3      from ml_emp a join ml_dept b on a.deptno=b.deptno
4      order by job)
5 where sal_rank <= 2

```

Below the editor, the results are displayed in a table:

EFNAME	ELNAME
DEVIN	BOOKER
CHRISTIAN	FORD
MIKE	MILLER
JOHN	ADAMS
NICHOLAS	JONES
BLAKE	GRIFFIN
MARTIN	KING
GRAYSON	ALLEN
WILLIAM	TURNER

Below the table, it says 'Download CSV' and '9 rows selected.' The footer of the interface includes Oracle branding, 'Integrated Cloud Applications & Platform Services', and copyright information for 2021 Oracle Corporation.

V.

```

select a.empno, a.efname, a.elname, b.dname, d.pname, c.hours
from ml_emp a join ml_dept b on a.deptno=b.deptno join ml_proemp c on
a.empno=c.empno join ml_project d on c.projid=d.projid join
      (select deptno, avg(sal) avg_sal from ml_emp group by deptno) e on
e.deptno=a.deptno
where a.sal > e.avg_sal;

```

≡

Live SQL

FeedbackHelplimrayngzz@gmail.com

Home

SQL Worksheet

My Session

Schema

Quick SQL

My Scripts

My Tutorials

Code Library

SQL Worksheet

ClearFindActionsSaveRun

1select a.empno, a.efname, a.elname, b.dname, d.pname, c.hours

2from ml_emp a join ml_dept b on a.deptno=b.deptno join ml_proemp c on a.empno=c.empno join ml_project d on c.projid=d.projid join

3(select deptno, avg(sal) avg_sal from ml_emp group by deptno) e on e.deptno=a.deptno

4where a.sal > e.avg_sal;

EMPNO	EFNAME	ELNAME	DNAME	PNAME	HOURS
7499	GRAYSON	ALLEN	SALES	Upgrade Marketing Database	42
7698	BLAKE	GRIFFIN	SALES	Create Sales Data Warehouse	48
7698	BLAKE	GRIFFIN	SALES	Design and Create New DB	48
7698	BLAKE	GRIFFIN	SALES	Develop Employee Leave System	50
7698	BLAKE	GRIFFIN	SALES	Design Backup Procedure	60
7566	NICHOLAS	JONES	RESEARCH	Upgrade Marketing Database	30
7902	CHRISTIAN	FORD	RESEARCH	Upgrade Marketing Database	38
7788	DEVIN	BOOKER	RESEARCH	Design and Create New DB	28
7788	DEVIN	BOOKER	RESEARCH	Develop Employee Leave System	38
7566	NICHOLAS	JONES	RESEARCH	Design Backup Procedure	75
7788	DEVIN	BOOKER	RESEARCH	Design Backup Procedure	42
7902	CHRISTIAN	FORD	RESEARCH	Design Backup Procedure	36

Download CSV

ORACLEIntegrated Cloud

Applications & Platform Services

© 2021 Oracle Corporation · Privacy · Terms of Use

Oracle Learning Library · Ask Tom · Dev Gym · Database Documentation · Follow on Twitter

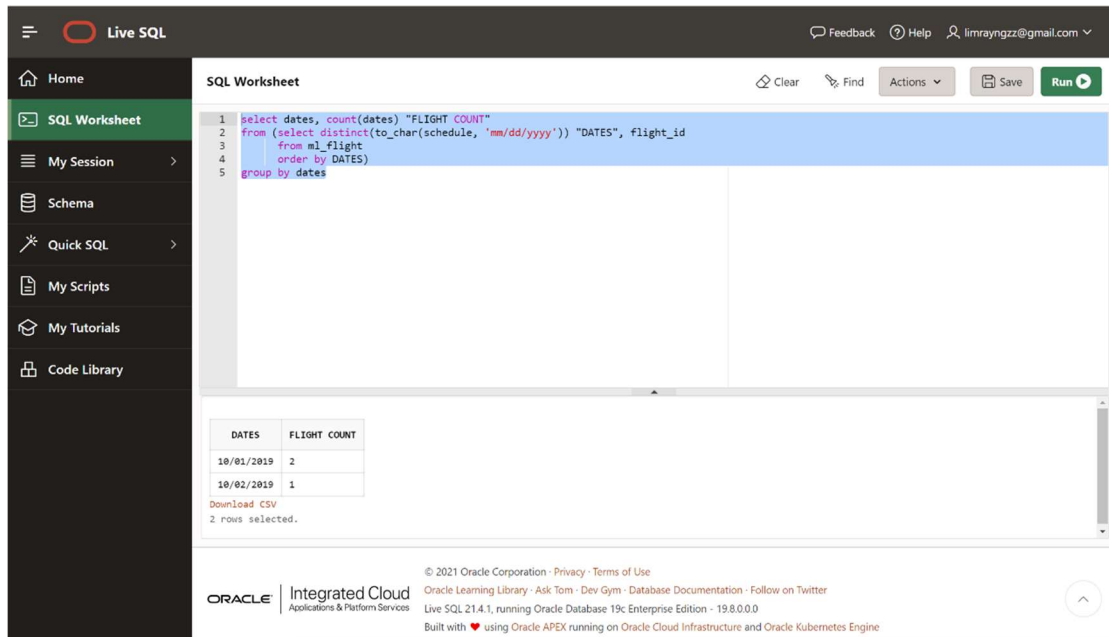
Live SQL 21.4.1, running Oracle Database 19c Enterprise Edition - 19.8.0.0.0

Built with using Oracle APEX running on Oracle Cloud Infrastructure and Oracle Kubernetes Engine

Problem2

I.

```
select count(dates)
from (select distinct(to_char(schedule, 'dd-mon-yy')) "DATES", flight_id
      from ml_flight
      order by DATES)
group by dates
```



The screenshot shows the Oracle Live SQL interface. On the left is a sidebar with navigation options: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains a text editor with the following SQL query:

```
1 select dates, count(dates) "FLIGHT COUNT"
2 from (select distinct(to_char(schedule, 'mm/dd/yyyy')) "DATES", flight_id
3      from ml_flight
4      order by DATES)
5 group by dates
```

Below the editor, the results are displayed in a table:

DATES	FLIGHT COUNT
18/01/2019	2
18/02/2019	1

Below the table, it says 'Download CSV' and '2 rows selected.' The footer of the interface includes the Oracle logo, 'Integrated Cloud Applications & Platform Services', and copyright information for 2021 Oracle Corporation.

II.

```
select dates, sum(capacity)/1000 || 'k' "TOTAL_CAPACITY"
from (select distinct(to_char(a.schedule, 'mm/dd/yyyy')) "DATES", a.flight_id,
      b.capacity
      from ml_flight a join ml_size b on a.flight_id=b.flight_id
      Order by DATES)
group by dates;
```

Live SQL

Feedback Help limrayngzz@gmail.com

Home

SQL Worksheet

My Session

Schema

Quick SQL

My Scripts

My Tutorials

Code Library

SQL Worksheet

ClearFindActionsSaveRun

```
1 select dates, sum(capacity)/1000 || 'k' "TOTAL_CAPACITY"
2 from (select distinct(to_char(a.schedule, 'mm/dd/yyyy')) "DATES", a.flight_id, b.capacity
3      from m1_flight a join m1_size b on a.flight_id=b.flight_id
4      Order by DATES)
5 group by dates;
6
```

DATES	TOTAL_CAPACITY
18/01/2019	17k
18/02/2019	18k

Download CSV
2 rows selected.

ORACLEIntegrated Cloud

© 2021 Oracle Corporation · Privacy · Terms of Use

Oracle Learning Library · Ask Tom · Dev Gym · Database Documentation · Follow on Twitter

Live SQL 21.4.1, running Oracle Database 19c Enterprise Edition - 19.8.0.0.0

Built with using Oracle APEX running on Oracle Cloud Infrastructure and Oracle Kubernetes Engine

Problem3

I.

create or replace view myview

as

select a.cust_id, a.fname, a.lname, b.order_id, b.order_date, d.prod_id, d.descr, c.quant,
d.unit_price,

c.quant*d.unit_price "TOTAL_PRICE", e.shade

from ml_cust a join ml_order b on a.cust_id=b.cust_id join ml_ord_prod c on
c.order_id=b.order_id

join ml_prod d on d.prod_id=c.prod_id join ml_finish e on
e.finish_id=c.finish_id

where c.quant*d.unit_price >= 1000

order by total_price

with read only;

select * from myview;

The screenshot shows the Live SQL interface. The SQL Worksheet contains the following code:

```
1 create or replace view myview
2 as
3 select a.cust_id, a.fname, a.lname, b.order_id, b.order_date, d.prod_id, d.descr, c.quant, d.unit_price,
4       c.quant*d.unit_price "TOTAL_PRICE", e.shade
5 from ml_cust a join ml_order b on a.cust_id=b.cust_id join ml_ord_prod c on c.order_id=b.order_id
6      join ml_prod d on d.prod_id=c.prod_id join ml_finish e on e.finish_id=c.finish_id
7 where c.quant*d.unit_price >= 1000
8 order by total_price
9 with read only;
10
11 select * from myview;
```

The results table shows 12 rows of data:

CUST_ID	FNAM	LNAM	ORDER_ID	ORDER_DATE	PROD_ID	DESCR	QUANT	UNIT_PRICE	TOTAL_PRICE	SHADE
12345	James	Potter	1007	21-NOV-20	11	4-Dr Dresser	2	500	1000	Oak
33776	David	Taylor	1029	21-DEC-20	6	Settee	2	500	1000	Puritan Pine
98711	Joseph	Iopez	1014	05-DEC-18	10	Bookcase	1	1000	1000	Red Chestnut
98987	Richard	Martin	1015	20-FEB-17	2	Cabinets	2	525	1050	Natural Ash
45678	John	Miller	1022	10-JUN-18	4	Entertainment Center	2	650	1300	Red Chestnut
67890	Mary	Brown	1020	09-JUL-20	3	Couch	2	670	1340	Provincial
10987	Karen	Iglesias	1009	12-APR-21	13	Armoire	1	1500	1500	Fruitwood
99876	Linda	Wilson	1024	05-MAY-18	14	Windsor Chair	2	890	1780	Natural
99876	Linda	Wilson	1012	18-SEP-18	8	Wardrobe	2	900	1800	Classic Gray
90876	Robert	Jones	1010	24-FEB-20	10	Bookcase	2	1000	2000	Cherry
45678	John	Miller	1022	10-JUN-18	10	Bookcase	3	1000	3000	Cherry
22334	William	Martinez	1023	03-JUL-17	13	Armoire	2	1500	3000	Simply White

Download CSV
12 rows selected.

II.

select prod_id, descr, shade, total_quantity_sold

from (select prod_id, descr, shade, total_quantity_sold, dense_rank() over(order by
TOTAL_QUANTITY_SOLD desc) SOLD_RANK

from (select b.prod_id, descr, shade, sum(b.quant)
TOTAL_QUANTITY_SOLD

from ml_prod a join ml_ord_prod b on a.prod_id=b.prod_id join
ml_order c on c.order_id=b.order_id join ml_finish d on b.finish_id=d.finish_id

where months_between(sysdate, order_date) <= 6
group by b.prod_id, descr, shade))
where sold_rank<=3;

The screenshot shows the Live SQL interface with a SQL Worksheet. The query is as follows:

```

1 select prod_id, descr, shade, total_quantity_sold
2 from (select prod_id, descr, shade, total_quantity_sold, dense_rank() over(order by TOTAL_QUANTITY_SOLD desc) SOLD_RANK
3      from (select b.prod_id, descr, shade, sum(b.quant) TOTAL_QUANTITY_SOLD
4            from ml_prod a join ml_ord_prod b on a.prod_id=b.prod_id join ml_order c on c.order_id=b.order_id join ml_finish d on b.finish_id=d.finish_id
5            where months_between(sysdate, order_date) <= 6
6            group by b.prod_id, descr, shade))
7 where sold_rank<=3;
8

```

The results table shows the following data:

PROD_ID	DESCR	SHADE	TOTAL_QUANTITY_SOLD
5	Writer Desk	Natural	1

Download CSV

© 2021 Oracle Corporation · Privacy · Terms of Use
Oracle Learning Library · Ask Tom · Dev Gym · Database Documentation · Follow on Twitter
Live SQL 21.4.1, running Oracle Database 19c Enterprise Edition - 19.8.0.0.0
Built with ❤️ using Oracle APEX running on Oracle Cloud Infrastructure and Oracle Kubernetes Engine

III.

select a.prod_id, c.shade
from ml_ord_prod a join ml_order b on a.order_id=b.order_id join ml_finish c on
c.finish_id=a.finish_id
where b.order_date not between to_date('2020-10-1', 'YYYY-MM-DD') and
to_date('2020-12-31', 'YYYY-MM-DD');

The screenshot shows the Live SQL interface with a SQL Worksheet. The query is as follows:

```

1 select a.prod_id, c.shade
2 from ml_ord_prod a join ml_order b on a.order_id=b.order_id join ml_finish c on c.finish_id=a.finish_id
3 where b.order_date not between to_date('2020-10-1', 'YYYY-MM-DD') and to_date('2020-12-31', 'YYYY-MM-DD');
4

```

The results table shows the following data:

PROD_ID	SHADE
1	Driftwood
1	Puritan Pine
2	Natural Ash
3	Provincial
4	Cherry
4	Red Chestnut
4	Green Velvet
5	Driftwood
5	Natural
6	Fruitwood
7	Natural Ash
8	Classic Gray
8	Green Velvet
9	Gunstock
9	Simply White
9	Green Velvet
10	Cherry
10	Cherry
10	Red Chestnut
13	Simply White
13	Fruitwood
14	Natural
15	Puritan Pine
15	Fruitwood

Download CSV
24 rows selected.

© 2021 Oracle Corporation · Privacy · Terms of Use
Oracle Learning Library · Ask Tom · Dev Gym · Database Documentation · Follow on Twitter
Live SQL 21.4.1, running Oracle Database 19c Enterprise Edition - 19.8.0.0.0
Built with ❤️ using Oracle APEX running on Oracle Cloud Infrastructure and Oracle Kubernetes Engine