Assignment - 3

(Team Assignment)

Team Members:

Keerthanaa Kannan	001593845
Nency Panchani	002746779
Aashay Pawar	002134382
Pawan Kumar Krishnan	002743773

Q1) **10M**

To store Brick color and shape we have a table called **BRICKS** with 2 columns **COLOR** and **SHAPE**. Data in table is as below:

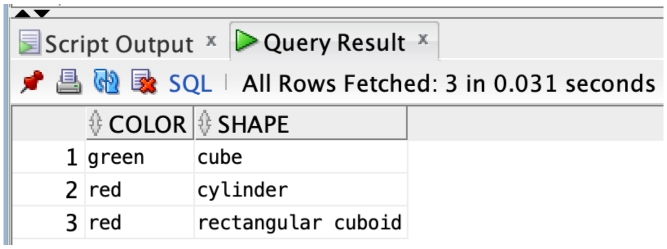
COLOR	**SHAPE**
blue	cube
blue	cylinder
green	cube
red	cylinder
red	rectangular cuboid
yellow	rectangular cuboid

Create the above table and insert the rows in your own database instance (your TEAM's cloud database instance). You should identify the datatype and length of each column so that the provided data can be stored. Additionally, Write a query that will return the rows from this table where the value for COLOR is green or red in 2 different ways.

Query:

```
create table bricks(
color varchar(255),
shape varchar(255)
);

insert into bricks
(color, shape)
with mydata as(
select 'blue', 'cube' from dual union all
select 'blue', 'cylinder' from dual union all
select 'green', 'cube' from dual union all
select 'red', 'cylinder' from dual union all
select 'red', 'rectangular cuboid' from dual union all
select 'yellow', 'rectangular cuboid' from dual
)
select * from mydata;

select * from bricks where color = 'green' or color = 'red';
select * from bricks where color in ('green', 'red');
```

Screenshot:



All Rows Fetched: 3 in 0.031 seconds

	COLOR	SHAPE
1	green	cube
2	red	cylinder
3	red	rectangular cuboid

Q2) **10M**

Create a table and store people's details in the table called **PEOPLE** with below data (FULL_NAME and HEIGHT_IN_CM are the columns and HEIGHT_IN_CM is a VARCHAR column and you should store data in table as provided below as-is)

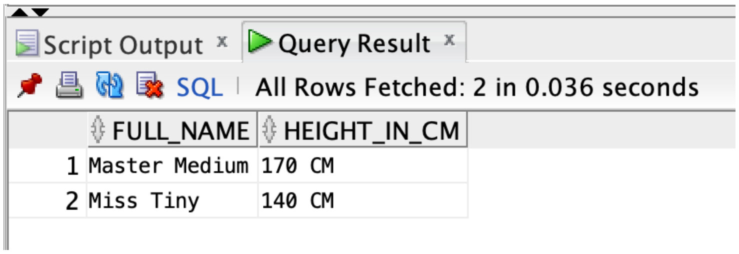
FULL_NAME	HEIGHT_IN_CM
Mr. Tall	200 CM
Master Medium	170 CM
Miss Tiny	140 CM

Write a query to return rows for people 170cm and smaller.

Query:

```
create table people(
full_name varchar(255),
height_in_cm varchar(255)
);

insert into people
(full_name, height_in_cm)
with mydata as(
select 'Mr. Tall', '200' from dual union all
select 'Master Medium', '170' from dual union all
select 'Miss Tiny ', '140' from dual
)
select * from mydata;

select full_name, (height_in_cm || ' CM') as height_in_cm from people where height_in_cm <= '170';
```

Screenshot:



All Rows Fetched: 2 in 0.036 seconds

	FULL_NAME	HEIGHT_IN_CM
1	Master Medium	170 CM
2	Miss Tiny	140 CM

Q3) **20M**

Use below connection details connect to the database

Username: inventory_mgmt_ro **password**: **EveryOneGetsAccess123#** **schema**: inventory_mgmt

wallet file: Wallet_DAMG7370.zip

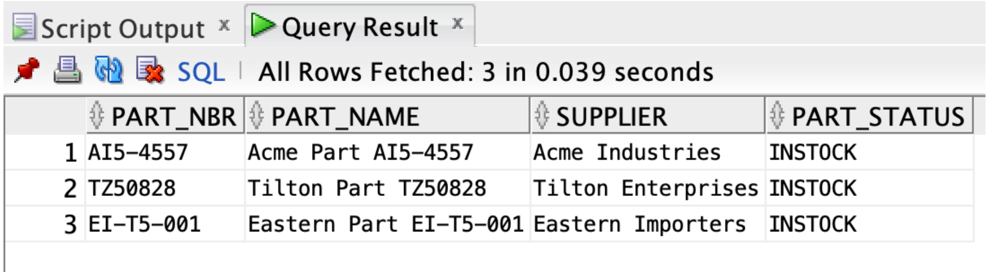
Download Wallet_DAMG7370.zip

You need to explore the 25 tables, understand the tables structure and write a query to show the below data.

```
PART_NBR      PART_NAME           SUPPLIER          PART_STATUS

---------------- ----------------------- -------------------    ----------

AI5-4557      Acme Part AI5-4557     Acme Industries     In Stock

TZ50828       Tilton Part TZ50828    Tilton Enterprises    In Stock

EI-T5-001     Eastern Part EI-T5-001  Eastern Importers   In Stock
```

Query:
select a.part_nbr part_nbr, a.name part_name, b.name supplier, a.status part_status from
inventory_mgmt.PART a
inner join
(select * from inventory_mgmt.SUPPLIER) b
on b.supplier_id = a.supplier_id;

Screenshot:



All Rows Fetched: 3 in 0.039 seconds

	PART_NBR	PART_NAME	SUPPLIER	PART_STATUS
1	AI5-4557	Acme Part AI5-4557	Acme Industries	INSTOCK
2	TZ50828	Tilton Part TZ50828	Tilton Enterprises	INSTOCK
3	EI-T5-001	Eastern Part EI-T5-001	Eastern Importers	INSTOCK

Q4) **20M**

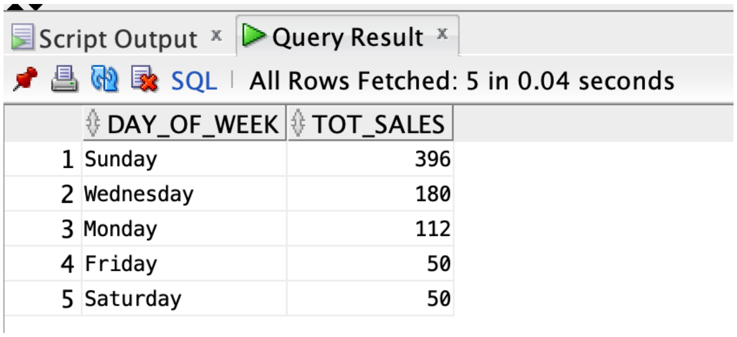
Using same database connection details (as question 3), Find DAY wise total sales (output should show as below)

```
DAY_OF_WEEK            TOT_SALES

------------ -         ------------

SUNDAY         396

WEDNESDAY       180

MONDAY          112

FRIDAY           50

SATURDAY         50
```

Query:

```
select day_of_week, sum(sale_price) tot_sales from
(select to_char(to_date(order_dt, 'dd-mm-yy'), 'Day') day_of_week, sale_price from
inventory_mgmt.DISPUTED_ORDERS where sale_price is not null)
group by day_of_week order by tot_sales desc;
```

Screenshot:



All Rows Fetched: 5 in 0.04 seconds

	DAY_OF_WEEK	TOT_SALES
1	Sunday	396
2	Wednesday	180
3	Monday	112
4	Friday	50
5	Saturday	50

Q5) 10M

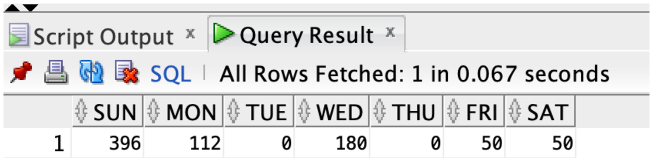
Using same database connection details (as question 3), Find DAY wise total sales (output should show as below. Output Similar to Question 4 however need the output transposed from rows to columns)

```
       SUN        MON        TUE        WED        THU        FRI        SAT
---------- ---------- ---------- ---------- ---------- ---------- ----------
       396        112          0        180          0         50         50
```

Query:

```
select
   sum(nvl(case when to_char(ORDER_DT,'DAY') like 'SUNDAY%' then SALE_PRICE END,0)) as sun
   ,sum(nvl(case when to_char(ORDER_DT,'DAY') like 'MON%' then SALE_PRICE END,0)) as mon
   ,sum(nvl(case when to_char(ORDER_DT,'DAY') like 'TUES%' then SALE_PRICE END,0)) as tue
   ,sum(nvl(case when to_char(ORDER_DT,'DAY') like 'WEDNESDAY' then SALE_PRICE END,0)) as wed
   ,sum(nvl(case when to_char(ORDER_DT,'DAY') like 'THURSD%' then SALE_PRICE END,0)) as thu
   ,sum(nvl(case when to_char(ORDER_DT,'DAY') like 'FRID%' then SALE_PRICE END,0)) as fri
   ,sum(nvl(case when to_char(ORDER_DT,'DAY') like 'SATURDA%' then SALE_PRICE END,0)) as sat
   from inventory_mgmt.CUST_ORDER;
```

Screenshot:



All Rows Fetched: 1 in 0.067 seconds

	SUN	MON	TUE	WED	THU	FRI	SAT
1	396	112	0	180	0	50	50

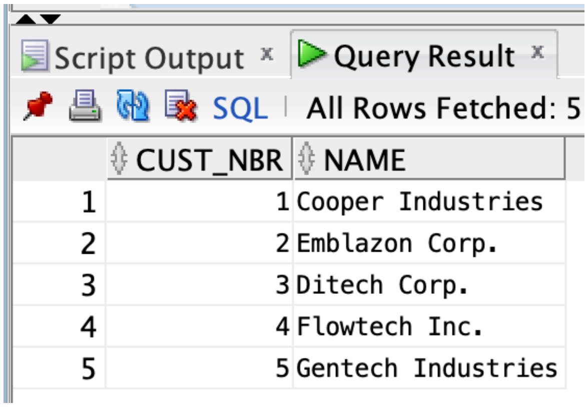
Q6) **20M**

Using same database connection details (as question 3), Write a query to retrieves all the customer numbers and names in region 5 AND include all the customer numbers and names who ae with the sales representative 'MARTIN'

Query:

```
select a.cust_nbr, a.name from inventory_mgmt.CUSTOMER a where a.region_id = '5';
```

Screenshot:



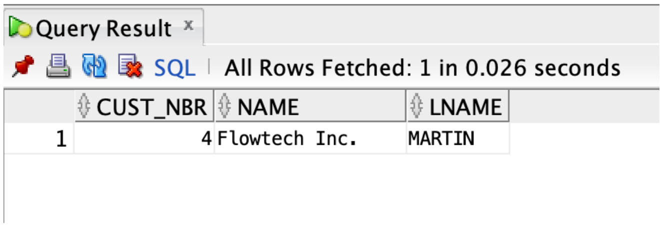
All Rows Fetched: 5

	CUST_NBR	NAME
1	1	Cooper Industries
2	2	Emblazon Corp.
3	3	Ditech Corp.
4	4	Flowtech Inc.
5	5	Gentech Industries

Query:

```
select a.cust_nbr, a.name, c.lname from inventory_mgmt.CUSTOMER a
inner join inventory_mgmt.cust_order b
on a.cust_nbr = b.cust_nbr
inner join inventory_mgmt.EMPLOYEE c
on c.emp_id  = b.sales_emp_id
where lower(c.lname) like '%martin%' and a.region_id = 5 group by a.cust_nbr, a.name, c.lname;
```

Screenshot:



All Rows Fetched: 1 in 0.026 seconds

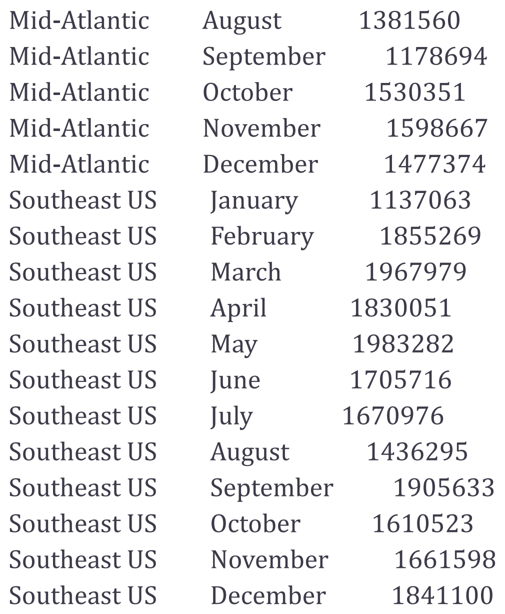
	CUST_NBR	NAME	LNAME
1	4	Flowtech Inc.	MARTIN

Q7) **10M**

Using same database connection details (as question 3), Write a query to print the monthly total sales for each region

Output should be as below

```
REGION           MONTH     SUM(O.TOT_SALES)
-------------------- --------- ----------------
New England        January        1527645
New England        February        1847238
New England        March          1699449
New England        April          1792866
New England        May            1698855
New England        June           1510062
New England        July           1678002
New England        August         1642968
New England        September        1726767
New England        October        1648944
New England        November        1384185
New England        December        1599942
Mid-Atlantic      January        1832091
Mid-Atlantic      February        1286028
Mid-Atlantic      March          1911093
Mid-Atlantic      April          1623438
Mid-Atlantic      May            1778805
Mid-Atlantic      June           1504455
Mid-Atlantic      July           1820742
```



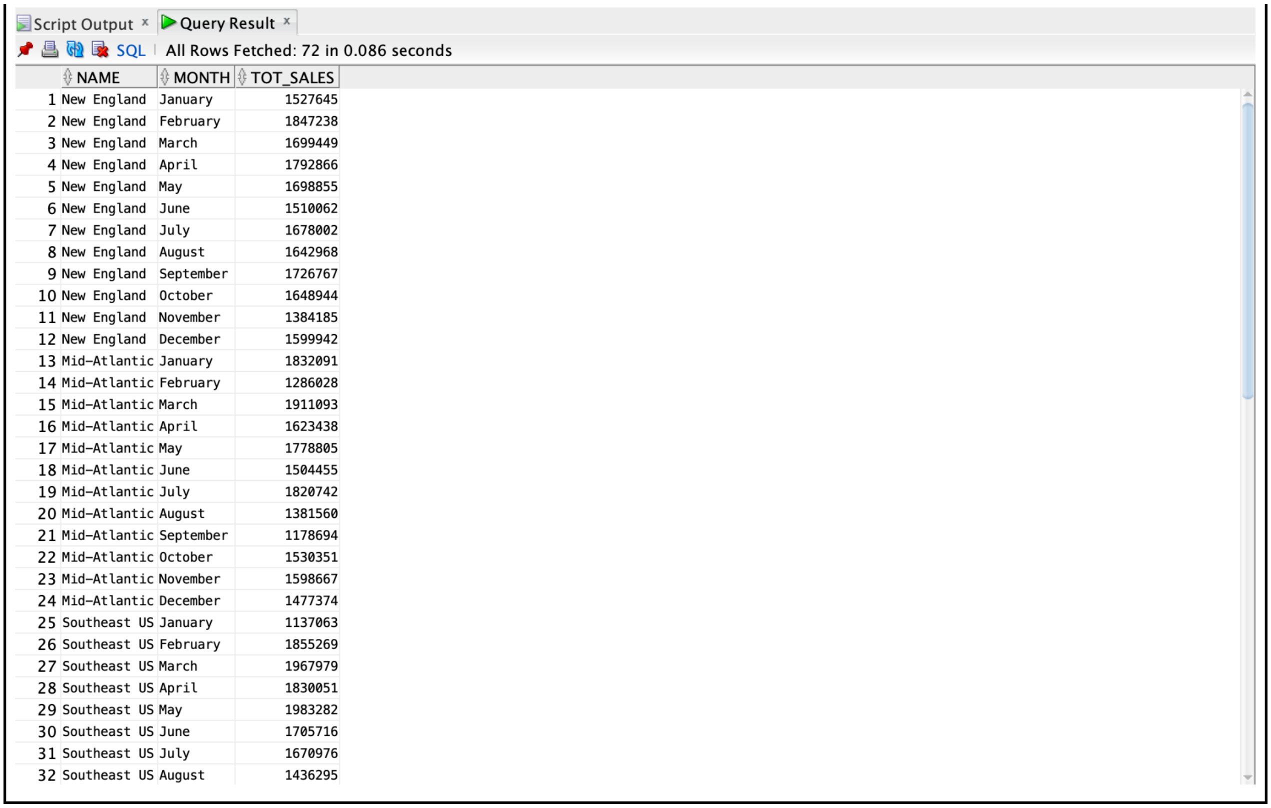
```
Mid-Atlantic       August          1381560
Mid-Atlantic       September        1178694
Mid-Atlantic       October         1530351
Mid-Atlantic       November         1598667
Mid-Atlantic       December         1477374
Southeast US       January         1137063
Southeast US       February         1855269
Southeast US       March           1967979
Southeast US       April           1830051
Southeast US       May             1983282
Southeast US       June            1705716
Southeast US       July            1670976
Southeast US       August          1436295
Southeast US       September        1905633
Southeast US       October         1610523
Southeast US       November         1661598
Southeast US       December         1841100
```

Query:

```
select a.name, b.month, b.tot_sales from inventory_mgmt.REGION a
inner join
(select region_id, to_char(to_date(month, 'MM'), 'Month') month, sum(tot_sales)*3 as tot_sales from
inventory_mgmt.ORDERS group by region_id, month order by region_id) b
on b.region_id = a.region_id;
```

Screenshot:

Same as **DISPLAYED** output.



All Rows Fetched: 72 in 0.086 seconds

	NAME	MONTH	TOT_SALES
1	New England	January	1527645
2	New England	February	1847238
3	New England	March	1699449
4	New England	April	1792866
5	New England	May	1698855
6	New England	June	1510062
7	New England	July	1678002
8	New England	August	1642968
9	New England	September	1726767
10	New England	October	1648944
11	New England	November	1384185
12	New England	December	1599942
13	Mid-Atlantic	January	1832091
14	Mid-Atlantic	February	1286028
15	Mid-Atlantic	March	1911093
16	Mid-Atlantic	April	1623438
17	Mid-Atlantic	May	1778805
18	Mid-Atlantic	June	1504455
19	Mid-Atlantic	July	1820742
20	Mid-Atlantic	August	1381560
21	Mid-Atlantic	September	1178694
22	Mid-Atlantic	October	1530351
23	Mid-Atlantic	November	1598667
24	Mid-Atlantic	December	1477374
25	Southeast US	January	1137063
26	Southeast US	February	1855269
27	Southeast US	March	1967979
28	Southeast US	April	1830051
29	Southeast US	May	1983282
30	Southeast US	June	1705716
31	Southeast US	July	1670976
32	Southeast US	August	1436295