ASSIGNMENT :-3

NAME:
ROLL NO:

PROBLEM STATEMENTS: Consider database created in assignment 1 to design SQL queries using SQL DML
satatements : Insert, Select, Update, Delete with operators, function and set operator.

****

A) Exercise on retrieving records from a table.

****

1) Find the name of all Client.

```
mysql> select name from CLIENT_MASTER;
+---------------+
| name          |
+---------------+
| IVAN BAYROSS  |
| MAMTA MUZUMDAR |
| CHHAYA BANKER  |
| ASHWINI JOSHI  |
| HANSEL COLACO  |
| DIPAK SHARMA   |
+---------------+
6 rows in set (0.00 sec)
```

---

2) Retrieve the contents of the CLIENT_MASTER table.

```
mysql> SELECT * FROM CLIENT_MASTER;
+-----------+---------------+-----------+------------+-----------+---------+-------------+----------+-----------+
| CLIENT_NO | NAME          | ADDRESS1  | ADDRESS2   | CITY      | PINCODE | STATE       |
BAL_DUE  | TELEPHONE |
+-----------+---------------+-----------+------------+-----------+---------+-------------+----------+-----------+
| C00001    | IVAN BAYROSS  | NASHIK    | YEOLA      | MUMBAI    |  400054 | MAHARASHTRA |
15000.00 |      NULL |
| C00002    | MAMTA MUZUMDAR | NIMANI    | CBS        | MADRAS    |  780001 | TAMIL NADU  |
0.00 |      NULL |
| C00003    | CHHAYA BANKER  | DWARKA    | ASHOKNAGAR | MUMBAI    |  400057 |
MAHARASHTRA |  5000.00 |      NULL |
| C00004    | ASHWINI JOSHI  | RK        | SHALIMAR   | BANGALORE |  560001 | KARNATAKA   |
0.00 |      NULL |
| C00005    | HANSEL COLACO  | KOPARGAON | LASLGAON   | MUMBAI    |  400060 |
MAHARASHTRA |  2000.00 |      NULL |
| C00006    | DIPAK SHARMA   | VAIJAPUR  | VINCHUR    | BANGALORE |  560050 | KARNATAKA   |
0.00 |      NULL |
+-----------+---------------+-----------+------------+-----------+---------+-------------+----------+-----------+
6 rows in set (0.00 sec)
```

---

3) Retrieve the list of names, city, and the sates of the clients.

```
mysql> SELECT NAME,CITY,STATE FROM CLIENT_MASTER;
+---------------+-----------+-------------+
| NAME          | CITY      | STATE       |
+---------------+-----------+-------------+
| IVAN BAYROSS  | MUMBAI    | MAHARASHTRA |
| MAMTA MUZUMDAR | MADRAS   | TAMIL NADU  |
| CHHAYA BANKER | MUMBAI    | MAHARASHTRA |
| ASHWINI JOSHI | BANGALORE | KARNATAKA   |
| HANSEL COLACO | MUMBAI    | MAHARASHTRA |
| DIPAK SHARMA  | BANGALORE | KARNATAKA   |
+---------------+-----------+-------------+
6 rows in set (0.00 sec)
```

---

4) List the various products available from the Product_Master tabel.

```
mysql> SELECT DESCRIPTION FROM PRODUCT_MASTER;
+--------------+
| DESCRIPTION  |
+--------------+
| T-SHIRTS     |
| SHIRTS       |
| COTTON JEANS |
| JEANS        |
| TROUSERS     |
| PULL OVERS   |
| DEIM SHIRTS  |
| LYCRA TOPS   |
| SKIRTS       |
+--------------+
9 rows in set (0.01 sec)
```

---

5) List all the Client who are located in Mumbai.

```
mysql> SELECT * FROM CLIENT_MASTER WHERE CITY='MUMBAI';
+-----------+---------------+-----------+------------+--------+---------+-------------+----------+-----------+
| CLIENT_NO | NAME          | ADDRESS1  | ADDRESS2   | CITY   | PINCODE | STATE       | BAL_DUE
| TELEPHONE |
+-----------+---------------+-----------+------------+--------+---------+-------------+----------+-----------+
| C00001    | IVAN BAYROSS  | NASHIK    | YEOLA      | MUMBAI |  400054 | MAHARASHTRA |
15000.00 |      NULL |
| C00003    | CHHAYA BANKER | DWARKA    | ASHOKNAGAR | MUMBAI |  400057 | MAHARASHTRA
|  5000.00 |      NULL |
| C00005    | HANSEL COLACO | KOPARGAON | LASLGAON   | MUMBAI |  400060 |
MAHARASHTRA |  2000.00 |      NULL |
+-----------+---------------+-----------+------------+--------+---------+-------------+----------+-----------+
3 rows in set (0.00 sec)
```

---

6) Find the names of salesman who have a salary equal to Rs. 3000.

```
mysql> SELECT SALESMAN_NAME FROM Sman_mast WHERE SAL_AMT = '3000';
+---------------+
| SALESMAN_NAME |
+---------------+
| AMAN          |
| OMKAR         |
| RAJ           |
| ASHISH        |
+---------------+
4 rows in set (0.00 sec)
```

****

B) Exercise on updating records in a table.

****

1) Change the city of ClientNo C00005 to 'Bangalore'.

```
mysql> UPDATE CLIENT_MASTER SET CITY='BANGLORE' WHERE CLIENT_NO='C00005';
Query OK, 1 row affected (0.06 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT CLIENT_NO,CITY FROM CLIENT_MASTER WHERE CITY='BANGLORE';
+-----------+----------+
| CLIENT_NO | CITY     |
+-----------+----------+
| C00005    | BANGLORE |
+-----------+----------+
1 row in set (0.00 sec)
```

---

2) Change the BalDue of ClientNo C00001 to Rs.1000.

```
mysql> UPDATE CLIENT_MASTER SET BAL_DUE='1000' WHERE CLIENT_NO='C00001';
Query OK, 1 row affected (0.08 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT CLIENT_NO,BAL_DUE FROM CLIENT_MASTER WHERE BAL_DUE='1000';
+-----------+---------+
| CLIENT_NO | BAL_DUE |
+-----------+---------+
| C00001    | 1000.00 |
+-----------+---------+
1 row in set (0.00 sec)
```

---

3) Change the Cost Price of Trouser to Rs.950.00.

```
mysql> UPDATE PRODUCT_MASTER SET COST_PRICE = '950.00' WHERE
DESCRIPTION='TROUSERS';
Query OK, 1 row affected (0.06 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> SELECT DESCRIPTION,COST_PRICE FROM PRODUCT_MASTER WHERE
DESCRIPTION='TROUSERS';
+-------------+------------+
| DESCRIPTION | COST_PRICE |
+-------------+------------+
| TROUSERS    |     950.00 |
+-------------+------------+
1 row in set (0.00 sec)
```

---

4) Change the city of the salesman to Pune.

```
mysql> UPDATE Sman_mast SET CITY='PUNE';
Query OK, 4 rows affected (0.09 sec)
Rows matched: 4  Changed: 4  Warnings: 0

mysql> SELECT CITY FROM Sman_mast;
+------+
| CITY |
+------+
| PUNE |
| PUNE |
| PUNE |
| PUNE |
+------+
4 rows in set (0.01 sec)
```

****

C) Exercise on deleting records in table.

****

1) Delete all salesman from the salesman_master where salary are equal to Rs. 3500.

```
mysql> DELETE FROM Sman_mast  WHERE SAL_AMT=3500;
Query OK, 0 rows affected (0.02 sec)

mysql> SELECT * FROM Sman_mast  WHERE SAL_AMT=3500;
Empty set (0.00 sec)
```

---

2) Delete all Products from Product_Master where qty_on_hand is equal to 100.

```
mysql> DELETE FROM PRODUCT_MASTER  WHERE QTY_ON_HAND=100;
```

---

3) Delete from Client_Master where the cloumn satet holds the value 'Tamil Nadu'.

```
DELETE FROM CLIENT_MASTER  WHERE CITY='TAMILNADU';
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM CLIENT_MASTER  WHERE CITY='TAMILNADU';
Empty set (0.00 sec)
```

****

D) Perform the following computations on the table data.

****

1) List the name of all client having 'a' as the second letter in their names.

```
mysql> SELECT NAME FROM CLIENT_MASTER WHERE NAME LIKE'_a%';
+----------------+
| NAME           |
+----------------+
| MAMTA MUZUMDAR |
| HANSEL COLACO  |
+----------------+
2 rows in set (0.00 sec)
```

---

2) List the client who stay in in the city whose first letter is 'M'.

```
mysql> SELECT NAME,CITY FROM CLIENT_MASTER WHERE CITY LIKE'M%';
+----------------+--------+
| NAME           | CITY   |
+----------------+--------+
| IVAN BAYROSS   | MUMBAI |
| MAMTA MUZUMDAR | MADRAS |
| CHHAYA BANKER  | MUMBAI |
+----------------+--------+
3 rows in set (0.00 sec)
```

---

3) List all clients who stays in 'Banglore' or 'Manglore'.

```
mysql> SELECT NAME,CITY FROM CLIENT_MASTER WHERE CITY LIKE'BANGLORE' OR CITY
LIKE'MANGALORE';

+---------------+----------+
| NAME          | CITY     |
+---------------+----------+
| HANSEL COLACO | BANGLORE |
+---------------+----------+
1 row in set (0.00 sec)
```

---

4) List all the clients whose Bal_due is greater than value 10000.

```
mysql> SELECT * FROM CLIENT_MASTER WHERE BAL_DUE > 10000;
Empty set (0.00 sec)
```

---

5) List all information from the Sales_Order table for Orders placed in month of june.

```
mysql> SELECT * FROM SALES_ORDER WHERE ORDER_DATE LIKE '_____06%' ;
+----------+-----------+------------+-----------+-------------+-----------+--------+------------+--------------+
| ORDER_NO | CLIENT_NO | ORDER_DATE | DELY_ADDR | SALESMAN_NO | DELY_TYPE |
BILLYN | DELY_DATE  | ORDER_STATUS |
+----------+-----------+------------+-----------+-------------+-----------+--------+------------+--------------+
```

```
| O19001   | C00001    | 2004-06-12 | NASHIK    | S00001      | F         | N      | 2002-07-20 | In Process
|
| O19002   | C00002    | 2004-06-25 | NASHIK    | S00002      | P         | N      | 2002-06-27 | Cancelled
|
+----------+-----------+------------+-----------+-------------+-----------+--------+------------+--------------+
2 rows in set, 1 warning (0.00 sec)
```

---

6) List the order information for ClientNO 'C00001' and 'C00002'.

```
mysql> SELECT * FROM SALES_ORDER WHERE CLIENT_NO IN('C00001','C00002');
+----------+-----------+------------+-----------+-------------+-----------+--------+------------+--------------+
| ORDER_NO | CLIENT_NO | ORDER_DATE | DELY_ADDR | SALESMAN_NO | DELY_TYPE |
BILLYN | DELY_DATE  | ORDER_STATUS |
+----------+-----------+------------+-----------+-------------+-----------+--------+------------+--------------+
| O19001   | C00001    | 2004-06-12 | NASHIK    | S00001      | F         | N      | 2002-07-20 | In Process
|
| O19002   | C00002    | 2004-06-25 | NASHIK    | S00002      | P         | N      | 2002-06-27 | Cancelled
|
| O19003   | C00001    | 2004-04-03 | NASHIK    | S00001      | F         | Y      | 2002-04-07 | Fulfilled    |
+----------+-----------+------------+-----------+-------------+-----------+--------+------------+--------------+
3 rows in set (0.00 sec)
```

---

7) List the products whose selling price is greater than 500 and less than or equal to 750.

```
mysql> SELECT * FROM PRODUCT_MASTER WHERE SELL_PRICE>500 AND SELL_PRICE<=750;

+------------+--------------+-----------------+-------------+-------------+-------------+------------+------------+
| PRODUCT_NO | DESCRIPTION  | PROFIT_PERSCENT | UNIT_MEASURE | QTY_ON_HAND |
REORDER_LVL | SELL_PRICE | COST_PRICE |
+------------+--------------+-----------------+-------------+-------------+-------------+------------+------------+
| P06734     | COTTON JEANS |            5.00 | PIECE       |         100 |          20 |     600.00 |     450.00 |
| P07865     | JEANS        |            5.00 | PIECE       |         100 |          20 |     750.00 |     500.00 |
| P07885     | PULL OVERS   |            2.50 | PIECE       |          80 |          30 |     700.00 |     450.00 |
+------------+--------------+-----------------+-------------+-------------+-------------+------------+------------+
3 rows in set (0.00 sec)
```

---

8) List products whose selling price is more than 500 and less than or equal to 750.

```
mysql> SELECT * FROM PRODUCT_MASTER WHERE SELL_PRICE>500 AND SELL_PRICE<=750;

+------------+--------------+-----------------+-------------+-------------+-------------+------------+------------+
| PRODUCT_NO | DESCRIPTION  | PROFIT_PERSCENT | UNIT_MEASURE | QTY_ON_HAND |
REORDER_LVL | SELL_PRICE | COST_PRICE |
+------------+--------------+-----------------+-------------+-------------+-------------+------------+------------+
| P06734     | COTTON JEANS |            5.00 | PIECE       |         100 |          20 |     600.00 |     450.00 |
| P07865     | JEANS        |            5.00 | PIECE       |         100 |          20 |     750.00 |     500.00 |
| P07885     | PULL OVERS   |            2.50 | PIECE       |          80 |          30 |     700.00 |     450.00 |
+------------+--------------+-----------------+-------------+-------------+-------------+------------+------------+
3 rows in set (0.00 sec)
```

---

9) List products whose selling price is more than 500., Calculate a new selling price

as,
original selling price *.15. Rename the new column in the output of the above query as new_price.

```
mysql> SELECT DESCRIPTION, SELL_PRICE*0.15 “NEW_PRICE” FROM PRODUCT_MASTER
WHERE SELL_PRICE > 500;
+--------------+-----------------+
| DESCRIPTION  | “NEW_PRICE”     |
+--------------+-----------------+
| COTTON JEANS |          90.00 |
| JEANS        |         112.50 |
| TROUSERS     |         127.50 |
| PULL OVERS   |         105.00 |
+--------------+-----------------+
4 rows in set (0.00 sec)
```

---

10) List the Names, City, and State of clients who are not in the state of 'Maharashtra'.

```
mysql> SELECT NAME,CITY,STATE FROM CLIENT_MASTER WHERE STATE<> 'MAHARASHTRA';
+----------------+-----------+------------+
| NAME           | CITY      | STATE      |
+----------------+-----------+------------+
| MAMTA MUZUMDAR | MADRAS    | TAMIL NADU |
| ASHWINI JOSHI  | BANGALORE | KARNATAKA  |
| DIPAK SHARMA   | BANGALORE | KARNATAKA  |
+----------------+-----------+------------+
3 rows in set (0.00 sec)
```

---

11) Count total number of all the products.

```
mysql> SELECT COUNT(*) "TOTAL_NO_OF_PRODUCTS" FROM PRODUCT_MASTER;
+----------------------+
| TOTAL_NO_OF_PRODUCTS |
+----------------------+
|                    9 |
+----------------------+
1 row in set (0.01 sec)
```

---

12) Calculate the average price of all the products.

```
mysql> SELECT AVG(SELL_PRICE)"AVERAGE_PRICE" FROM PRODUCT_MASTER;
+---------------+
| AVERAGE_PRICE |
+---------------+
|    538.888889 |
+---------------+
1 row in set (0.00 sec)
```

---

13) Determine the maximum and minimum product prices. Rename the output as max_price

and min_price respectively.

```
mysql> SELECT MAX(SELL_PRICE)"MAX_PRICE"  FROM PRODUCT_MASTER;
+-----------+
| MAX_PRICE |
+-----------+
|    850.00 |
+-----------+
1 row in set (0.00 sec)

mysql>  SELECT MIN(SELL_PRICE)"MIN_PRICE" FROM PRODUCT_MASTER;
+-----------+
| MIN_PRICE |
+-----------+
|    300.00 |
+-----------+
1 row in set (0.00 sec)
```

---

14) Count the number of products having price less than or equal to 500.

```
mysql> SELECT COUNT(SELL_PRICE) FROM PRODUCT_MASTER WHERE SELL_PRICE<=500;
+-------------------+
| COUNT(SELL_PRICE) |
+-------------------+
|                 5 |
+-------------------+
1 row in set (0.00 sec)
```

---

15) List all the products whose QtyonHand is less than reorder level.

```
mysql> SELECT DESCRIPTION FROM PRODUCT_MASTER WHERE
QTY_ON_HAND<REORDER_LVL;
Empty set (0.00 sec)
```

****

E) Exercise on Date Manupulation.

****

1) List the order number and day on which clients placed their order.

```
mysql> SELECT ORDER_NO,DAYNAME(ORDER_DATE) FROM SALES_ORDER;

+----------+--------------------+
| ORDER_NO | DAYNAME(ORDER_DATE) |
+----------+--------------------+
| O19001   | Saturday           |
| O19002   | Friday             |
| O19003   | Saturday           |
| O19008   | Monday             |
| O46865   | Wednesday          |
| O46866   | Thursday           |
+----------+--------------------+
6 rows in set (0.02 sec)
```

---

2) List the month(in alphabhets) and date when the orders must be delivered.

```
mysql> SELECT ORDER_NO,MONTHNAME(DELY_DATE),DAY(DELY_DATE) FROM
SALES_ORDER;
+----------+---------------------+---------------+
| ORDER_NO | MONTHNAME(DELY_DATE) | DAY(DELY_DATE) |
+----------+---------------------+---------------+
| O19001   | July                |            20 |
| O19002   | June                |            27 |
| O19003   | April               |             7 |
| O19008   | July                |            26 |
| O46865   | February            |            20 |
| O46866   | May                 |            22 |
+----------+---------------------+---------------+
6 rows in set (0.00 sec)
```

---

3) List the OrderDate in the format 'DD-MM-YY'. Eg. 12-February-02.

```
mysql> SELECT ORDER_NO,DATE_FORMAT(ORDER_DATE,'%d-%M-%y') FROM SALES_ORDER;
+----------+-----------------------------------+
| ORDER_NO | DATE_FORMAT(ORDER_DATE,'%d-%M-%y') |
+----------+-----------------------------------+
| O19001   | 12-June-04                        |
| O19002   | 25-June-04                        |
| O19003   | 03-April-04                       |
| O19008   | 24-May-04                         |
| O46865   | 18-February-04                    |
| O46866   | 20-May-04                         |
+----------+-----------------------------------+
6 rows in set (0.00 sec)
```

---

4) List the date, 15 days after today's date.

```
mysql> SELECT CURDATE();
+------------+
| CURDATE()  |
+------------+
| 2017-09-05 |
+------------+
1 row in set (0.00 sec)

mysql> SELECT DATE_ADD(CURDATE(),INTERVAL 15 DAY);
+-------------------------------------+
| DATE_ADD(CURDATE(),INTERVAL 15 DAY) |
+-------------------------------------+
| 2017-09-20                          |
+-------------------------------------+
1 row in set (0.00 sec)
```

****