QUERY 01 : Create a table named SALES_Rollno for eg: SALES_12co56 and insert the data in the table as given above….

```
MariaDB [21co56]> CREATE TABLE SALES_21co56
    ->              (
    ->                      order_no int(5) PRIMARY KEY,
    ->                      purchase_amt float,
    ->                      order_date date,
    ->                      customer_id int(5),
    ->                      salesman_id int(5)
    ->              );
Query OK, 0 rows affected (0.019 sec)
```

QUERY 02 : Display the contents of the SALES_21co56

```
MariaDB [21co56]> INSERT INTO SALES_21co56 VALUES
    ->              (70001, 150.5, "2012-10-05", 3005, 5002),
    ->              (70009, 270.65, "2012-09-10", 3001, 5005),
    ->              (70002, 65.26, "2012-10-05", 3002, 5001),
    ->              (70004, 110.5, "2012-08-17", 3009, 5003),
    ->              (70007, 948.5, "2012-09-10", 3005, 5002),
    ->              (70005, 2400.6, "2012-07-27", 3007, 5001),
    ->              (70008, 5760, "2012-09-10", 3002, 5001),
    ->              (70010, 1983.43, "2012-10-10", 3004, 5006),
    ->              (70003, 2480.4, "2012-10-10", 3009, 5003),
    ->              (70012, 250.45, "2012-06-27", 3008, 5002),
    ->              (70011, 75.29, "2012-08-17", 3003, 5007),
    ->              (70013, 3045.6, "2012-04-25", 3002, 5001);
Query OK, 12 rows affected (0.010 sec)
Records: 12  Duplicates: 0  Warnings: 0

MariaDB [21co56]> DESCRIBE SALES_21co56;
+--------------+---------+------+-----+---------+-------+
| Field        | Type    | Null | Key | Default | Extra |
+--------------+---------+------+-----+---------+-------+
| order_no     | int(5)  | NO   | PRI | NULL    |       |
| purchase_amt | float   | YES  |     | NULL    |       |
| order_date   | date    | YES  |     | NULL    |       |
| customer_id  | int(5)  | YES  |     | NULL    |       |
| salesman_id  | int(5)  | YES  |     | NULL    |       |
+--------------+---------+------+-----+---------+-------+
5 rows in set (0.019 sec)

MariaDB [21co56]> SELECT * FROM SALES_21co56;
+----------+--------------+------------+-------------+-------------+
| order_no | purchase_amt | order_date | customer_id | salesman_id |
+----------+--------------+------------+-------------+-------------+
|    70001 |        150.5 | 2012-10-05 |        3005 |        5002 |
|    70002 |        65.26 | 2012-10-05 |        3002 |        5001 |
|    70003 |       2480.4 | 2012-10-10 |        3009 |        5003 |
|    70004 |        110.5 | 2012-08-17 |        3009 |        5003 |
|    70005 |       2400.6 | 2012-07-27 |        3007 |        5001 |
|    70007 |        948.5 | 2012-09-10 |        3005 |        5002 |
|    70008 |         5760 | 2012-09-10 |        3002 |        5001 |
|    70009 |       270.65 | 2012-09-10 |        3001 |        5005 |
|    70010 |      1983.43 | 2012-10-10 |        3004 |        5006 |
|    70011 |        75.29 | 2012-08-17 |        3003 |        5007 |
|    70012 |       250.45 | 2012-06-27 |        3008 |        5002 |
|    70013 |       3045.6 | 2012-04-25 |        3002 |        5001 |
+----------+--------------+------------+-------------+-------------+
12 rows in set (0.001 sec)
```

QUERY 03 : Write a SQL statement to find the total purchase amount for all orders.

```
MariaDB [21co56]> SELECT sum(purchase_amt) FROM SALES_21co56 AS total_p_amt;
+--------------------+
| sum(purchase_amt)  |
+--------------------+
| 17541.180145263672 |
+--------------------+
1 row in set (0.001 sec)
```

QUERY 04 : Write a SQL statement to find the average purchase amount of all orders.

```
MariaDB [21co56]> SELECT avg(purchase_amt) FROM SALES_21co56 AS average;
+--------------------+
| avg(purchase_amt) |
+--------------------+
| 1461.765012105306 |
+--------------------+
1 row in set (0.001 sec)
```

QUERY 05 : Write a SQL statement to find the number of salesmen currently listing for all of their customers.

```
MariaDB [21co56]> SELECT COUNT(DISTINCT salesman_id) AS "no_salesman"
    ->            FROM sales_21co56;
+-------------+
| no_salesman |
+-------------+
|           6 |
+-------------+
1 row in set (0.001 sec)
```

QUERY 06 : Write a SQL statement to get the maximum purchase amount of all the orders

```
MariaDB [21co56]> SELECT max(purchase_amt) FROM sales_21co56;
+-------------------+
| max(purchase_amt) |
+-------------------+
|              5760 |
+-------------------+
1 row in set (0.001 sec)
```

QUERY 07 : Write a SQL statement to get the minimum purchase amount of all the orders.

```
MariaDB [21co56]> SELECT min(purchase_amt) FROM sales_21co56;
+-------------------+
| min(purchase_amt) |
+-------------------+
|             65.26 |
+-------------------+
1 row in set (0.001 sec)
```

QUERY 08 : Find the highest purchase amount ordered by each customer

```
MariaDB [21co56]> SELECT customer_id, max(purchase_amt) FROM sales_21co56
    ->            GROUP by customer_id;
+-------------+-------------------+
| customer_id | max(purchase_amt) |
+-------------+-------------------+
|        3001 |            270.65 |
|        3002 |              5760 |
|        3003 |             75.29 |
|        3004 |           1983.43 |
|        3005 |             948.5 |
|        3007 |            2400.6 |
|        3008 |            250.45 |
|        3009 |            2480.4 |
+-------------+-------------------+
8 rows in set (0.001 sec)
```

QUERY 09 : Write a SQL statement to find the highest purchase amount ordered by the each customer on a particular date with their ID, order date and highest purchase amount.

```
MariaDB [21co56]> SELECT customer_id, order_date, max(purchase_amt)
    ->            FROM sales_21co56
    ->            GROUP BY customer_id, order_date;
+-------------+------------+-------------------+
| customer_id | order_date | max(purchase_amt) |
+-------------+------------+-------------------+
|        3001 | 2012-09-10 |            270.65 |
|        3002 | 2012-04-25 |            3045.6 |
|        3002 | 2012-09-10 |              5760 |
|        3002 | 2012-10-05 |             65.26 |
|        3003 | 2012-08-17 |             75.29 |
|        3004 | 2012-10-10 |           1983.43 |
|        3005 | 2012-09-10 |             948.5 |
|        3005 | 2012-10-05 |             150.5 |
|        3007 | 2012-07-27 |            2400.6 |
|        3008 | 2012-06-27 |            250.45 |
|        3009 | 2012-08-17 |             110.5 |
|        3009 | 2012-10-10 |            2480.4 |
+-------------+------------+-------------------+
12 rows in set (0.001 sec)
```

QUERY 10 : Write a SQL statement to find the highest purchase amount on a date 2012-08-17 for each salesman with their ID

```
MariaDB [21co56]> SELECT salesman_id, order_date, max(purchase_amt)
    ->            FROM sales_21co56
    ->            GROUP BY salesman_id, order_date
    ->            HAVING order_date="2012-08-17";
+-------------+------------+-------------------+
| salesman_id | order_date | max(purchase_amt) |
+-------------+------------+-------------------+
|        5003 | 2012-08-17 |             110.5 |
|        5007 | 2012-08-17 |             75.29 |
+-------------+------------+-------------------+
2 rows in set (0.001 sec)
```

QUERY 11 : Write a SQL statement to find the highest purchase amount with their ID and order date, for only those customers who have highest purchase amount in a day is more than 2000

```
MariaDB [21co56]> SELECT customer_id, salesman_id, order_date, max(purchase_amt)
    ->            FROM sales_21co56
    ->            GROUP BY customer_id, salesman_id, order_date
    ->            HAVING max(purchase_amt) > 2000;
+-------------+-------------+------------+--------------------+
| customer_id | salesman_id | order_date | max(purchase_amt) |
+-------------+-------------+------------+--------------------+
|        3002 |        5001 | 2012-04-25 |             3045.6 |
|        3002 |        5001 | 2012-09-10 |               5760 |
|        3007 |        5001 | 2012-07-27 |             2400.6 |
|        3009 |        5003 | 2012-10-10 |             2480.4 |
+-------------+-------------+------------+--------------------+
4 rows in set (0.001 sec)
```

QUERY 12 : Write a SQL statement to find the highest purchase amount with their ID and order date, for those customers who have a higher purchase amount in a day is within the range 2000 and 6000.

```
MariaDB [21co56]> SELECT customer_id, salesman_id, order_date, max(purchase_amt)
    ->            FROM sales_21co56
    ->            GROUP BY customer_id, salesman_id, order_date
    ->            HAVING max(purchase_amt) BETWEEN 2000 AND 6000;
+-------------+-------------+------------+--------------------+
| customer_id | salesman_id | order_date | max(purchase_amt) |
+-------------+-------------+------------+--------------------+
|        3002 |        5001 | 2012-04-25 |             3045.6 |
|        3002 |        5001 | 2012-09-10 |               5760 |
|        3007 |        5001 | 2012-07-27 |             2400.6 |
|        3009 |        5003 | 2012-10-10 |             2480.4 |
+-------------+-------------+------------+--------------------+
4 rows in set (0.001 sec)
```

QUERY 13 : Write a SQL statement to find the highest purchase amount with their ID and order date, for only those customers who have a higher purchase amount in a day is within the list 2000, 3000, 5760 and 6000.

```
MariaDB [21co56]> SELECT customer_id, salesman_id, order_date, max(purchase_amt)
    ->            FROM sales_21co56
    ->            GROUP BY customer_id, salesman_id, order_date
    ->            HAVING max(purchase_amt) IN ("2000", "3000", "5760", "6000");
+-------------+-------------+------------+--------------------+
| customer_id | salesman_id | order_date | max(purchase_amt) |
+-------------+-------------+------------+--------------------+
|        3002 |        5001 | 2012-09-10 |               5760 |
+-------------+-------------+------------+--------------------+
1 row in set (0.001 sec)
```

QUERY 14 : Write a SQL statement that counts all orders for a date August 17th, 2012.

```
MariaDB [21co56]> SELECT order_date, count(order_date)
    ->            FROM sales_21co56
    ->            GROUP BY order_date
    ->            HAVING order_date="2012-08-17";
+------------+--------------------+
| order_date | count(order_date) |
+------------+--------------------+
| 2012-08-17 |                  2 |
+------------+--------------------+
1 row in set (0.001 sec)
```