**Table Name: orders**

mysql> create database orders;

Query OK, 1 row affected (0.02 sec)

mysql> use orders;

Database changed

mysql> create table orders(ord\_no int,purch\_amt float,ord\_date varchar(50),customer\_id int,salesman\_id int);

Query OK, 0 rows affected (0.10 sec)

mysql> desc orders;

+-------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+-------------+-------------+------+-----+---------+-------+

| ord\_no | int | YES | | NULL | |

| purch\_amt | float | YES | | NULL | |

| ord\_date | varchar(50) | YES | | NULL | |

| customer\_id | int | YES | | NULL | |

| salesman\_id | int | YES | | NULL | |

+-------------+-------------+------+-----+---------+-------+

5 rows in set (0.04 sec)

mysql> insert into orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)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 (0.02 sec)

Records: 12 Duplicates: 0 Warnings: 0

mysql> select\* from orders;

+--------+-----------+------------+-------------+-------------+

| ord\_no | purch\_amt | ord\_date | customer\_id | salesman\_id |

+--------+-----------+------------+-------------+-------------+

| 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 |

+--------+-----------+------------+-------------+-------------+

12 rows in set (0.00 sec)

**1.From the following table, write a SQL query to calculate total purchase amount of all orders. Return total purchase amount.**

mysql> select sum(purch\_amt) as total from orders;

+--------------------+

| total |

+--------------------+

| 17541.180145263672 |

+--------------------+

1 row in set (0.00 sec)

**2.From the following table, write a SQL query to calculate the average purchase amount of all orders. Return average purchase amount.**

mysql> select avg(purch\_amt) as total from orders;

+-------------------+

| total |

+-------------------+

| 1461.765012105306 |

+-------------------+

1 row in set (0.00 sec)

**3.From the following table, write a SQL query that counts the** **number of unique salespeople. Return number of salespeople.**

mysql> select count(distinct salesman\_id) from orders;

+-----------------------------+

| count(distinct salesman\_id) |

+-----------------------------+

| 6 |

+-----------------------------+

1 row in set (0.00 sec)

**4.From the following table, write a SQL query to find the maximum purchase amount.**

mysql> select max(purch\_amt) from orders;

+----------------+

| max(purch\_amt) |

+----------------+

| 5760 |

+----------------+

1 row in set (0.00 sec)

**5.From the following table, write a SQL query to find the minimum purchase amount**

mysql> select min(purch\_amt) from orders;

+----------------+

| min(purch\_amt) |

+----------------+

| 65.26 |

+----------------+

1 row in set (0.00 sec)

**6.From the following table, write a SQL query to find the highest purchase amount ordered by each customer. Return customer ID, maximum purchase amount.**

mysql> select max(purch\_amt),customer\_id from orders group by customer\_id ;

+----------------+-------------+

| max(purch\_amt) | customer\_id |

+----------------+-------------+

| 948.5 | 3005 |

| 270.65 | 3001 |

| 5760 | 3002 |

| 2480.4 | 3009 |

| 2400.6 | 3007 |

| 1983.43 | 3004 |

| 250.45 | 3008 |

| 75.29 | 3003 |

+----------------+-------------+

8 rows in set (0.00 sec)

**7.From the following table, write a SQL query to find the highest purchase**

**amount ordered by each customer on a particular date.**

**Return, order date and highest purchase amount.**

mysql> select max(purch\_amt),ord\_date from orders group by ord\_date;

+----------------+------------+

| max(purch\_amt) | ord\_date |

+----------------+------------+

| 150.5 | 2012-10-05 |

| 5760 | 2012-09-10 |

| 110.5 | 2012-08-17 |

| 2400.6 | 2012-07-27 |

| 2480.4 | 2012-10-10 |

| 250.45 | 2012-06-27 |

| 3045.6 | 2012-04-25 |

+----------------+------------+

7 rows in set (0.00 sec)

**8.From the following table, write a SQL query to determine the highest purchase amount made by each salesperson on '2012-08-17'. Return salesperson ID, purchase amount**

mysql> select max(purch\_amt),salesman\_id from orders where ord\_date='2012-08-17' group by salesman\_id;

+----------------+-------------+

| max(purch\_amt) | salesman\_id |

+----------------+-------------+

| 110.5 | 5003 |

| 75.29 | 5007 |

+----------------+-------------+

2 rows in set (0.00 sec)

**9.From the following table, write a SQL query to find the highest order (purchase) amount by each customer on a particular order date. Filter the result by highest order (purchase) amount above 2000.00. Return customer id, order date and maximum purchase amount.**

mysql> select max(purch\_amt),ord\_date,customer\_id from orders where ord\_date='2012-08-17' group by customer\_id order by max(purch\_amt);

+----------------+------------+-------------+

| max(purch\_amt) | ord\_date | customer\_id |

+----------------+------------+-------------+

| 75.29 | 2012-08-17 | 3003 |

| 110.5 | 2012-08-17 | 3009 |

+----------------+------------+-------------+

2 rows in set (0.00 s

**10.From the following table, write a SQL query to find the maximum order (purchase) amount in the range 2000 - 6000 (Begin and end values are included.) by combination of each customer and order date. Return customer id, order date and maximum purchase amount**

mysql> select max(purch\_amt),customer\_id from orders where (customer\_id between 3002 and 3007) group by customer\_id;

+----------------+-------------+

| max(purch\_amt) | customer\_id |

+----------------+-------------+

| 948.5 | 3005 |

| 5760 | 3002 |

| 2400.6 | 3007 |

| 1983.43 | 3004 |

| 75.29 | 3003 |

+----------------+-------------+

5 rows in set (0.00 sec)

**11.From the following table, write a SQL query to find the maximum order (purchase) amount based on the combination of each customer and order date. Filter the rows for maximum order (purchase) amount is either 2000, 3000, 5760, 6000. Return customer id, order date and maximum purchase amount.**

mysql> select max(purch\_amt),customer\_id from orders where (customer\_id between 3002 and 3007) and (purch\_amt)>1000 group by customer\_id;

+----------------+-------------+

| max(purch\_amt) | customer\_id |

+----------------+-------------+

| 2400.6 | 3007 |

| 5760 | 3002 |

| 1983.43 | 3004 |

+----------------+-------------+

3 rows in set (0.00 sec)

**12.From the following table, write a SQL query to determine the maximum order amount for each customer. The customer ID should be in the range 3002 and 3007(Begin and end values are included.). Return customer id and maximum purchase amount.**

select max(purch\_amt),salesman\_id from orders where (salesman\_id between 5003 and 5008) group by salesman\_id;

+----------------+-------------+

| max(purch\_amt) | salesman\_id |

+----------------+-------------+

| 270.65 | 5005 |

| 2480.4 | 5003 |

| 1983.43 | 5006 |

| 75.29 | 5007 |

+----------------+-------------+

4 rows in set (0.00 sec)

**13.From the following table, write a SQL query to find the maximum order (purchase) amount for each customer. The customer ID should be in the range 3002 and 3007(Begin and end values are included.). Filter the rows for maximum order (purchase) amount is higher than 1000. Return customer id and maximum purchase amount.**

mysql> select count(\*) from orders where ord\_date='2012-08-17';

+----------+

| count(\*) |

+----------+

| 2 |

+----------+

1 row in set (0.00 s

**14. From the following table, write a SQL query to determine the maximum order (purchase) amount generated by each salesperson. Filter the rows for the salesperson ID is in the range 5003 and 5008 (Begin and end values are included.). Return salesperson id and maximum purchase amount.**

mysql> select max(purch\_amt),salesman\_id

-> from orders

-> where(salesman\_id between 5003 and 5008)group by salesman\_id;

+----------------+-------------+

| max(purch\_amt) | salesman\_id |

+----------------+-------------+

| 270.65 | 5005 |

| 2480.4 | 5003 |

| 1983.43 | 5006 |

| 75.29 | 5007 |

+----------------+-------------+

4 rows in set (0.01 sec)

**15.From the following table, write a SQL query to count all the orders generated on '2012-08-17'. Return number of orders.**

mysql> select count(\*)

-> from orders

-> where ord\_date='2012-08-17';

+----------+

| count(\*) |

+----------+

| 2 |

+----------+

1 row in set (0.00 sec)

**16.From the following table, write a SQL query to count the number of orders based on the combination of each order date and salesperson. Return order date, salesperson id.**

mysql> select count(ord\_date),salesman\_id,ord\_date

-> from orders

-> group by ord\_date,salesman\_id;

+-----------------+-------------+------------+

| count(ord\_date) | salesman\_id | ord\_date |

+-----------------+-------------+------------+

| 1 | 5002 | 2012-10-05 |

| 1 | 5005 | 2012-09-10 |

| 1 | 5001 | 2012-10-05 |

| 1 | 5003 | 2012-08-17 |

| 1 | 5002 | 2012-09-10 |

| 1 | 5001 | 2012-07-27 |

| 1 | 5001 | 2012-09-10 |

| 1 | 5006 | 2012-10-10 |

| 1 | 5003 | 2012-10-10 |

| 1 | 5002 | 2012-06-27 |

| 1 | 5007 | 2012-08-17 |

| 1 | 5001 | 2012-04-25 |

+-----------------+-------------+------------+

12 rows in set (0.00 sec)

**17. Write a query to display the columns in a specific order,**

**such as order date, salesman ID, order number, and purchase amount**

**for all orders.**

mysql> select ord\_date,salesman\_id,ord\_no,purch\_amt,ord\_no

-> from orders;

+------------+-------------+--------+-----------+--------+

| ord\_date | salesman\_id | ord\_no | purch\_amt | ord\_no |

+------------+-------------+--------+-----------+--------+

| 2012-10-05 | 5002 | 70001 | 150.5 | 70001 |

| 2012-09-10 | 5005 | 70009 | 270.65 | 70009 |

| 2012-10-05 | 5001 | 70002 | 65.26 | 70002 |

| 2012-08-17 | 5003 | 70004 | 110.5 | 70004 |

| 2012-09-10 | 5002 | 70007 | 948.5 | 70007 |

| 2012-07-27 | 5001 | 70005 | 2400.6 | 70005 |

| 2012-09-10 | 5001 | 70008 | 5760 | 70008 |

| 2012-10-10 | 5006 | 70010 | 1983.43 | 70010 |

| 2012-10-10 | 5003 | 70003 | 2480.4 | 70003 |

| 2012-06-27 | 5002 | 70012 | 250.45 | 70012 |

| 2012-08-17 | 5007 | 70011 | 75.29 | 70011 |

| 2012-04-25 | 5001 | 70013 | 3045.6 | 70013 |

+------------+-------------+--------+-----------+--------+

12 rows in set (0.00 sec)

**18. From the following table, write a SQL query to identify the unique salespeople ID. Return salesman\_id.**

mysql> select distinct salesman\_id

-> from orders;

+-------------+

| salesman\_id |

+-------------+

| 5002 |

| 5005 |

| 5001 |

| 5003 |

| 5006 |

| 5007 |

+-------------+

6 rows in set (0.00 sec**)**

**19. From the following table, write a SQL query to find orders that are delivered by a salesperson with ID. 5001. Return ord\_no, ord\_date, purch\_amt.**

mysql> select ord\_date,ord\_no,purch\_amt

-> from orders

-> where salesman\_id=5001;

+------------+--------+-----------+

| ord\_date | ord\_no | purch\_amt |

+------------+--------+-----------+

| 2012-10-05 | 70002 | 65.26 |

| 2012-07-27 | 70005 | 2400.6 |

| 2012-09-10 | 70008 | 5760 |

| 2012-04-25 | 70013 | 3045.6 |

+------------+--------+-----------+

4 rows in set (0.00 sec)

**20. From the following table, write a SQL query to find all the orders. Sort the result-set in ascending order by ord\_no. Return all fields.**

mysql> select\* from orders

-> order by ord\_no;

+--------+-----------+------------+-------------+-------------+

| ord\_no | purch\_amt | ord\_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 (0.00 sec)

**21.From the following table, write a SQL query to find all the orders. Sort the result-set in descending order by ord\_date. Return all fields.**

mysql> select\* from orders

-> order by ord\_date desc;

+--------+-----------+------------+-------------+-------------+

| ord\_no | purch\_amt | ord\_date | customer\_id | salesman\_id |

+--------+-----------+------------+-------------+-------------+

| 70010 | 1983.43 | 2012-10-10 | 3004 | 5006 |

| 70003 | 2480.4 | 2012-10-10 | 3009 | 5003 |

| 70001 | 150.5 | 2012-10-05 | 3005 | 5002 |

| 70002 | 65.26 | 2012-10-05 | 3002 | 5001 |

| 70009 | 270.65 | 2012-09-10 | 3001 | 5005 |

| 70007 | 948.5 | 2012-09-10 | 3005 | 5002 |

| 70008 | 5760 | 2012-09-10 | 3002 | 5001 |

| 70004 | 110.5 | 2012-08-17 | 3009 | 5003 |

| 70011 | 75.29 | 2012-08-17 | 3003 | 5007 |

| 70005 | 2400.6 | 2012-07-27 | 3007 | 5001 |

| 70012 | 250.45 | 2012-06-27 | 3008 | 5002 |

| 70013 | 3045.6 | 2012-04-25 | 3002 | 5001 |

+--------+-----------+------------+-------------+-------------+

12 rows in set (0.00 sec)

**22.From the following table, write a SQL query to find all the orders. Sort the result-set in descending order by ord\_date and purch\_amt. Return all fields.**

mysql> select\* from orders

-> order by ord\_date,purch\_amt desc;

+--------+-----------+------------+-------------+-------------+

| ord\_no | purch\_amt | ord\_date | customer\_id | salesman\_id |

+--------+-----------+------------+-------------+-------------+

| 70013 | 3045.6 | 2012-04-25 | 3002 | 5001 |

| 70012 | 250.45 | 2012-06-27 | 3008 | 5002 |

| 70005 | 2400.6 | 2012-07-27 | 3007 | 5001 |

| 70004 | 110.5 | 2012-08-17 | 3009 | 5003 |

| 70011 | 75.29 | 2012-08-17 | 3003 | 5007 |

| 70008 | 5760 | 2012-09-10 | 3002 | 5001 |

| 70007 | 948.5 | 2012-09-10 | 3005 | 5002 |

| 70009 | 270.65 | 2012-09-10 | 3001 | 5005 |

| 70001 | 150.5 | 2012-10-05 | 3005 | 5002 |

| 70002 | 65.26 | 2012-10-05 | 3002 | 5001 |

| 70003 | 2480.4 | 2012-10-10 | 3009 | 5003 |

| 70010 | 1983.43 | 2012-10-10 | 3004 | 5006 |

+--------+-----------+------------+-------------+-------------+

12 rows in set (0.00 sec)

**23.From the following table, write a SQL query that calculates the maximum purchase amount generated by each salesperson for each order date. Sort the result-set by salesperson id and order date in ascending order. Return salesperson id, order date and maximum purchase amount.**

mysql> select salesman\_id,ord\_date,max(purch\_amt)

-> from orders

-> group by ord\_date,salesman\_id order by salesman\_id,ord\_date;

+-------------+------------+----------------+

| salesman\_id | ord\_date | max(purch\_amt) |

+-------------+------------+----------------+

| 5001 | 2012-04-25 | 3045.6 |

| 5001 | 2012-07-27 | 2400.6 |

| 5001 | 2012-09-10 | 5760 |

| 5001 | 2012-10-05 | 65.26 |

| 5002 | 2012-06-27 | 250.45 |

| 5002 | 2012-09-10 | 948.5 |

| 5002 | 2012-10-05 | 150.5 |

| 5003 | 2012-08-17 | 110.5 |

| 5003 | 2012-10-10 | 2480.4 |

| 5005 | 2012-09-10 | 270.65 |

| 5006 | 2012-10-10 | 1983.43 |

| 5007 | 2012-08-17 | 75.29 |

+-------------+------------+----------------+

12 rows in set (0.00 sec)

**24.From the following table, write a SQL query that counts the unique orders and**

**the highest purchase amount for each customer.**

**Sort the result-set in descending order on 2nd field.**

**Return customer ID, number of distinct orders and**

**highest purchase amount by each customer.**

mysql> SELECT COUNT(\*) AS orders, MAX(purch\_amt) AS max\_purch\_amt, customer\_id FROM orders GROUP BY customer\_id HAVING MAX(purch\_amt) = (SELECT MAX(purch\_amt) FROM orders) ORDER BY max\_purch\_amt DESC;

+--------+---------------+-------------+

| orders | max\_purch\_amt | customer\_id |

+--------+---------------+-------------+

| 3 | 5760 | 3002 |

+--------+---------------+-------------+

1 row in set (0.00 sec)

**25.From the following table, write a SQL query to calculate the**

**summation of purchase amount, total commission (15% for all salespeople)**

**by each order date.**

**Sort the result-set on order date.**

**Return order date, summation of purchase amount and commission.**

mysql> SELECT COUNT(\*) AS `order`, SUM(purch\_amt) AS total\_purch\_amt, customer\_id, MAX(ord\_date) AS latest\_ord\_date FROM orders GROUP BY customer\_id HAVING SUM(purch\_amt) = (SELECT MAX(purch\_amt) FROM orders) ORDER BY total\_purch\_amt DESC;

Empty set (0.00 sec)