

1. From the following table, write a SQL query to find the details of the customers who have a grade value above 100. Return customer_id, cust_name, city, grade, and salesman_id.

Sample table: customer

customer_id	cust_name	city	grade	salesman_id
5001	3002	Nick Rimando	New York	100
5001	3007	Brad Davis	New York	200
5002	3005	Graham Zusi	California	200
5002	3008	Julian Green	London	300
5006	3004	Fabian Johnson	Paris	300
5003	3009	Geoff Cameron	Berlin	100
5007	3003	Jozy Altidor	Moscow	200
5005	3001	Brad Guzan	London	

Query:

mysql> select customer_id, cust_name, city, grade, salesman_id from customer where grade > 100;

customer_id	cust_name	city	grade	salesman_id
3007	Brad Davis	New York	200	5001
3005	Graham Zusi	California	200	5002
3008	Julian Green	London	300	5002
3004	Fabian Johnson	Paris	300	5006
3003	Jozy Altidor	Moscow	200	5007

2. From the following table, write a SQL query to find all the customers in 'New York' city who have a grade value above 100. Return customer_id, cust_name, city, grade, and salesman_id.

Sample table: customer

customer_id	salesman_id	cust_name	city	grade
5001	3002	Nick Rimando	New York	100
5001	3007	Brad Davis	New York	200
5002	3005	Graham Zusi	California	200
5002	3008	Julian Green	London	300
5006	3004	Fabian Johnson	Paris	300
5003	3009	Geoff Cameron	Berlin	100
5007	3003	Jozy Altidor	Moscow	200
5005	3001	Brad Guzan	London	

```
mysql> select * from customer where city='New York' and grade>100;
```

```
+-----+-----+-----+-----+-----+
| customer_id | cust_name | city   | grade | salesman_id |
+-----+-----+-----+-----+-----+
| 3007 | Brad Davis | New York | 200 | 5001 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

3. From the following table, write a SQL query to find the customers who belong to either the city 'New York' or have a grade above 100. Return customer_id, cust_name, city, grade, and salesman_id.

Sample table: customer

customer_id	salesman_id	cust_name	city	grade
5001	3002	Nick Rimando	New York	100

5001	3007	Brad Davis	New York	200	
5002	3005	Graham Zusi	California	200	
5002	3008	Julian Green	London	300	
5006	3004	Fabian Johnson	Paris	300	
5003	3009	Geoff Cameron	Berlin	100	
5007	3003	Jozy Altidor	Moscow	200	
5005	3001	Brad Guzan	London		

```
mysql> select * from customer where city='new york'or grade>100;
```

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3007	Brad Davis	New York	200	5001
3005	Graham Zusi	California	200	5002
3008	Julian Green	London	300	5002
3004	Fabian Johnson	Paris	300	5006
3003	Jozy Altidor	Moscow	200	5007

4. From the following table, write a SQL query to find the customers who belong to either the city 'New York' or not have a grade above 100. Return customer_id, cust_name, city, grade, and salesman_id.

Sample table: customer

customer_id	cust_name	city	grade	salesman_id
5001	3002	Nick Rimando	New York	100
5001	3007	Brad Davis	New York	200
5002	3005	Graham Zusi	California	200
5002	3008	Julian Green	London	300

5006	3004	Fabian Johnson	Paris	300	
5003	3009	Geoff Cameron	Berlin	100	
5007	3003	Jozy Altidor	Moscow	200	
5005	3001	Brad Guzan	London		

mysql> select * from customer where city='new york' or grade<=100;

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3007	Brad Davis	New York	200	5001
3009	Geoff Cameron	Berlin	100	5003

5. From the following table, write a SQL query to find those customers who belong to neither the 'New York' city nor their grade value exceeds 100. Return customer_id, cust_name, city, grade, and salesman_id.

Sample table: customer

customer_id	cust_name	city	grade	salesman_id
5001	3002	Nick Rimando	New York	100
5001	3007	Brad Davis	New York	200
5002	3005	Graham Zusi	California	200
5002	3008	Julian Green	London	300
5006	3004	Fabian Johnson	Paris	300
5003	3009	Geoff Cameron	Berlin	100
5007	3003	Jozy Altidor	Moscow	200

```

3001 | Brad Guzan | London |
5005
mysql> select * from customer where city<>'new york' and grade<=100;
+-----+-----+-----+-----+-----+
| customer_id | cust_name | city | grade | salesman_id |
+-----+-----+-----+-----+-----+
| 3009 | Geoff Cameron | Berlin | 100 | 5003 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

```

6. From the following table, write a SQL query to find details of all order excluding combination of ord_date equal to '2012-09-10' and salesman_id higher than 5005 or purch_amt greater than 1000. Return ord_no, purch_amt, ord_date, customer_id and salesman_id.

Sample table : 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

```

mysql> select * from orders where ord_date<>'2012-09-10' and (salesman_id>5005
or purch_amt >1000);
+-----+-----+-----+-----+-----+
| ord_no | purch_amt | ord_date | customer_id | salesman_id |
+-----+-----+-----+-----+-----+
| 70005 | 2400.6 | 2012-07-27 | 3007 | 5001 |
| 70010 | 1983.43 | 2012-10-10 | 3004 | 5006 |
| 70003 | 2480.4 | 2012-10-10 | 3009 | 5003 |
| 70011 | 75.29 | 2012-08-17 | 3003 | 5007 |
| 70013 | 3045.6 | 2012-04-25 | 3002 | 5001 |
+-----+-----+-----+-----+-----+

```

7. From the following table, write a SQL query to find the details of those salespeople whose commissions range from 0.10 to 0.12. Return salesman_id, name, city, and commission.

Sample table : salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

```
mysql> select * from salesman where commission between 0.10 and 0.12;
```

salesman_id	name	city	commission
5005	Pit Alex	London	0.11
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

8. From the following table, write a SQL query to find details of all order where purchase amount less than 200 or excluding combination of order date greater than or equal to '2012-02-10' and customer ID less than 3009. Return ord_no, purch_amt, ord_date, customer_id and salesman_id.

Sample table : orders

```
mysql> SELECT * FROM orders WHERE(purch_amt<200 OR NOT(ord_date>='2012-02-10' AND customer_id<3009));
```

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
70004	110.5	2012-08-17	3009	5003
70003	2480.4	2012-10-10	3009	5003
70011	75.29	2012-08-17	3003	5007

9. From the following table, write a SQL query to find all orders subject to following conditions. Exclude combination of order date equal to '2012-08-17' or customer ID higher than 3005 and purchase amount less than 1000.

Sample table : 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

```
mysql> select * from orders where not((ord_date='2012-08-17')or (customer_id>3005 and purch_amt<1000));
```

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
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
70013	3045.6	2012-04-25	3002	5001

10. Write a SQL query to display order number, purchase amount, achieved, the unachieved percentage for those order which exceeds the 50% of the target value of 6000.

Sample table: 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

```
mysql> SELECT ord_no, purch_amt (100*purch_amt)/6000 Achieved %,
      (100*(6000-purch_amt)/6000) Unachieved % FROM orders
      WHERE (100*purch_amt)/6000>50;
```

```
+-----+-----+-----+-----+
| ord_no | purch_amt | Achieved % | Unachieved % |
+-----+-----+-----+-----+
| 70008 | 5760 | 96 | 4 |
| 70013 | 3045.6 | 50.76000162760417 | 49.23999837239583 |
+-----+-----+-----+-----+
```

11. From the following table, write a SQL query to find the details of all employees whose last name is 'Dosni' or 'Mardy'. Return emp_idno, emp_fname, emp_lname, and emp_dept.

Sample table : emp_details

EMP_IDNO	EMP_FNAME	EMP_LNAME	EMP_DEPT
127323	Michale	Robbin	57
526689	Carlos	Snares	63
843795	Enric	Dosio	57
328717	Jhon	Snares	63
444527	Joseph	Dosni	47
659831	Zanifer	Emily	47
847674	Kuleswar	Sitaraman	57
748681	Henrey	Gabriel	47
555935	Alex	Manuel	57
539569	George	Mardy	27
733843	Mario	Saule	63
631548	Alan	Snappy	27
839139	Maria	Foster	57


```
mysql> select * from emp_details where emp_lname='Dosni' or
emp_lname='Mardy';
```

emp_idno	emp_fname	emp_lname	emp_dept
444527	Joseph	Dosni	47
539569	George	Mardy	27

12. From the following table, write a SQL query to find the employees who works at depart 47 or 63. Return emp_idno, emp_fname, emp_lname, and emp_dept.

Sample table : emp_details

EMP_IDNO	EMP_FNAME	EMP_LNAME	EMP_DEPT
127323	Michale	Robbin	57
526689	Carlos	Snares	63
843795	Enric	Dosio	57
328717	Jhon	Snares	63
444527	Joseph	Dosni	47
659831	Zanifer	Emily	47
847674	Kuleswar	Sitaraman	57
748681	Henrey	Gabriel	47
555935	Alex	Manuel	57
539569	George	Mardy	27
733843	Mario	Saule	63
631548	Alan	Snappy	27
839139	Maria	Foster	57

```
mysql> select * from emp_details where emp_dept=47 or emp_dept=63;
```

```
+-----+-----+-----+-----+
```

```
| Emp_idno | emp_fname | emp_lname | emp_dept |
```

```
+-----+-----+-----+-----+
```

```
| 526689 | Carlos | snares | 63 |
```

```
| 328717 | Jhon | Snares | 63 |
```

```
| 444527 | Joseph | Dosni | 47 |
```

```
| 659831 | Zanifer | Emily | 47 |
```

```
| 748681 | Henrey | Gabriel | 47 |
```

```
| 733843 | Mario | Saule | 63 |
```

```
+-----+-----+-----+-----+
```