Create table

Vehicle(vid,vname,price,cid,reg_city) ---only vehicles which are sold

Customer(cid,cname,mobile)

Brand(vname,bname) example active---Honda, I10-Hyudai, civic---honda

1. Find all customer names and vname, for all customers who do not buy any vehicle also display brands for which no vehicle is sold.
   ```
   Select cname,vname,null
   From vehicle v right join customer c on v.cid=c.cid
   Where v.vname is null
   union
   Select null,vname,bname
   From vehicle v right join brand b on v.vname=b.vname
   Where v.vname is null;
   ```

2. Find all customer names and vname, and also display customers who do not bought any vehicle.
   ```
   Select v.vname,c.cname
   From vehicle v right join customer c on v.cid=c.cid
   ```

3. Display cname and mobile for all customer who bought activa
   ```
   Select cid,cname ,mobile
   From customer
   Where cid in (select cid
                 From vehicle
                 Where vname='Activa')
   ```
4. Find all customers who do not buy any vehicle
   ```
   Select cname
   From customer c where not exists(select *
        From vehicle v where v.cid=c.cid);
   ```

5. Find all vehicles name, cname and bname of all vehicles with price > 200000
   ```
   Select vname,cname,bname
   From vehicle v,customer c, brand b
   Where v.cid=c.cid and v.vname=b.vname and price > 200000;
   ```

6. Find all brands for which no vehicle is sold
   ```
   Select bname from brand b
   where not exists (select * from vehicle v where v.vname=b.vname)
   ```
7. Find all vehicles which are registered in pune
   ```
   Select * from vehicle where reg_city='Pune';
   ```

   Set operators ----- union, intersect, minus, union all
   But in mysql only union works

Union --- will display common rows only once
Union all ---- will display common rows twice
Intersect ------ will display only common rows
Minus ------- will display only rows available in table 1 and not common rows.

If you want to combine o/p of multiple queries use set operators.
For set operators the number of columns in all queries should be same, corresponding columns data type should match.

If we have following 3 tables

```
mysql> select * from emp_india;
+------+---------+----------+
| id   | name    | location |
+------+---------+----------+
|  103 | Shivesh | India    |
|  104 | Shivali | India    |
+------+---------+----------+
2 rows in set (0.01 sec)

mysql> select * from emp_us;
+------+---------+----------+
| id   | name    | location |
+------+---------+----------+
|  100 | Rajan   | India    |
|  101 | Rajat   | US       |
|  103 | Shivesh | India    |
+------+---------+----------+
3 rows in set (0.00 sec)

mysql> select * from emp_japan
    -> ;
+------+---------+----------+
| id   | name    | location |
+------+---------+----------+
|  103 | Shivesh | India    |
|  104 | Ashish  | Japan    |
+------+---------+----------+
2 rows in set (0.00 sec)
```

To list all employees who works either in India, or in US or in Japan;

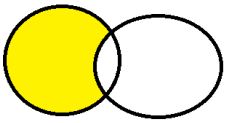
```
mysql> select * from emp_india
    -> union
    -> select * from emp_us
    -> union
    -> select * from emp_japan;
+------+---------+----------+
| id   | name    | location |
+------+---------+----------+
```

```
|  103 | Shivesh | India    |
|  104 | Shivali | India    |
|  100 | Rajan   | India    |
|  101 | Rajat   | US       |
|  104 | Ashish  | Japan    |
+------+---------+----------+
5 rows in set (0.00 sec)

mysql> select * from emp_india
   -> ^C
mysql> select * from emp_us
   -> minus
   -> select * from emp_india;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
corresponds to your MySQL server version for the right syntax to use near 'select * from
emp_india' at line 3
mysql> select * from emp_us
   -> intersect
   -> select * from emp_india;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
corresponds to your MySQL server version for the right syntax to use near 'select * from
emp_india' at line 3
```


A∥ A union B

A intersect B



A minus B

Display all employee names and sum(sal) departmentwise
Select deptno, ename,sal,max(sal) over (partition by deptno)

Display all employee names and sum(sal) departmentwise and arrange data in sorted order within each category.

```
select pid,pname,price,cid,avg(price) over(partition by cid order by price)
   -> from product;From emp;
```