--Create Table Emp with Emp Details

```
CREATE TABLE Emp (

EmpNo int,

EName varchar(20),

Job varchar(20),

MGR int,

HireDate datetime,

Sal int,

Comm int,

DeptNo int)
```

--Insert Data in Emp table

```
INSERT INTO Emp(EmpNo,EName,Job,MGR,HireDate,Sal,Comm,DeptNo)
values(1234,'Amit','Waiter',8382,'2018-10-19',50000,500,50)

INSERT INTO Emp(EmpNo,EName,Job,MGR,HireDate,Sal,Comm,DeptNo)
values(5678,'Ashish','Analyst',8635,'2018-11-02',60000,200,51)

 INSERT INTO Emp(EmpNo,EName,Job,MGR,HireDate,Sal,Comm,DeptNo)
values(8635,'Arun','Analyst',9001,'2016-08-22',80000,400,51)
```

--Create Table Dept with Dept Details

```
CREATE TABLE Dept (

DeptNo int,

Dname varchar(20),

Loc varchar(20))
```

--Insert Data in Dept table

```
INSERT INTO Dept(DeptNo,Dname,Loc) values(50,'Service','Delhi')

INSERT INTO Dept(DeptNo,Dname,Loc) values(51,'Account','Mumbai')
```

--Q1. Show all the data from Emp table.

```
SELECT * FROM Emp
```

--Q2. Show all the data from Dept table.

```
SELECT * FROM Dept
```

--Q3. Display distinct jobs from Emp table.

```
SELECT DISTINCT Job FROM Emp
```

--Q4. Number of employees.

```
SELECT COUNT(EmpNo) FROM Emp
```

--Q5. List the employee in the ascending order of salary.

```
SELECT EName FROM Emp
ORDER BY Sal ASC
```

--Q6. Show the employee information of the Managers.(Managers are also Employees so we need their information)

```
SELECT * FROM Emp
WHERE EmpNo IN (SELECT MGR FROM Emp)
```

--Q7. List of employees who were hired before 2018.

```
SELECT * FROM Emp WHERE HireDate<'2018-01-01'
```

--Q8. List the detail of employees along with the annual salary, order it on the annual salary.

```
SELECT *,
Sal*12 AS AnnualSalary
 FROM Emp
```

```
ORDER BY AnnualSalary
```

--Q9. Display number of months of experience of all the Managers.

```
SELECT *,DATEDIFF(mm,HireDate,GetDate()) AS Experience_Months FROM Emp
WHERE EmpNo IN (SELECT MGR FROM Emp)
```

--Q10. Display the name of the employees with Commission(Comm) less than Salary (Sal).

```
SELECT EName FROM Emp
WHERE Comm<Sal
```

--Q11. Display the name of the employee with Daily income more than 200.

```
SELECT EName,Sal/30 AS Daily_Income FROM Emp
WHERE Sal/30>200
```

--Q12. Show information of all the Waiters.

```
SELECT * FROM Emp
WHERE Job='Waiter'
```

--Q13. Show all the employee who joined on 22-Aug-2016, 4-Aug-2018, 19-Oct-2018 in descending order of Hire Date.

```
SELECT * FROM Emp
WHERE HireDate IN ('2016-08-22','2018-08-04','2018-10-19')
ORDER BY HireDate DESC
```

--Q14. List the employees who joined in 2018.

```
SELECT * FROM Emp
WHERE HireDate BETWEEN '2018-01-01' AND '2018-12-31'
```

--Q15. Employees with Annual Salary between 600000 and 1000000.

```
SELECT * FROM Emp
WHERE Sal*12 BETWEEN 600000 AND 1000000
```

--Q16. List the employees with name starting with A and containing 5 alphabets.

```
SELECT * FROM Emp
WHERE EName LIKE 'A____'
```

--Q17. List the employee with the third alphabet in their name as H.

```
SELECT * FROM Emp
WHERE EName LIKE '__H%'
```

--Q18. Show the name of the employees who joined in August month of any year.

```
SELECT EName FROM Emp
WHERE DATEPART(month,HireDate)=8
```

--Q19. Show the employee details of those who were hired in the 90s.

```
SELECT * FROM Emp
WHERE DATEPART(year,HireDate) BETWEEN 1990 AND 1990
```

--Q20. Show the employee who were not hired in the month of October.

```
SELECT * FROM Emp
WHERE DATEPART(month,HireDate)!=10
```

--Q21. List the total information of the employees along with DName and Location of people working under 'Accounts'.

```
SELECT A.*,
B.Dname,
B.Loc
 FROM Emp AS A
```

```
    INNER JOIN
    Dept AS B
    ON A.DeptNo=B.DeptNo
    WHERE B.Dname='Account'
```

--Q22. List all the employees with more than 3 years of experience as of now.

```
    SELECT * FROM EMP
    WHERE DATEDIFF(YY,HireDate,GetDate())>3
```

--Q23. List the detail of all the employees whose salary is less than that of Arun.

```
    SELECT * FROM EMP
    WHERE Sal < (SELECT Sal FROM EMP WHERE EName='Arun')
```

--Q24. Show the name of those employees who are senior to their own Manager.

```
    SELECT A.EName FROM Emp AS A
  INNER JOIN
  Emp AS B
  ON A.MGR=B.EmpNo
  WHERE A.HireDate<B.HireDate
```

--Q25. Show the employees who are senior to Ashish.

```
    SELECT * FROM Emp
    WHERE HireDate<(SELECT HireDate FROM Emp WHERE EName='Ashish')
```

--Q26. Show the employees who are senior to Ashish and are working in Delhi or Bangalore.

```
    SELECT A.* FROM Emp AS A
  INNER JOIN
  Dept AS B
```

```
ON A.DeptNo=B.DeptNo
WHERE A.HireDate < (SELECT HireDate FROM Emp WHERE EName='Ashish')
AND B.Loc IN ('Delhi','Bangalore')
```

--Q27. Show the employees with the same job as Arun.

```
SELECT * FROM Emp
WHERE Job = (SELECT Job FROM Emp WHERE EName='Arun')
```

--Q28. Find the highest salary of any employee.

```
SELECT MAX(Sal) AS MAX_Sal FROM Emp
```

--Q29. Find the detail of the employee with the minimum pay.

```
SELECT * FROM Emp
WHERE Sal = (SELECT MIN(Sal) FROM Emp)
```

--Q30. Show the detail of the recently hired employee working in Delhi.

```
SELECT A.* FROM Emp AS A
INNER JOIN
 Dept AS B
ON A.DeptNo=B.DeptNo
WHERE B.Loc='Delhi' AND A.HireDate=(SELECT MAX(A.HireDate) FROM Emp AS A
INNER JOIN
Dept AS B
ON A.DeptNo=B.DeptNo
WHERE B.Loc='Delhi')
```

--Q31. How to find the second highest salary in an employee table with employee number and employee salary.

```
SELECT EmpNo, Sal
```

```
FROM Emp
ORDER BY Sal ASC
OFFSET 1 ROW
 FETCH NEXT 1 ROW ONLY
 OR
 SELECT EmpNo, Sal
 FROM Emp AS t1
 WHERE 2 = (SELECT COUNT(DISTINCT Sal) FROM Emp AS t2 WHERE t1.Sal<=t2.Sal)
 OR
 SELECT TOP 1 Sal FROM
 (SELECT DISTINCT TOP 2 Sal FROM Emp ORDER BY Sal  DESC)  ORDER BY Sal
 OR
 With CTE AS (SELECT EmpNo,EName, Sal, Row_Number() OVER(ORDER BY Sal  DESC) as Rank
 From Emp)
 SELECT *  From CTE WHERE Rank = 2
```

--Q32. Write a query to find maximum salary of each Job.

```
SELECT Job, MAX(Sal) AS Max_Sal
FROM Emp
GROUP BY Job
```

--Q33. Write a query to get employees name starting with vowels.

```
SELECT * FROM Emp
WHERE LOWER(EName) LIKE 'a%' OR LOWER(EName) LIKE 'e%' OR LOWER(EName) LIKE 'i%' OR LOWER(EName) LIKE 'o%' OR LOWER(EName) LIKE 'u%'
```

--Q34. Write a query to get employee name starting and ending with vowels.

```
SELECT * FROM Emp
```

```
    WHERE LOWER(EName) LIKE 'a%' OR LOWER(EName) LIKE 'e%' OR LOWER(EName) LIKE 'i%' OR
LOWER(EName) LIKE 'o%' OR LOWER(EName) LIKE 'u%' OR LOWER(EName) LIKE '%a' OR
LOWER(EName) LIKE '%e' OR LOWER(EName) LIKE '%i' OR LOWER(EName) LIKE '%o' OR
LOWER(EName) LIKE '%u'
```

--Q35. Find the count of different Departments in Emp table.

```
    SELECT DeptNo,COUNT(DeptNo) AS Cnt FROM Emp

    GROUP BY DeptNo
```

--Q36. Print the Department occuing more than once in Emp table with their count.

```
    SELECT DeptNo,COUNT(DeptNo) FROM Emp

     GROUP BY DeptNo

     HAVING COUNT(DeptNo)>1
```

--Q37. Select the maximum and second maximum salary from the employee table.

```
    SELECT * FROM EMP

  ORDER BY Sal DESC

  OFFSET 0 ROWS

  FETCH NEXT 2 ROWS ONLY

        OR

  SELECT (SELECT MAX(Sal) FROM Emp) AS Max_Salary,

         (SELECT MAX(Sal) FROM Emp WHERE Sal NOT IN (SELECT MAX(Sal) FROM Emp)) AS
Second_max_salary
```

--Q38. Get all employee detail from Employee table whose “Name” not start with any single character between ‘a-p’.

```
    SELECT * FROM EMP

     WHERE LOWER(EName) LIKE '[^a-p]%'
```

--Q39. Write the query to get the DepartmentName and DepartmentName wise total(sum) Commission, display it in ascending order according to Commission.

```
SELECT B.Dname,
SUM(Comm) AS Total_Comm
FROM Emp AS A
INNER JOIN
Dept AS B
ON A.DeptNo=B.DeptNo
GROUP BY Dname
ORDER BY Total_Comm
```

--Q40. Get department name and average salary of all the employees (department wise).

```
SELECT B.Dname,
  AVG(Sal) AS Avg_Sal
  FROM Emp AS A
INNER JOIN
Dept AS B
  ON A.DeptNo=B.DeptNo
  GROUP BY Dname
```

--Q41. Get department wise number of employees.

```
SELECT B.Dname,
  COUNT(EmpNo) AS No_of_Emp
  FROM Emp AS A
INNER JOIN
Dept AS B
  ON A.DeptNo=B.DeptNo
  GROUP BY Dname
```

--Q42. Write the syntax to create a new column using Row Number over the Salary column.

```
SELECT *,
 ROW_NUMBER() OVER (ORDER BY Sal) AS RwNo
 FROM Emp
```

--Q43. Write the syntax to create a new column using Row Number over the Salary column on the basis of Jobs.

```
SELECT *,
 ROW_NUMBER() OVER (PARTITION BY Job ORDER BY Sal) AS RwNo
FROM Emp
```

--Q44. Write the query to rank the Employees based on Salary.

```
SELECT *,
 RANK() OVER (ORDER BY Sal) AS Rank_Emp
FROM Emp
```

--Q45. Write the query to rank the Employees over the Salary on the basis of Jobs.

```
SELECT *,
RANK() OVER (PARTITION BY Job ORDER BY Sal) AS Rank_Emp
FROM Emp
```

--Q46. Write the query to dense rank the Employees based on Salary.

```
SELECT *,
DENSE_RANK() OVER (ORDER BY Sal) AS Rank_Emp
FROM Emp
```

--Q47. Write the query to dense the Employees over the Salary on the basis of Jobs.

```
SELECT *,
DENSE_RANK() OVER (PARTITION BY Job ORDER BY Sal) AS Rank_Emp
```

```
FROM Emp
```

--Q46. Write the query to use NTILE Function on the Employees based on Salary.

```
SELECT *,
NTILE(2) OVER (ORDER BY Sal) AS Ntile_Emp
FROM Emp
```