SQL Top 10 Interview Queries

First, create two tables for the practice which are Employee Table and Employee Detail Table as below.

EmpID	EmpName	Gender	Salary	City
1	Arjun	M	75000	Pune
2	Ekadanta	M	125000	Bangalore
3	Lalita	F	150000	Mathura
4	Madhav	M	250000	Delhi
5	Visakha	F	120000	Mathura

⟸ Employee Table

EmployeDetail Table ⟹

EmpID	Project	EmpPosition	DOJ	City
1	P1	Executive	26/01/2019	Pune
2	P2	Executive	04/05/2020	Bangalore
3	P1	Lead	21/10/2021	Mathura
4	P3	Manager	29/11/2018	Delhi
5	P2	Manager	01/08/2020	Mathura

Create Tables: Employee and EmployeeDetail

```
CREATE TABLE Employee (
EmpID int NOT NULL,
EmpName Varchar,
Gender Char,
Salary int,
City Char(20) )
--- first run the above code then below code
INSERT INTO Employee
VALUES (1, 'Arjun', 'M', 75000,
'Pune'),
(2, 'Ekadanta', 'M', 125000,
'Bangalore'),
(3, 'Lalita', 'F', 150000 , 'Mathura'),
(4, 'Madhav', 'M', 250000 , 'Delhi'),
(5, 'Visakha', 'F', 120000 , 'Mathura')
```

```
CREATE TABLE EmployeeDetail (
EmpID int NOT NULL,
Project Varchar,
EmpPosition Char(20),
DOJ date )

--- first run the above code then below code
INSERT INTO EmployeeDetail
VALUES (1, 'P1', 'Executive', '26-01-2019'),
(2, 'P2', 'Executive', '04-05-2020'),
(3, 'P1', 'Lead', '21-10-2021'),
(4, 'P3', 'Manager', '29-11-2019'),
(5, 'P2', 'Manager', '01-08-2020')
```

Q1(a): Find the list of employees whose salary ranges between 2L to 3L.

```
select * from employee where salary between 200000 and 300000;
```

Q1(b): Write a query to retrieve the list of employees from the same city.

```
select E1.empname,E1.city from employee E1, employee E2 where E1.city=E2.city and E1.empid
!=E2.empid;
```

Q1(c): Query to find the null values in the Employee table.

```
select * from employee where empid is null;
```

Q2(a): Query to find the cumulative sum of employee's salary.

```
select empid,salary,sum(salary) over  (order by empid) as "cumulative sum" from employee;
```

Q2(b): What's the male and female employees ratio.

```
select (count(*) filter (where gender='M')*100/count(*)) as "Male Ratio",
(count(*) filter (where gender='F')*100/count(*)) as "Female Ratio" from employee;
```

Q2(c): Write a query to fetch 50% records from the Employee table.

```
select * from employee where empid <= (select count(empid)/2 from employee)
```

Q3: Query to fetch the employee's salary but replace the LAST 2 digits with 'XX' i.e., 12345 will be 123XX

```
SELECT Salary, CONCAT(LEFT(CAST(Salary AS text), LENGTH(CAST(Salary AS text))-2), 'XX') AS
masked_number FROM Employee
```

Q4: Write a query to fetch even and odd rows from Employee table.

```
select * from employee where mod(empid,2)=0; Fetch Even rows
select * from employee where mod(empid,2)<>0; Fetch odd rows
```

Q5(a): Write a query to find all the Employee names whose name:

- Begin with ‘A’
- Contains ‘A’ alphabet at second place
- Contains ‘Y’ alphabet at second last place
- Ends with ‘L’ and contains 4 alphabets
- Begins with ‘V’ and ends with ‘A’

```
--begins with “A”
select * from employee where empname like 'A%'

--contains ‘A’ alphabet at second place
select * from employee where empname like '_a%'
--contains ‘Y’ alphabet at second place
select * from employee where empname like '%y_'

---ends with ‘L’ and contains 4 alphabets
select * from employee where empname like '____l'

----begins with ‘V’ and ends with ‘A’.
select * from employee where empname like 'V%a'
```

Q5(b): Write a query to find the list of Employee names which is:

- starting with vowels (a, e, i, o, or u), without duplicates
- ending with vowels (a, e, i, o, or u), without duplicates
- starting & ending with vowels (a, e, i, o, or u), without duplicates

```
---starting with vowels [aeiou] without duplicates
select distinct(empname) from employee where lower(empname)
similar to '[aeiou]%'

---ending with vowels [aeiou] without duplicates
select distinct(empname) from employee where lower(empname)
similar to '%[eioua]'

--starting and ending with vowels [aeiou] without duplicates
select distinct(empname) from employee where lower(empname)
similar to '%[eioua]' and lower(empname) similar to '[aeiou]%'

or,
select distinct(empname) from employee where lower(empname)
similar to '[aeiou]%[eioua]'
```

Q6: Find Nth highest salary from employee table with using the

LIMIT keywords.

```
select salary from employee order by salary desc limit 1 offset 1
```

Q7(a): Write a query to find and remove duplicate records from a table.

```
---finding the duplicate records from table
select empid,empname,gender,salary,city,count(*) as "duplicate count" from employee group by
empid,empname,gender,salary,city having count(*)>1;

---removing duplicate records from table
delete from employee where empid in (select empid from employee   group by empid   having
count(*)>1)
```

Q7(b): Query to retrieve the list of employees working in same project.

```
 select E.empname,E1.project from employee E join employeedetail E1 on E.empid=E1.empid join
employeedetail E2 on E2.project=E1.project where E.empid<>E2.empid
```

Q8: Show the employee with the highest salary for each project.

```
SELECT E1.project,MAX(E.salary) AS "Maximum Project Salary" FROM employee AS E INNER JOIN
employeedetail AS E1 ON E.empid = E1.empid GROUP BY E1.project order by E1.project
```

Q9: Query to find the total count of employees joined each year.

```
SELECT EXTRACT('year' FROM doj) as Year, count(*) FROM employeedetail group by Year order by
Year;
```

Q10: Create 3 groups based on salary col, salary less than 1L is low, between 1 - 2L is medium and above 2L is High.

```
select Salary,
case when Salary >200000 Then 'High' when Salary >=100000 and Salary <=200000 Then 'Medium' else
'Low' end as "Salary Status" from employee;
```

Q11. Query to pivot the data in the Employee table and retrieve the total salary for each city.
The result should display the EmpID, EmpName, and separate columns for each city
(Mathura, Pune, Delhi), containing the corresponding total salary.

```
select empid,empname,
sum(case when city='Mathura' then salary end) as Mathura,
sum(case when city='Pune' then salary end) as Pune,
sum(case when city='Delhi' then salary end) as Delhi,
sum(case when city='Bangalore' then salary end) as Bangalore from employee group by
empid,empname order by empid;
```