Dbms sql queries

**EmployeeInfo Table:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **EmpID** | **EmpFname** | **EmpLname** | **Department** | **Project** | **Address** | **DOB** | **Gender** |
| 1 | Sanjay | Mehra | HR | P1 | Hyderabad(HYD) | 01/12/1976 | M |
| 2 | Ananya | Mishra | Admin | P2 | Delhi(DEL) | 02/05/1968 | F |
| 3 | Rohan | Diwan | Account | P3 | Mumbai(BOM) | 01/01/1980 | M |
| 4 | Sonia | Kulkarni | HR | P1 | Hyderabad(HYD) | 02/05/1992 | F |
| 5 | Ankit | Kapoor | Admin | P2 | Delhi(DEL) | 03/07/1994 | M |

**EmployeePosition Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **EmpID** | **EmpPosition** | **DateOfJoining** | **Salary** |
| 1 | Manager | 01/05/2022 | 500000 |
| 2 | Executive | 02/05/2022 | 75000 |
| 3 | Manager | 01/05/2022 | 90000 |
| 2 | Lead | 02/05/2022 | 85000 |
| 1 | Executive | 01/05/2022 | 300000 |

### ****Q4. Write a query to retrieve the first four characters of  EmpLname from the EmployeeInfo table.****

|  |  |
| --- | --- |
| 1 | **SELECT** SUBSTRING(EmpLname, 1, 4) **FROM** EmployeeInfo; |

### ****Q5. Write a query to fetch only the place name(string before brackets) from the Address column of EmployeeInfo table.****

Using the MID function in [MySQL](https://www.edureka.co/blog/what-is-mysql/)

|  |  |
| --- | --- |
| 1 | **SELECT** MID(Address, 0, LOCATE('(',Address)) **FROM** EmployeeInfo; |

Using SUBSTRING

|  |  |
| --- | --- |
| 1 | **SELECT** SUBSTRING(Address, 1, CHARINDEX('(',Address)) **FROM** EmployeeInfo; |

### ****Q6. Write a query to create a new table which consists of data and structure copied from the other table.****

Using the SELECT INTO command:

|  |  |
| --- | --- |
| 1 | **SELECT** \* **INTO** NewTable **FROM** EmployeeInfo **WHERE** 1 = 0; |

Using the [CREATE command](https://www.edureka.co/blog/create-table-in-sql/) in MySQL:

**CREATE** **TABLE** NewTable **AS** **SELECT** \* **FROM** EmployeeInfo;

### ****Q9.**** Write a query to fetch top N records.

By using the TOP command in SQL Server:

|  |  |
| --- | --- |
| 1 | **SELECT** **TOP** N \* **FROM** EmployeePosition **ORDER** **BY** Salary **DESC**; |

By using the LIMIT command in MySQL:

|  |  |
| --- | --- |
| 1 | **SELECT** \* **FROM** EmpPosition **ORDER** **BY** Salary **DESC** LIMIT N; |

### ****Q10. Write a query to retrieve the EmpFname and EmpLname in a single column as “FullName”. The first name and the last name must be separated with space.****

|  |  |
| --- | --- |
| 1 | **SELECT** CONCAT(EmpFname, ' ', EmpLname) **AS** 'FullName' **FROM** EmployeeInfo; |

### ****Q11. Write a query find number of employees whose DOB is between 02/05/1970 to 31/12/1975 and are grouped according to gender****

|  |  |
| --- | --- |
| 1 | **SELECT** COUNT(\*), Gender **FROM** EmployeeInfo **WHERE** DOB BETWEEN '02/05/1970 ' AND  '31/12/1975' **GROUP** **BY** Gender; |

### ****Q16. Write a query to fetch all employees who also hold the managerial position.****

|  |  |
| --- | --- |
| 1  2  3 | **SELECT** E.EmpFname, E.EmpLname, P.EmpPosition  **FROM** EmployeeInfo E **INNER** JOIN EmployeePosition P **ON**  E.EmpID = P.EmpID AND P.EmpPosition IN ('Manager'); |

### ****Q17.**** Write a query to fetch the department-wise count of employees sorted by department’s count in ascending order.

|  |  |
| --- | --- |
| 1  2  3 | **SELECT** Department, count(EmpID) **AS** EmpDeptCount  **FROM** EmployeeInfo **GROUP** **BY** Department  **ORDER** **BY** EmpDeptCount **ASC**; |

### ****Q18. Write a query to calculate the even and odd records from a table.****

To retrieve the even records from a table, you have to use the MOD() function as follows:

|  |  |
| --- | --- |
| 1 | **SELECT** EmpID **FROM** (**SELECT** rowno, EmpID **from** EmployeeInfo) **WHERE** MOD(rowno,2)=0; |

Similarly, to retrieve the odd records from a table, you can write a query as follows:

|  |  |
| --- | --- |
| 1 | **SELECT** EmpID **FROM** (**SELECT** rowno, EmpID **from** EmployeeInfo) **WHERE** MOD(rowno,2)=1; |

### ****Q19.**** Write a SQL query to retrieve employee details from EmployeeInfo table who have a date of joining in the EmployeePosition table.

|  |  |
| --- | --- |
| 1  2  3 | **SELECT** \* **FROM** EmployeeInfo E  **WHERE** EXISTS  (**SELECT** \* **FROM** EmployeePosition P **WHERE** E.EmpId = P.EmpId); |

### ****Q20. Write a query to retrieve two minimum and maximum salaries from the EmployeePosition table.****

To retrieve two minimum salaries, you can write a query as below:

|  |  |
| --- | --- |
| 1  2  3 | **SELECT** **DISTINCT** Salary **FROM** EmployeePosition E1  **WHERE** 2 >= (SELECTCOUNT(**DISTINCT** Salary)**FROM** EmployeePosition E2  **WHERE** E1.Salary >= E2.Salary) **ORDER** **BY** E1.Salary **DESC**; |

To retrieve two maximum salaries, you can write a query as below:

|  |  |
| --- | --- |
| 1  2  3 | **SELECT** **DISTINCT** Salary **FROM** EmployeePosition E1  **WHERE** 2 >= (SELECTCOUNT(**DISTINCT** Salary) **FROM** EmployeePosition E2  **WHERE** E1.Salary <= E2.Salary) **ORDER** **BY** E1.Salary **DESC**; |

### ****Q21.**** Write a query to find the Nth highest salary from the table without using TOP/limit keyword.

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | **SELECT** Salary  **FROM** EmployeePosition E1  **WHERE** N-1 = (  **SELECT** COUNT( **DISTINCT** ( E2.Salary ) )  **FROM** EmployeePosition E2  **WHERE** E2.Salary >  E1.Salary ); |

### ****Q22. Write a query to retrieve duplicate records from a table.****

|  |  |
| --- | --- |
| 1  2  3 | **SELECT** EmpID, EmpFname, Department COUNT(\*)  **FROM** EmployeeInfo **GROUP** **BY** EmpID, EmpFname, Department  **HAVING** COUNT(\*) > 1; |

### ****Q23. Write a query to retrieve the list of employees working in the same department.****

|  |  |
| --- | --- |
| 1  2  3 | **Select** **DISTINCT** E.EmpID, E.EmpFname, E.Department  **FROM** EmployeeInfo E, Employee E1  **WHERE** E.Department = E1.Department AND E.EmpID != E1.EmpID; |

### ****Q24. Write a query to retrieve the last 3 records from the EmployeeInfo table.****

|  |  |
| --- | --- |
| 1  2  3  4 | **SELECT** \* **FROM** EmployeeInfo **WHERE**  EmpID <=3 **UNION** **SELECT** \* **FROM**  (**SELECT** \* **FROM** EmployeeInfo E **ORDER** **BY** E.EmpID **DESC**)  **AS** E1 **WHERE** E1.EmpID <=3; |

### ****Q25. Write a query to find the third-highest salary from the EmpPosition table.****

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | **SELECT** **TOP** 1 salary  **FROM**(  **SELECT** **TOP** 3 salary  **FROM** employee\_table  **ORDER** **BY** salary **DESC**) **AS** emp  **ORDER** **BY** salary **ASC**; |

### ****Q26. Write a query to display the first and the last record from the EmployeeInfo table.****

To display the first record from the EmployeeInfo table, you can write a query as follows:

|  |  |
| --- | --- |
| 1 | **SELECT** \* **FROM** EmployeeInfo **WHERE** EmpID = (**SELECT** **MIN**(EmpID) **FROM** EmployeeInfo); |

To display the last record from the EmployeeInfo table, you can write a query as follows:

|  |  |
| --- | --- |
| 1 | **SELECT** \* **FROM** EmployeeInfo **WHERE** EmpID = (**SELECT** **MAX**(EmpID) **FROM** EmployeeInfo); |

### ****Q27. Write a query to add email validation to your database****

|  |  |
| --- | --- |
| 1 | **SELECT** Email **FROM** EmployeeInfo **WHERE** NOT REGEXP\_LIKE(Email, ‘[A-Z0-9.\_%+-]+@[A-  Z0-9.-]+.[A-Z]{2,4}’, ‘i’); |

### ****Q28. Write a query to retrieve Departments who have less than 2 employees working in it.****

|  |  |
| --- | --- |
| 1 | **SELECT** DEPARTMENT, COUNT(EmpID) **as** 'EmpNo' **FROM** EmployeeInfo **GROUP** **BY** DEPARTMEN  T **HAVING** COUNT(EmpD) < 2; |

### ****Q29. Write a query to retrieve EmpPostion along with total salaries paid for each of them.****

|  |  |
| --- | --- |
| 1 | **SELECT** EmpPosition, SUM(Salary) **from** EmployeePosition **GROUP** **BY** EmpPosition; |

### ****Q30. Write a query to fetch 50% records from the EmployeeInfo table.****

|  |  |
| --- | --- |
| 1  2  3 | **SELECT** \*  **FROM** EmployeeInfo **WHERE**  EmpID <= (**SELECT** COUNT(EmpID)/2 **from** EmployeeInfo); |