**(Employee Details and Salary)**

**By**

**Sachin Gunjal**

CREATE DATABASE Employee\_Assign2

USE Employee\_Assign2

CREATE TABLE EmployeeDetails (

EmpId INT PRIMARY KEY,

FullName VARCHAR(100),

ManagerId INT,

DateOfJoining DATE,

City VARCHAR(100)

);

CREATE TABLE EmployeeSalary (

EmpId INT,

Project VARCHAR(50),

Salary DECIMAL(10, 2),

Variable DECIMAL(10, 2),

FOREIGN KEY (EmpId) REFERENCES

EmployeeDetails(EmpId)

);

INSERT INTO EmployeeSalary (EmpId, Project, Salary, Variable)

VALUES

(101, 'P1', 7500, 500),

(102, 'P2', 9200, 700),

(103, 'P1', 6700, 600),

(104, 'P3', 8300, 900),

(105, 'P2', 7800, 800),

(106, 'P3', 9100, 1000),

(107, 'P1', 6200, 400),

(108, 'P2', 8800, 750),

(109, 'P3', 9500, 1100),

(110, 'P1', 7200, 650),

(111, 'P2', 8700, 850),

(112, 'P3', 9300, 1200),

(113, 'P1', 7900, 550),

(114, 'P2', 6800, 450),

(115, 'P3', 8400, 900),

(116, 'P1', 7600, 500),

(117, 'P2', 8900, 1000),

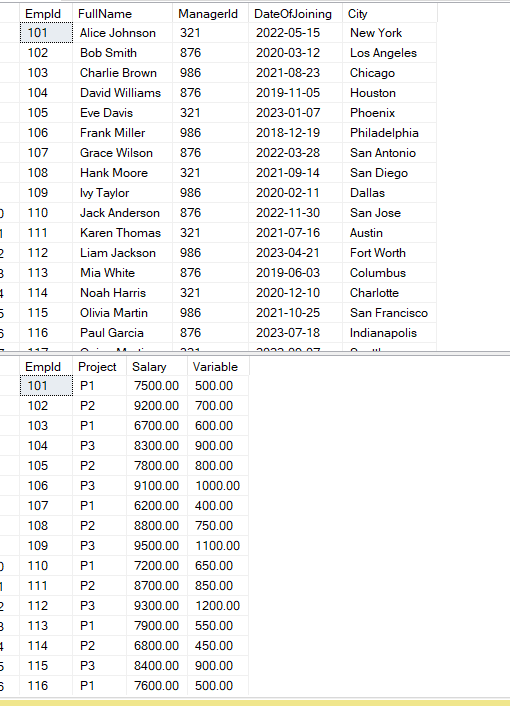
(118, 'P3', 9200, 1100),

(119, 'P1', 8100, 600),

(120, 'P2', 8300, 750);

SELECT \* FROM EmployeeDetails

SELECT \* FROM EmployeeSalary



**PART-1**

1. **SQL Query to fetch records that are present in one table but not in another table.**

SELECT ED.\*

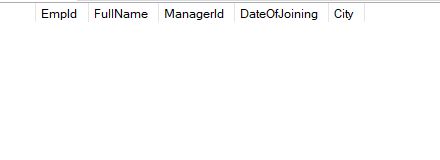
FROM EmployeeDetails ED

LEFT JOIN

EmployeeSalary ES

ON ED.EmpId = ES.EmpId

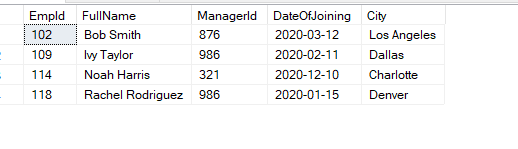
WHERE ES.EmpId IN NULL



**3. SQL query to fetch all the Employees from EmployeeDetails who joined in the Year 2020.**

SELECT \* FROM EmployeeDetails

WHERE YEAR(DateOfJoining) = 2020



**4.Fetch all employees from EmployeeDetails who have a salary record in EmployeeSalary**.

SELECT ED.\*, ES.Salary

FROM EmployeeDetails ED

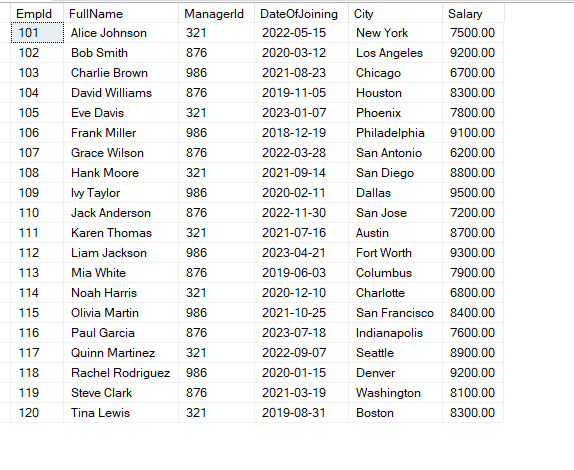
INNER JOIN

EmployeeSalary ES

ON

ED.EmpId = ES.EmpId

WHERE ES.Salary IS NOT NULL

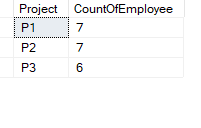


5. **Write an SQL query to fetch a project-wise count of employees.**

SELECT Project, COUNT(\*) as CountOfEmployee

FROM EmployeeSalary

GROUP BY Project



6. **Fetch employee names and salaries even if the salary value is not present for the employee.**

SELECT ED.FullName, ES.Salary

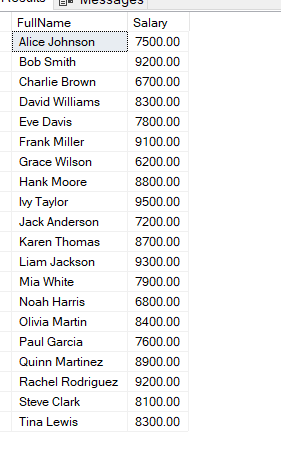
FROM EmployeeDetails ED

LEFT JOIN

EmployeeSalary ES

ON

ED.EmpId = ES.EmpId



7**. Write an SQL query to fetch all the Employees who are also managers.**

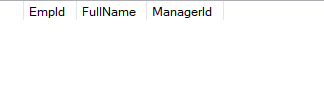
SELECT EmpId, FullName, ManagerId

FROM EmployeeDetails

WHERE EmpId IN (

SELECT ManagerId FROM EmployeeDetails

WHERE ManagerId IS NOT NULL)

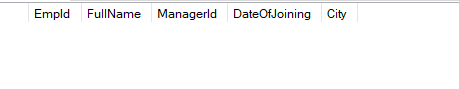


8. **Write an SQL query to fetch duplicate records from EmployeeDetails.**

SELECT \* FROM EmployeeDetails

GROUP BY EmpId, FullName,ManagerId,DateOfJoining,City

HAVING COUNT(\*) > 1



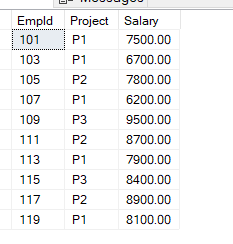
9. **Write an SQL query to fetch only odd rows from the table**.

SELECT E.EmpId , E.Project, E.Salary FROM (

SELECT \*, ROW\_NUMBER() OVER (ORDER BY EmpId) AS RowNum

FROM EmployeeSalary) E

WHERE E.RowNum % 2 = 1



**10.Write a query to find the 3rd highest salary from a table without top or limit keyword.**

SELECT DISTINCT SALARY FROM EmployeeSalary salary1

WHERE 2 = (

SELECT COUNT(DISTINCT SALARY)

FROM EmployeeSalary salary2

WHERE salary2.Salary > salary1.SALARY )

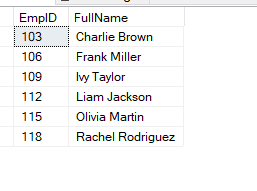


**PART- 2**

**1. Write an SQL query to fetch the EmpId and FullName of all the employees working under the Manager with id – 986.**

SELECT EmpID ,FullName FROM EmployeeDetails

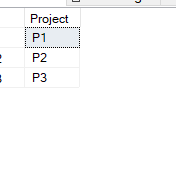
WHERE ManagerID = '986'



**2. Write an SQL query to fetch the different projects available from the EmployeeSalary table.**

SELECT DISTINCT Project FROM EmployeeSalary

GROUP BY Project

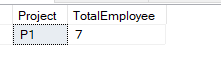


**3. Write an SQL query to fetch the count of employees working in project 'P1'**

SELECT Project, COUNT(\*) As TotalEmployee FROM EmployeeSalary

GROUP BY Project

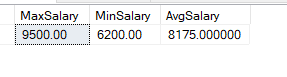
HAVING Project = 'P1'



**4. Write an SQL query to find the maximum, minimum, and average salary of the employees.**

SELECT MAX(Salary) AS MaxSalary, MIN(Salary) AS MinSalary, AVG(Salary) AS AvgSalary

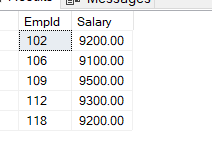
FROM EmployeeSalary



**5. Write an SQL query to find the employee id whose salary lies in the range of 9000 and 15000.**

SELECT EmpId, Salary FROM EmployeeSalary

WHERE Salary BETWEEN 9000 AND 15000



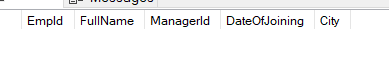
**6. Write an SQL query to fetch those employees who live in Toronto and work under the manager with ManagerId – 321.**

SELECT \* FROM EmployeeDetails

WHERE City = 'Toronto'

AND

ManagerId = 321



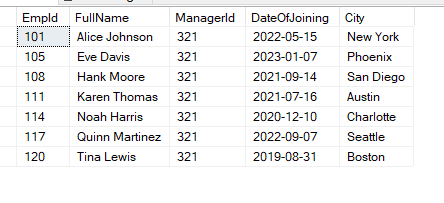
**7. Write an SQL query to fetch all the employees who either live in California or work under a manager with ManagerId – 321.**

SELECT \* FROM EmployeeDetails

WHERE City = 'California'

OR

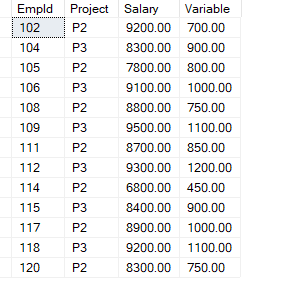
ManagerId = 321



**8. Write an SQL query to fetch all those employees who work on Projects other than P1.**

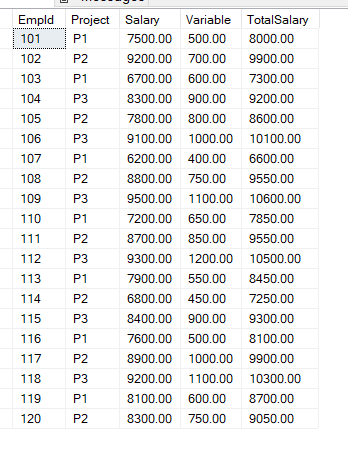
SELECT \* FROM EmployeeSalary

WHERE Project <> 'P1'



**9. Write an SQL query to display the total salary of each employee adding the Salary with Variable value.**

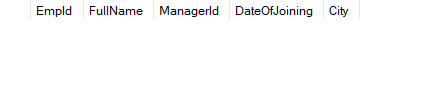
SELECT \*, (Salary+ Variable) AS TotalSalary FROM EmployeeSalary



**10. Write an SQL query to fetch the employees whose name begins with any two characters, followed by a text “hn” and ends with any sequence of characters**

SELECT \* FROM EmployeeDetails

WHERE FullName LIKE '\_\_hn%';



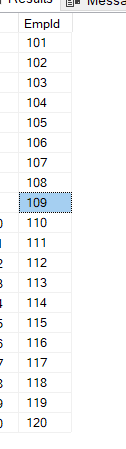
**Part 3**

**1. Write an SQL query to fetch all the EmpIds which are present in either of the tables – EmployeeDetails and „EmployeeSalary‟.**

SELECT EmpId FROM EmployeeDetails

UNION

SELECT EmpId FROM EmployeeSalary

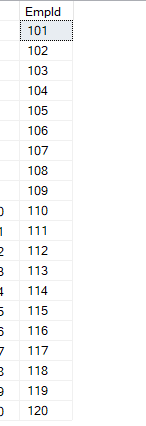


**2. Write an SQL query to fetch common records between two tables.**

SELECT EmpId FROM EmployeeDetails

INTERSECT

SELECT EmpId FROM EmployeeSalary



**3. Write an SQL query to fetch records that are present in one table but not in another table**.

SELECT ED.\* FROM EmployeeDetails ED

LEFT JOIN

EmployeeSalary ES

ON

ED.EmpId = ES.EmpId

WHERE ES.EmpId IS NULL

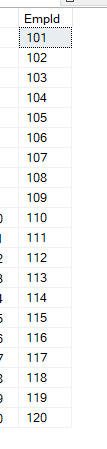


**4. Write an SQL query to fetch the EmpIds that are present in both the tables – „EmployeeDetails‟ and „EmployeeSalary.**

SELECT EmpId FROM EmployeeDetails

INTERSECT

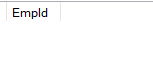
SELECT EmpId FROM EmployeeSalary



**5. Write an SQL query to fetch the EmpIds that are present in EmployeeDetails but not in EmployeeSalary.**

SELECT EmpId FROM EmployeeDetails

WHERE EmpId NOT IN (SELECT EmpId FROM EmployeeSalary)



**6. Write an SQL query to fetch the employee‟s full names and replace the space**

SELECT REPLACE(FullName, ' ','') as ReplacedFullName

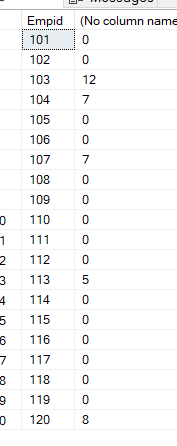
FROM EmployeeDetails



**7. Write an SQL query to fetch the position of a given character(s) in a field**.

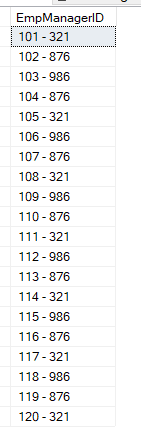
SELECT Empid, CHARINDEX('W', FullName)

FROM EmployeeDetails



**8. Write an SQL query to display both the EmpId and ManagerId together.**

SELECT CONCAT(EmpId,' - ', ManagerId) AS EmpManagerID FROM EmployeeDetails



**9. Write a query to fetch only the first name(string before space) from the FullName column of the EmployeeDetails table.**

SELECT FullName, SUBSTRING(FullName, 1, CHARINDEX(' ', Fullname)-1) as FirstName

FROM EmployeeDetails



**10. Write an SQL query to uppercase the name of the employee and lowercase the city values.**

SELECT UPPER(FullName) AS CapitalizedNames, LOWER(City) AS Lowercased FROM EmployeeDetails



**Part -4**

1. **Write an SQL query to find the count of the total occurrences of a particular character – 'n' in the FullName field.**

SELECT FullName,

LEN(FULLNAME) - LEN(REPLACE(Fullname, 'a', '')) as CharacterCount

FROM EmployeeDetails



**2. Write an SQL query to update the employee names by removing leading and trailing spaces**.

SELECT TRIM(FULLNAME) AS TrimmedSpacesFullName FROM EMPLOYEEDETAILS



**3. Fetch all the employees who are not working on any project.**

SELECT D.\*, S.Project FROM EmployeeDetails D

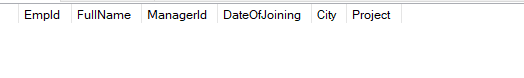
INNER JOIN

EmployeeSalary S

ON

D.EmpId = S.EmpId

WHERE S.Project IS NULL



**4. Write an SQL query to fetch employee names having a salary greater than or equal to 5000 and less than or equal to 10000.**

SELECT D.\*, S.Project, S.Salary FROM EmployeeDetails D

INNER JOIN

EmployeeSalary S

ON

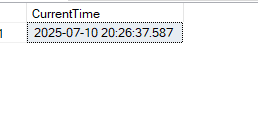
D.EmpId = S.EmpId

WHERE S.Salary BETWEEN 5000 AND 10000



**5. Write an SQL query to find the current date-time.**

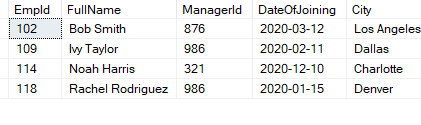
SELECT CURRENT\_TIMESTAMP AS CURRENT\_TIME

****

**6. Write an SQL query to fetch all the Employee details from the EmployeeDetails table who joined in the Year 2020.**

SELECT \* FROM EmployeeDetails

WHERE YEAR(DATEOFJOINING) = 2020



**7. Write an SQL query to fetch all employee records from the EmployeeDetails table who have a salary record in the EmployeeSalary table.**

SELECT D.\*, S.Salary FROM EmployeeDetails D

INNER JOIN

EmployeeSalary S

ON

D.EmpId = S.EmpId

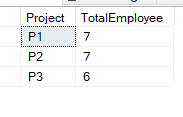


**8. Write an SQL query to fetch the project-wise count of employees sorted by project‟s count in descending order.**

SELECT Project, COUNT(\*) AS TotalEmployee FROM EmployeeSalary

GROUP BY Project

ORDER BY COUNT(\*) DESC



**9. Write a query to fetch employee names and salary records. Display the employee details even if the salary record is not present for the employee.**

SELECT D.\*, S.Salary FROM EmployeeDetails D

LEFT JOIN

EmployeeSalary S

ON

D.EmpId = S.EmpId



**10. Write an SQL query to join 3 tables.**

CREATE TABLE EMPLOYEE(

EMPID INT IDENTITY(101,1) NOT NULL,

FirstName VARCHAR(50),

LastName VARCHAR(50),

Locatin VARCHAR(50)

)

INSERT INTO EMPLOYEE VALUES

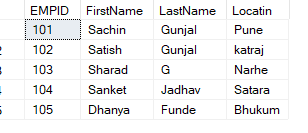
('Sachin','Gunjal','Pune'),

('Satish','Gunjal','katraj'),

('Sharad','G','Narhe'),

('Sanket','Jadhav','Satara'),

('Dhanya','Funde','Bhukum');



SELECT \* FROM EMPLOYEE

SELECT \* from EmployeeDetails

SELECT \* from Employeesalary

SELECT \* FROM EMPLOYEE E

RIGHT JOIN

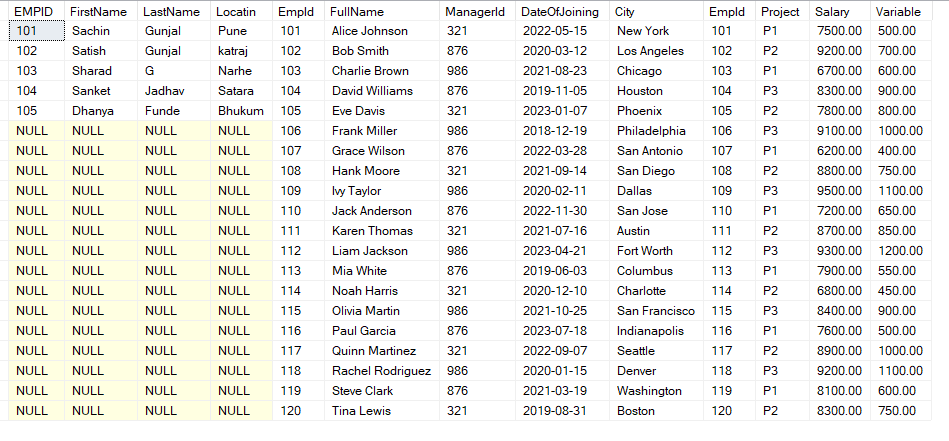
EmployeeDetails ED

ON E.EMPID = ED.EmpId

INNER JOIN

EmployeeSalary ES

ON ED.EmpId = ES.EmpId

****