EMPLOYEE MANAGEMENT SYSTEM

INTRODUCTION: The Employee Management System (EMS) SQL project is designed to streamline and automate the management of employee records within an organization. This system consists of three primary tables: Departments, Employees, and Salaries. Each table plays a vital role in organizing and maintaining the company's employee data.

ABSTRACT: The system is built around three core tables: Departments**,** Employees, and Salaries, each serving a distinct purpose in organizing and maintaining essential employee data. The Departments table categorizes employees into various organizational units, while the Employees table stores personal and professional details such as names, departments, hire dates, and birthdates. The Salaries table records salary amounts and tracks changes over time. This system provides comprehensive functionality for administrative tasks, including querying employee details, calculating employee tenure, and analysing salary distributions.

Problem Statements

1. Retrieve the first and last names of all employees born between 1980 and 1990
2. Find the first and last names of employees who share the same last name as at least one otheremployee.
3. Retrieve the first names of employees who were hired after the youngest employee was born
4. Find the employees with minimum age in company
5. Find the employee who have been with company the longest.
6. List the employees in descending order by their age share only the top 10 oldest.
7. Find the employee with highest salary.
8. Count the number of employees in each department.
9. List al employees, showing their ages calculated by their Date of Birth.
10. Find the employee born on the same date.
11. Retrieve the employee with most recent hire date.
12. Retrieve the last name of employee but with the first character lower
13. Find the employees with the first name longer than their last name.
14. Find average age of employees
15. Count the number of employees hired each year.
16. Find number of employees with same last name.
17. Find employees with salaries that are less than the average salary (68120).
18. Get Employees who do not have salary record.
19. Calculate age and salary of employee older than 30. Query must return first name last name age and salary.
20. Find employees born in the 1990s
21. Get a list of employees along with their hire dates, but only those hired in 2015.
22. Retrieve employee name and their respective salaries. Filtering out those with salaries below 30k.
23. Retrieve all employees and their salaries, show the difference between highest salary in department and each employee's salary.
24. Get a list of all employees and their salaries and identify maximum salary from their respective departments.
25. Find number of employees in each department including department name.
26. Retrieve employee who same hire date as any other employee.
27. List all the employees and their department

- **Query to create a database/schema.**

```
create database EmployeeManagementSystem;
```

- **Query to use the EmployeeManagementSystem database for further operations.**

```
use EmployeeManagementSystem;
```

- **Query to create table named Departments**

```
CREATE TABLE Departments (
   DepartmentID INT PRIMARY KEY,
   DepartmentName VARCHAR(100)
);
```

- **Query to create table named Employees**

```
CREATE TABLE Employees (
   EmployeeID INT PRIMARY KEY,
   FirstName VARCHAR(50),
   LastName VARCHAR(50),
   DepartmentID INT,
   DateOfBirth DATE,
   HireDate DATE,
   FOREIGN KEY (DepartmentID) REFERENCES Departments(DepartmentID)
);
```

- **Query to create table named Salaries**

```
CREATE TABLE Salaries (
   SalaryID INT PRIMARY KEY AUTO_INCREMENT,
   EmployeeID  INT,
   SalaryAmount DECIMAL(10, 2),
```

EffectiveDate DATE,

FOREIGN KEY (EmployeeID) REFERENCES Employees(EmployeeID));

- **Query to insert records into Departments table**

  INSERT INTO Departments (DepartmentID, DepartmentName) VALUES

  (1, 'Human Resources'),

  (2, 'Finance'),

  (3, 'Engineering'),

  (4, 'Marketing'),

  (5, 'Sales'),

  (6, 'IT'),

  (7, 'Customer Service'),

  (8, 'Research and Development'),

  (9, 'Legal'),

  (10, 'Administration');

- **Query to insert records into Employees table**

  INSERT INTO Employees (EmployeeID, FirstName, LastName, DepartmentID, DateOfBirth, HireDate) VALUES

  (1, 'Aarav', 'Sharma', 1, '1985-05-15', '2010-01-10'),

  (2, 'Vivaan', 'Gupta', 2, '1990-06-20', '2015-02-15'),

  (3, 'Aditya', 'Reddy', 3, '1988-07-25', '2012-03-20'),

  (4, 'Ananya', 'Kumar', 4, '1992-08-30', '2016-04-18'),

  (5, 'Kabir', 'Mehta', 5, '1987-09-10', '2011-05-05'),

  (6, 'Ishaan', 'Patel', 6, '1989-10-12', '2014-06-22'),

  (7, 'Saanvi', 'Verma', 7, '1995-11-15', '2019-07-30'),

  (8, 'Kaira', 'Bhatia', 8, '1991-12-18', '2013-08-12'),

  (9, 'Riya', 'Chopra', 9, '1984-01-20', '2009-09-01'),

  (10, 'Mira', 'Joshi', 10, '1993-02-25', '2017-10-14'),

  (11, 'Nikhil', 'Singh', 1, '1986-03-30', '2011-11-11'),

  (12, 'Pooja', 'Ahuja', 2, '1982-04-05', '2010-12-25'),

(13, 'Raj', 'Malhotra', 3, '1994-05-10', '2015-01-20'),

(14, 'Tanya', 'Kapoor', 4, '1990-06-15', '2016-02-05'),

(15, 'Sam', 'Chopra', 5, '1988-07-20', '2014-03-30'),

(16, 'Karan', 'Dutta', 6, '1995-08-25', '2018-04-15'),

(17, 'Neha', 'Sethi', 7, '1992-09-30', '2019-05-10'),

(18, 'Rohit', 'Ghosh', 8, '1987-10-05', '2013-06-20'),

(19, 'Diya', 'Rao', 9, '1994-11-10', '2016-07-25'),

(20, 'Krishna', 'Nair', 10, '1991-12-15', '2017-08-30'),

(21, 'Arjun', 'Nanda', 1, '1983-01-12', '2012-03-05'),

(22, 'Aditi', 'Desai', 2, '1996-02-22', '2017-04-15'),

(23, 'Vikram', 'Chawla', 3, '1984-03-18', '2011-05-25'),

(24, 'Ritika', 'Sharma', 4, '1990-04-30', '2018-06-30'),

(25, 'Kartik', 'Singhal', 5, '1987-05-05', '2013-07-15'),

(26, 'Sneha', 'Tiwari', 6, '1992-06-15', '2019-08-25'),

(27, 'Dev', 'Mishra', 7, '1988-07-20', '2015-09-10'),

(28, 'Mansi', 'Jain', 8, '1991-08-25', '2016-10-20'),

(29, 'Uday', 'Bansal', 9, '1995-09-30', '2020-11-15'),

(30, 'Lavanya', 'Agarwal', 10, '1983-10-05', '2012-12-25'),

(31, 'Amol', 'Pawar', 1, '1989-11-12', '2017-01-30'),

(32, 'Priya', 'Reddy', 2, '1993-12-18', '2014-02-28'),

(33, 'Siddharth', 'Kohli', 3, '1985-01-25', '2016-03-10'),

(34, 'Tisha', 'Mohan', 4, '1990-02-20', '2018-04-15'),

(35, 'Gaurav', 'Shukla', 5, '1986-03-15', '2019-05-20'),

(36, 'Ayesha', 'Bhatia', 6, '1991-04-10', '2015-06-25'),

(37, 'Vishal', 'Thakur', 7, '1995-05-05', '2020-07-30'),

(38, 'Chaitanya', 'Yadav', 8, '1992-06-01', '2013-08-15'),

(39, 'Neeraj', 'Gautam', 9, '1994-07-08', '2016-09-20'),

(40, 'Tanvi', 'Malik', 10, '1988-08-15', '2011-10-25'),

(41, 'Rahul', 'Kumar', 1, '1991-09-20', '2015-11-30'),

(42, 'Meera', 'Sen', 2, '1993-10-25', '2017-12-05'),

(43, 'Shivam', 'Sood', 3, '1986-11-30', '2019-01-10'),

(44, 'Kavya', 'Sharma', 4, '1990-12-15', '2016-02-18'),

(45, 'Advik', 'Rao', 5, '1985-01-25', '2014-03-25'),

(46, 'Rohan', 'Malhotra', 6, '1992-02-15', '2020-04-30'),

(47, 'Aanya', 'Kapoor', 7, '1987-03-20', '2015-05-10'),

(48, 'Kunal', 'Singh', 8, '1995-04-05', '2019-06-15'),

(49, 'Simran', 'Verma', 9, '1988-05-10', '2012-07-20'),

(50, 'Devansh', 'Gupta', 10, '1994-06-15', '2016-08-25');

- **Query to insert records into Salaries table**

  INSERT INTO Salaries (EmployeeID, SalaryAmount, EffectiveDate) VALUES

  (1, 60000.00, '2010-01-10'),

  (2, 70000.00, '2015-02-15'),

  (3, 80000.00, '2012-03-20'),

  (4, 75000.00, '2016-04-18'),

  (5, 62000.00, '2011-05-05'),

  (6, 65000.00, '2014-06-22'),

  (7, 55000.00, '2019-07-30'),

  (8, 58000.00, '2013-08-12'),

  (9, 90000.00, '2009-09-01'),

  (10, 72000.00, '2017-10-14'),

  (11, 61000.00, '2011-11-11'),

  (12, 69000.00, '2010-12-25'),

  (13, 80000.00, '2015-01-20'),

  (14, 75000.00, '2016-02-05'),

  (15, 68000.00, '2014-03-30'),

  (16, 56000.00, '2018-04-15'),

  (17, 60000.00, '2019-05-10'),

  (18, 59000.00, '2013-06-20'),

  (19, 72000.00, '2016-07-25'),

  (20, 75000.00, '2017-08-30'),

  (21, 64000.00, '2012-03-05'),

(22, 70000.00, '2017-04-15'),

(23, 78000.00, '2011-05-25'),

(24, 85000.00, '2018-06-30'),

(25, 61000.00, '2013-07-15'),

(26, 67000.00, '2019-08-25'),

(27, 59000.00, '2015-09-10'),

(28, 63000.00, '2016-10-20'),

(29, 55000.00, '2020-11-15'),

(30, 62000.00, '2012-12-25'),

(31, 68000.00, '2017-01-30'),

(32, 74000.00, '2014-02-28'),

(33, 70000.00, '2016-03-10'),

(34, 78000.00, '2018-04-15'),

(35, 65000.00, '2019-05-20'),

(36, 62000.00, '2015-06-25'),

(37, 58000.00, '2020-07-30'),

(38, 62000.00, '2013-08-15'),

(39, 70000.00, '2016-09-20'),

(40, 72000.00, '2011-10-25'),

(41, 75000.00, '2015-11-30'),

(42, 78000.00, '2017-12-05'),

(43, 80000.00, '2019-01-10'),

(44, 74000.00, '2016-02-18'),

(45, 68000.00, '2014-03-25'),

(46, 65000.00, '2020-04-30'),

(47, 60000.00, '2015-05-10'),

(48, 59000.00, '2019-06-15'),

(49, 70000.00, '2012-07-20'),

(50, 72000.00, '2016-08-25');

1) Retrieve the first and last names of all employees born between 1980 and 1990.

```
select firstName,lastName
from employees
WHERE DateOfBirth > '1980-01-01' AND '1990-12-31';
```

Output:

firstName	lastName
Aarav	Sharma
Vivaan	Gupta
Aditya	Reddy
Ananya	Kumar
Kabir	Mehta
Ishaan	Patel
Saanvi	Verma
Kaira	Bhatia
Riya	Chopra
Mira	Joshi
Nikhil	Singh
Pooja	Ahuja
Raj	Malhotra

2) Find the first and last names of employees who share the same last name as at least one other employee.

```
SELECT FirstName, LastName
FROM Employees
WHERE LastName IN (
    SELECT LastName
    FROM Employees
    GROUP BY LastName
    HAVING COUNT(*) > 1);
```

Output:

FirstName	LastName
Aarav	Sharma
Vivaan	Gupta
Aditya	Reddy
Ananya	Kumar
Saanvi	Verma
Kaira	Bhatia
Riya	Chopra
Nikhil	Singh
Raj	Malhotra
Tanya	Kapoor
Sam	Chopra

Diya	Rao
Ritika	Sharma
Priya	Reddy
Ayesha	Bhatia
Rahul	Kumar
Kavya	Sharma
Advik	Rao
Rohan	Malhotra
Aanya	Kapoor
Kunal	Singh
Simran	Verma
Devansh	Gupta

3) Retrieve the first names of employees who were hired after the youngest employee was born

```
SELECT FirstName

FROM Employees

WHERE HireDate > (SELECT MIN(DateOfBirth) FROM Employees);
```

Output:

FirstName
Aarav
Vivaan
Aditya
Ananya
Kabir
Ishaan
Saanvi
Kaira
Riya

4) Find the employee with the maximum age in the company.

```
SELECT FirstName, LastName,DateOfBirth

FROM Employees

WHERE DateOfBirth = (SELECT MIN(DateOfBirth) FROM Employees);
```

Output:

FirstName	LastName	DateOfBirth
Pooja	Ahuja	05-04-1982

5) Find the employee who have been with the company the longest.

```
SELECT FirstName, LastName, HireDate

FROM Employees

ORDER BY HireDate ASC

LIMIT 1;
```

Output:

FirstName	LastName	HireDate
Riya	Chopra	01-09-2009

6) List employees in descending order by their age, showing only the top 10 oldest.

```
SELECT FirstName, LastName, DateOfBirth

FROM Employees

ORDER BY DateOfBirth ASC

LIMIT 10;
```

Output:

FirstName	LastName	DateOfBirth
Pooja	Ahuja	05-04-1982
Arjun	Nanda	12-01-1983
Lavanya	Agarwal	05-10-1983
Riya	Chopra	20-01-1984
Vikram	Chawla	18-03-1984
Siddharth	Kohli	25-01-1985
Advik	Rao	25-01-1985
Aarav	Sharma	15-05-1985
Gaurav	Shukla	15-03-1986
Nikhil	Singh	30-03-1986

7) Find the employee with the highest salary.

```
SELECT FirstName, LastName, SalaryAmount

FROM Employees

JOIN Salaries ON Employees.EmployeeID = Salaries.EmployeeID

ORDER BY SalaryAmount DESC

LIMIT 1;
```

Output:

FirstName	LastName	SalaryAmount
Riya	Chopra	90000

8) Count the number of employees in each department.

```
SELECT d.DepartmentName, COUNT(e.EmployeeID) AS EmployeeCount

FROM Departments as d

LEFT JOIN Employees as e ON d.DepartmentID = e.DepartmentID

GROUP BY d.DepartmentID, d.DepartmentName;
```

Output:

DepartmentName	EmployeeCount
Human Resources	5
Finance	5
Engineering	5
Marketing	5
Sales	5
IT	5
Customer Service	5
Research and Development	5
Legal	5
Administration	5

9) List all employees, showing their ages calculated from their DateOfBirth

```
SELECT FirstName, LastName, YEAR(CURRENT_DATE) - YEAR(DateOfBirth) AS Age

FROM Employees;
```

Output:

FirstName	LastName	Age
Aarav	Sharma	39
Vivaan	Gupta	34
Aditya	Reddy	36
Ananya	Kumar	32
Kabir	Mehta	37
Ishaan	Patel	35
Saanvi	Verma	29
Kaira	Bhatia	33
Riya	Chopra	40

10) Find the employee who were born on the same date

```
SELECT FirstName, LastName, DateOfBirth

FROM Employees
```

```
WHERE DateOfBirth IN (
  SELECT DateOfBirth
  FROM Employees
  GROUP BY DateOfBirth
  HAVING COUNT(*) > 1
);
```

Output:

FirstName	LastName	DateOfBirth
Sam	Chopra	20-07-1988
Dev	Mishra	20-07-1988
Siddharth	Kohli	25-01-1985
Advik	Rao	25-01-1985

11) Retrieve the employee with the most recent hire date.

```
SELECT FirstName, LastName, HireDate
FROM Employees
ORDER BY HireDate DESC
LIMIT 1;
```

Output:

FirstName	LastName	HireDate
Uday	Bansal	15-11-2020

12) Retrieve the last name of employees, but with the first character lower

```
SELECT FirstName, CONCAT(lower(LEFT(LastName, 1)), LOWER(SUBSTRING(LastName, 2)))
AS loweredLastName
FROM Employees;
```

Output:

FirstName	loweredLastName
Aarav	sharma
Vivaan	gupta
Aditya	reddy
Ananya	kumar
Kabir	mehta
Ishaan	patel
Saanvi	verma
Kaira	bhatia
Riya	chopra

13) Find employees with first names longer than their last names.

```
SELECT FirstName, LastName

FROM Employees

WHERE LENGTH(FirstName) > LENGTH(LastName);
```

Output:

FirstName	LastName
Vivaan	Gupta
Aditya	Reddy
Ananya	Kumar
Ishaan	Patel
Saanvi	Verma
Nikhil	Singh
Diya	Rao
Krishna	Nair
Mansi	Jain

14) Find the average age of employees.

```
SELECT round(AVG(YEAR(CURRENT_DATE) - YEAR(DateOfBirth))) AS AverageAge

FROM Employees;
```

Output:

AverageAge
34

15) Count the number of employees hired each year.

```
SELECT YEAR(HireDate) AS HireYear, COUNT(*) AS NumberOfHires

FROM Employees

GROUP BY YEAR(HireDate);
```

Output:

HireYear	NumberOfHires
2010	2
2015	6
2012	4
2016	8
2011	4
2014	4
2019	6
2013	4
2009	1
2017	5
2018	3
2020	3

16)Find the number of employees with the same last name.

```
SELECT lastName, COUNT(*) AS Count
FROM Employees
GROUP BY lastName
HAVING COUNT(*) > 1;
```

Output:

lastName	Count
Sharma	3
Gupta	2
Reddy	2
Kumar	2
Verma	2
Bhatia	2
Chopra	2
Singh	2
Malhotra	2
Kapoor	2
Rao	2

17) Find employees with salaries that are less than the average salary(68120)

```
SELECT FirstName, LastName, SalaryAmount
FROM Employees e
JOIN Salaries s ON e.EmployeeID = s.EmployeeID
WHERE s.SalaryAmount < (SELECT AVG(SalaryAmount) FROM Salaries);
```

Output:

FirstName	LastName	SalaryAmount
Aarav	Sharma	60000
Kabir	Mehta	62000
Ishaan	Patel	65000
Saanvi	Verma	55000
Kaira	Bhatia	58000
Nikhil	Singh	61000
Sam	Chopra	68000
Karan	Dutta	56000
Neha	Sethi	60000

18) Get employees who do not have a salary record.

```
SELECT FirstName, LastName

FROM Employees e

LEFT JOIN Salaries s ON e.EmployeeID = s.EmployeeID

WHERE s.SalaryAmount IS NULL;
```

Output:

19)Calculate age and salary of employees older than 30. Query must return First name last name age and salary

```
SELECT firstname,lastname,(YEAR(CURRENT_DATE) - YEAR(e.DateOfBirth)) as
AGE,s.SalaryAmount AS salary

FROM Employees e

JOIN Salaries s ON e.EmployeeID = s.EmployeeID

WHERE (YEAR(CURRENT_DATE) - YEAR(e.DateOfBirth)) > 30;
```

Output:

firstname	lastname	AGE	salary
Aarav	Sharma	39	60000
Vivaan	Gupta	34	70000
Aditya	Reddy	36	80000
Ananya	Kumar	32	75000
Kabir	Mehta	37	62000
Ishaan	Patel	35	65000
Kaira	Bhatia	33	58000
Riya	Chopra	40	90000
Mira	Joshi	31	72000

20)Find employees who were born in the 1990s.

```
SELECT FirstName, LastName, DateOfBirth
FROM Employees
WHERE YEAR(DateOfBirth) BETWEEN 1990 AND 1999;
```

Output:

FirstName	LastName	DateOfBirth
Vivaan	Gupta	20-06-1990
Ananya	Kumar	30-08-1992
Saanvi	Verma	15-11-1995
Kaira	Bhatia	18-12-1991
Mira	Joshi	25-02-1993
Raj	Malhotra	10-05-1994
Tanya	Kapoor	15-06-1990
Karan	Dutta	25-08-1995
Neha	Sethi	30-09-1992

-- Joins

-- Inner Join

21) Get a list of employees along with their hire dates, but only those hired in 2015

```
SELECT e.FirstName, e.LastName, e.HireDate
FROM Employees e
INNER JOIN Salaries s ON e.EmployeeID = s.EmployeeID
WHERE YEAR(e.HireDate) = 2015;
```

Output:

FirstName	LastName	HireDate
Vivaan	Gupta	15-02-2015
Raj	Malhotra	20-01-2015
Dev	Mishra	10-09-2015
Ayesha	Bhatia	25-06-2015
Rahul	Kumar	30-11-2015
Aanya	Kapoor	10-05-2015

22) Retrieve employee names and their respective salaries, filtering out those with salaries below 30,000.

```
SELECT e.FirstName, e.LastName, s.SalaryAmount

FROM Employees e

INNER JOIN Salaries s ON e.EmployeeID = s.EmployeeID

WHERE s.SalaryAmount >= 30000;
```

Output:

FirstName	LastName	SalaryAmount
Aarav	Sharma	60000
Vivaan	Gupta	70000
Aditya	Reddy	80000
Ananya	Kumar	75000
Kabir	Mehta	62000
Ishaan	Patel	65000
Saanvi	Verma	55000
Kaira	Bhatia	58000
Riya	Chopra	90000

-- left Join

23) Retrieve all employees and their salaries,show the difference between the highest salary in the department and each employee's salary.

```
SELECT e.FirstName, e.LastName, s.SalaryAmount,

    (SELECT MAX(s2.SalaryAmount)

     FROM Salaries s2

     JOIN Employees e2 ON s2.EmployeeID = e2.EmployeeID

     WHERE e2.DepartmentID = e.DepartmentID) - s.SalaryAmount AS SalaryDifference

FROM Employees e

LEFT JOIN Salaries s ON e.EmployeeID = s.EmployeeID;
```

Output:

FirstName	LastName	SalaryAmount	SalaryDifference
Aarav	Sharma	60000	15000
Vivaan	Gupta	70000	8000
Aditya	Reddy	80000	0
Ananya	Kumar	75000	10000
Kabir	Mehta	62000	6000
Ishaan	Patel	65000	2000
Saanvi	Verma	55000	5000
Kaira	Bhatia	58000	5000
Riya	Chopra	90000	0

24) Get a list of all employees and their salaries and identify the maximum salary from their respective departments.

```
SELECT e.FirstName, e.LastName, s.SalaryAmount,
    (SELECT MAX(s2.SalaryAmount)
     FROM Salaries s2
     JOIN Employees e2 ON s2.EmployeeID = e2.EmployeeID
     WHERE e2.DepartmentID = e.DepartmentID) AS MaxDepartmentSalary
FROM Employees e
LEFT JOIN Salaries s ON e.EmployeeID = s.EmployeeID;
```

Output:

FirstName	LastName	SalaryAmount	MaxDepartmentSalary
Aarav	Sharma	60000	75000
Vivaan	Gupta	70000	78000
Aditya	Reddy	80000	80000
Ananya	Kumar	75000	85000
Kabir	Mehta	62000	68000
Ishaan	Patel	65000	67000
Saanvi	Verma	55000	60000
Kaira	Bhatia	58000	63000
Riya	Chopra	90000	90000

-- Right Join

-- 25) Find the number of employees in each department,including department name

```
SELECT d.DepartmentName, COUNT(e.EmployeeID) AS NumberOfEmployees
FROM Departments d
RIGHT JOIN Employees e ON d.DepartmentID = e.DepartmentID
GROUP BY d.DepartmentName;
```

Output:

DepartmentName	NumberOfEmployees
Human Resources	5
Finance	5
Engineering	5
Marketing	5
Sales	5
IT	5
Customer Service	5
Research and Development	5
Legal	5
Administration	5

-- Self Join

-- 26)Retrieve employees who have the same hire date as any other employee.

```
SELECT e1.FirstName, e1.LastName, e1.HireDate
FROM Employees e1
JOIN Employees e2 ON e1.HireDate = e2.HireDate
WHERE e1.EmployeeID <> e2.EmployeeID;
```

Output:

FirstName	LastName	HireDate
Tisha	Mohan	15-04-2018
Karan	Dutta	15-04-2018

-- union

27) List all employees and their departments

```
SELECT
   e.EmployeeID,
   e.FirstName,
   e.LastName,
   d.DepartmentName
FROM Employees e
LEFT JOIN Departments d ON e.DepartmentID = d.DepartmentID
UNION
SELECT
   NULL AS EmployeeID,
   NULL AS FirstName,
   NULL AS LastName,
   d.DepartmentName
FROM
   Departments d
WHERE
   d.DepartmentID NOT IN (SELECT DepartmentID FROM Employees);
```

Output:

EmployeeID	FirstName	LastName	DepartmentName
1	Aarav	Sharma	Human Resources
2	Vivaan	Gupta	Finance
3	Aditya	Reddy	Engineering
4	Ananya	Kumar	Marketing
5	Kabir	Mehta	Sales
6	Ishaan	Patel	IT
7	Saanvi	Verma	Customer Service
8	Kaira	Bhatia	Research and Development
9	Riya	Chopra	Legal
10	Mira	Joshi	Administration