**HR Management System.**

1. First create departments table

CREATE TABLE Departments (

DepartmentID INT PRIMARY KEY AUTO\_INCREMENT,

DepartmentName VARCHAR(100) not null,

ManagerID INT

);

1. Create employee table

CREATE TABLE Employees (

EmployeeID INT PRIMARY KEY AUTO\_INCREMENT,

FirstName VARCHAR(50) NOT null,

LastName VARCHAR(50) NOT null,

Email VARCHAR(100)UNIQUE,

Phone VARCHAR(15),

HireDate DATE,

DepartmentID INT,

ManagerID INT,

Salary DECIMAL(10, 2),

FOREIGN KEY (DepartmentID) REFERENCES Departments(DepartmentID),

FOREIGN KEY (ManagerID) REFERENCES Employees(EmployeeID)

);

1. Create performanceRiviews table

CREATE TABLE PerformanceReviews (

ReviewID INT PRIMARY KEY AUTO\_INCREMENT,

EmployeeID INT not null,

ReviewDate DATE not null,

PerformanceScore VARCHAR(20),

Comments ,

FOREIGN KEY (EmployeeID) REFERENCES Employees(EmployeeID)

);

1. Create payroll table

CREATE TABLE Payroll (

PayrollID INT PRIMARY KEY AUTO\_INCREMENT,

EmployeeID INT not null,

PaymentDate DATE ,

Amount DECIMAL(10, 2),

PaymentMethod VARCHAR(50),

FOREIGN KEY (EmployeeID) REFERENCES Employees(EmployeeID)

);

1. Insert data from department table

INSERT INTO Departments (DepartmentID, DepartmentName, ManagerID)

VALUES

(1, 'Human Resources', NULL),

(2, 'Finance', NULL),

(3, 'IT', NULL);

1. Insert data from employee table

INSERT INTO Employees (EmployeeID, FirstName, LastName, Email, Phone, HireDate, DepartmentID, ManagerID, Salary)

VALUES

(101, 'Amit', 'Sharma', 'amit.sharma@example.com', '9876543210', '2022-06-15', 1, NULL, 60000),

(102, 'Priya', 'Mehra', 'priya.mehra@example.com', '9876543211', '2023-02-20', 2, 101, 50000),

(103, 'Ravi', 'Kumar', 'ravi.kumar@example.com', '9876543212', '2021-08-01', 3, 101, 80000),

(104, 'Neha', 'Singh', 'neha.singh@example.com', '9876543213', '2023-07-10', 1, 101, 45000);

1. Insert data from performanceReviews table

INSERT INTO PerformanceReviews (ReviewID, EmployeeID, ReviewDate, PerformanceScore, Comments)

VALUES

(1, 101, '2023-12-01', 'Excellent', 'Outstanding performance'),

(2, 102, '2023-11-15', 'Good', 'Meeting expectations'),

(3, 103, '2023-10-20', 'Excellent', 'Consistently exceeding goals'),

(4, 104, '2023-09-30', 'Average', 'Needs improvement in some areas');

1. Insert data from payroll table

INSERT INTO Payroll (PayrollID, EmployeeID, PaymentDate, Amount, PaymentMethod) VALUES (1, 101, '2023-12-15', 60000, 'Bank Transfer'), (2, 102, '2023-12-15', 50000, 'Check'), (3, 103, '2023-12-15', 80000, 'Bank Transfer'), (4, 104, '2023-12-15', 45000, 'Check');

**Queires**

1. **Retrieve the names and contact details of employees hired after January 1, 2023.**

**🡺** SELECT FirstName,LastName,HireDate from employees WHERE HireDate > 2022-01-01

1. **Find the total payroll amount paid to each department.**

🡺 SELECT departments.DepartmentName,sum(payroll.Amount)as total\_payroll from payroll

JOIN employees on payroll.EmployeeID = employees.EmployeeID

JOIN departments on departments.ManagerID = employees.ManagerID

GROUP by departments.DepartmentName

1. **List all employees who have not been assigned a manager.**

🡺 SELECT employees.FirstName, employees.LastName,employees.ManagerID FROM employees

WHERE employees.ManagerID is null

1. **Retrieve the highest salary in each department along with the employee’s name.**

🡺 SELECT departments.DepartmentName ,MAX(employees.Salary)as Highest\_Salary FROM employees

JOIN departments on departments.DepartmentID = employees.DepartmentID

GROUP by departments.DepartmentName ORDER by Highest\_salary desc

1. **Find the most recent performance review for each employee.**

🡺 SELECT employees.FirstName,employees.LastName ,max(performancereviews.ReviewDate) as Recent\_Review FROM employees

JOIN performancereviews on performancereviews.EmployeeID = employees.EmployeeID

GROUP by performancereviews.EmployeeID

1. **Count the number of employees in each department.**

🡺 SELECT departments.DepartmentName,COUNT(employees.EmployeeID)as Employees\_In\_Department

FROM departments LEFT JOIN employees on departments.DepartmentID = employees.DepartmentID

GROUP by departments.DepartmentName

1. **List all employees who have received a performance score of "Excellent."**

🡺 SELECT employees.FirstName,employees.LastName,employees.Email ,performancereviews.PerformanceScore as performance FROM employees

JOIN performancereviews on performancereviews.EmployeeID = employees.EmployeeID

WHERE performancereviews.PerformanceScore = 'Excellent'

1. **Identify the most frequently used payment method in payroll.**

**🡺**

1. **Retrieve the top 5 highest-paid employees along with their departments.**

🡺 SELECT employees.FirstName,employees.LastName,departments.DepartmentName , employees.Salary FROM employees

JOIN departments on employees.DepartmentID =departments.DepartmentID

ORDER By employees.Salary DESC LIMIT 5

1. **Showdetails of all employees who report directly to a specific manager (e.g., ManagerID = 101).**

🡺 SELECT \* from employees WHERE employees.ManagerID = 101