

3. HR MANAGEMENT APPLICATION

```
CREATE DATABASE HR_MANAGEMENT;
```

```
USE HR_MANAGEMENT;
```

DEPARTMENTS tables

```
CREATE TABLE DEPARTMENTS (  
    DEPARTMENT_ID INT PRIMARY KEY,  
    DEPARTMENT_NAME VARCHAR(100),  
    MANAGER_ID INT  
);
```

```
INSERT INTO DEPARTMENTS (DEPARTMENT_ID, DEPARTMENT_NAME,  
MANAGER_ID) VALUES  
(1, 'Human Resources', 1001),  
(2, 'Finance', 1002),  
(3, 'Engineering', 1003),  
(4, 'Marketing', 1004),  
(5, 'IT Support', 1005);
```

Employees Table

```
CREATE TABLE EMPLOYEES (  
    EMPLOYEE_ID INT PRIMARY KEY,  
    FIRST_NAME VARCHAR(100),  
    LAST_NAME VARCHAR(100),  
    EMAIL VARCHAR(100),  
    PHONE VARCHAR(20),  
    HIRE_DATE DATE,  
    DEPARTMENT_ID INT,  
    MANAGER_ID INT,  
    SALARY INT,  
    FOREIGN KEY (DEPARTMENT_ID) REFERENCES  
DEPARTMENTS(DEPARTMENT_ID),  
    FOREIGN KEY (MANAGER_ID) REFERENCES EMPLOYEES(EMPLOYEE_ID)  
) ENGINE=InnoDB;
```

```
INSERT INTO EMPLOYEES (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, EMAIL,  
PHONE, HIRE_DATE, DEPARTMENT_ID, MANAGER_ID, SALARY) VALUES  
(101, 'Amit', 'Sharma', 'amit.sharma@email.com', '9876543210', '2018-06-15', 1, NULL,  
75000), -- top-level manager  
(102, 'Neha', 'Verma', 'neha.verma@email.com', '9876543211', '2019-01-10', 1, 101, 65000),  
(103, 'Ravi', 'Kumar', 'ravi.kumar@email.com', '9876543212', '2020-03-20', 2, 101, 60000),  
(104, 'Sneha', 'Mehta', 'sneha.mehta@email.com', '9876543213', '2017-11-05', 2, 103,  
58000),  
(105, 'Vikas', 'Singh', 'vikas.singh@email.com', '9876543214', '2016-09-15', 3, 101, 62000),  
(106, 'Anjali', 'Joshi', 'anjali.joshi@email.com', '9876543215', '2021-04-25', 3, 105, 55000),
```

(107, 'Suresh', 'Yadav', 'suresh.yadav@email.com', '9876543216', '2019-07-30', 4, 101, 61000),
 (108, 'Kiran', 'Kapoor', 'kiran.kapoor@email.com', '9876543217', '2020-12-12', 4, 107, 53000),
 (109, 'Alok', 'Bhatia', 'alok.bhatia@email.com', '9876543218', '2018-05-18', 5, 101, 59000),
 (110, 'Divya', 'Malhotra', 'divya.malhotra@email.com', '9876543219', '2022-02-14', 5, 109, 52000),
 (111, 'Nitin', 'Rao', 'nitin.rao@email.com', '9876543220', '2021-09-01', 1, 102, 50000),
 (112, 'Ruchi', 'Desai', 'ruchi.desai@email.com', '9876543221', '2020-10-10', 2, 104, 51000),
 (113, 'Arjun', 'Gill', 'arjun.gill@email.com', '9876543222', '2019-08-08', 3, 106, 54000),
 (114, 'Meera', 'Seth', 'meera.seth@email.com', '9876543223', '2017-02-19', 4, 108, 56000),
 (115, 'Tarun', 'Chopra', 'tarun.chopra@email.com', '9876543224', '2023-01-05', 5, 110, 48000);

PERFORMANCE_REVIEWS TABLE

```
CREATE TABLE PERFORMANCE_REVIEWS (
  REVIEW_ID INT PRIMARY KEY,
  EMPLOYEE_ID INT,
  REVIEW_DATE DATE,
  PERFORMANCE_SCORE
ENUM('EXCELLENT','GOOD','SATISFACTORY','AVERAGE','POOR'),
  COMMENT TEXT,
  FOREIGN KEY (EMPLOYEE_ID) REFERENCES EMPLOYEES(EMPLOYEE_ID)
);
```

```
INSERT INTO PERFORMANCE_REVIEWS (REVIEW_ID, EMPLOYEE_ID,
REVIEW_DATE, PERFORMANCE_SCORE, COMMENT) VALUES
(1, 101, '2023-12-15', 'EXCELLENT', 'Consistently exceeds expectations and leads team effectively.'),
(2, 102, '2023-12-10', 'GOOD', 'Strong performer with occasional need for guidance.'),
(3, 103, '2023-12-12', 'SATISFACTORY', 'Meets expectations; room for improvement in initiative.'),
(4, 104, '2023-12-08', 'AVERAGE', 'Performance is average; needs regular supervision.'),
(5, 105, '2023-12-11', 'GOOD', 'Technical skills are strong, should work on communication.'),
(6, 106, '2023-12-13', 'EXCELLENT', 'Very dependable and a strong team player.'),
(7, 107, '2023-12-14', 'POOR', 'Frequent delays in deliverables; needs major improvement.'),
(8, 108, '2023-12-09', 'GOOD', 'Handles tasks well under pressure.'),
(9, 109, '2023-12-07', 'SATISFACTORY', 'Maintains steady performance.'),
(10, 110, '2023-12-06', 'EXCELLENT', 'Takes ownership and drives projects forward.');
```

PAYROLL TABLE

```
CREATE TABLE PAYROLL (
  PAYROLL_ID INT PRIMARY KEY,
  EMPLOYEE_ID INT,
  PAYMENT_DATE DATE,
```

```
PAYMENT_METHOD ENUM('CREDIT_CARD', 'PAYPAL', 'DEBIT_CARD', 'CASH',  
'CHEQUE', 'UPI'),  
FOREIGN KEY (EMPLOYEE_ID) REFERENCES EMPLOYEES(EMPLOYEE_ID)  
);
```

```
INSERT INTO PAYROLL (PAYROLL_ID, EMPLOYEE_ID, PAYMENT_DATE,  
PAYMENT_METHOD) VALUES  
(1, 101, '2024-04-30', 'CREDIT_CARD'),  
(2, 102, '2024-04-30', 'UPI'),  
(3, 103, '2024-04-30', 'DEBIT_CARD'),  
(4, 104, '2024-04-30', 'CASH'),  
(5, 105, '2024-04-30', 'CHEQUE'),  
(6, 106, '2024-04-30', 'PAYPAL'),  
(7, 107, '2024-04-30', 'UPI'),  
(8, 108, '2024-04-30', 'CREDIT_CARD'),  
(9, 109, '2024-04-30', 'DEBIT_CARD'),  
(10, 110, '2024-04-30', 'CASH');
```

Assignment Queries

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

```
SELECT  
    FIRST_NAME,  
    LAST_NAME,  
    EMAIL,  
    PHONE,  
    HIRE_DATE  
FROM  
    EMPLOYEES  
WHERE  
    HIRE_DATE > '2023-01-01';
```

2.Find the total payroll amount paid to each department.

```

SELECT
    D.DEPARTMENT_NAME,
    SUM(E.SALARY) AS TOTAL_PAYROLL
FROM
    PAYROLL P
JOIN
    EMPLOYEES E ON P.EMPLOYEE_ID = E.EMPLOYEE_ID
JOIN
    DEPARTMENTS D ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
GROUP BY
    D.DEPARTMENT_NAME;

```

3.all list employees who have not been assigned a manager.

```

SELECT
    EMPLOYEE_ID,
    FIRST_NAME,
    LAST_NAME,
    EMAIL,
    PHONE,
    DEPARTMENT_ID,
    SALARY
FROM
    EMPLOYEES
WHERE
    MANAGER_ID IS NULL;

```

4.Retrieve the highest salary in each department along with the employee's name.

```

SELECT
    D.DEPARTMENT_NAME,
    E.FIRST_NAME,
    E.LAST_NAME,
    E.SALARY
FROM
    EMPLOYEES E
JOIN

```

```

DEPARTMENTS D ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
WHERE
  E.SALARY = (
    SELECT MAX(SALARY)
    FROM EMPLOYEES
    WHERE DEPARTMENT_ID = E.DEPARTMENT_ID
  );

```

5.Find the most recent performance review for each employee.

```

SELECT
  PR.EMPLOYEE_ID,
  E.FIRST_NAME,
  E.LAST_NAME,
  PR.REVIEW_DATE,
  PR.PERFORMANCE_SCORE,
  PR.COMMENT
FROM
  PERFORMANCE_REVIEWS PR
JOIN
  EMPLOYEES E ON PR.EMPLOYEE_ID = E.EMPLOYEE_ID
WHERE
  PR.REVIEW_DATE = (
    SELECT MAX(REVIEW_DATE)
    FROM PERFORMANCE_REVIEWS
    WHERE EMPLOYEE_ID = PR.EMPLOYEE_ID
  );

```

6.Count the number of employees in each department.

```

SELECT
  D.DEPARTMENT_NAME,
  COUNT(E.EMPLOYEE_ID) AS EMPLOYEE_COUNT
FROM
  DEPARTMENTS D
LEFT JOIN
  EMPLOYEES E ON D.DEPARTMENT_ID = E.DEPARTMENT_ID
GROUP BY
  D.DEPARTMENT_NAME;

```

7. list all employees who have received a performance score of "Excellent."Identify the most frequently used payment method in payroll

```
SELECT
    E.EMPLOYEE_ID,
    E.FIRST_NAME,
    E.LAST_NAME,
    PR.REVIEW_DATE,
    PR.COMMENT
FROM
    EMPLOYEES E
JOIN
    PERFORMANCE_REVIEWS PR ON E.EMPLOYEE_ID = PR.EMPLOYEE_ID
WHERE
    PR.PERFORMANCE_SCORE = 'EXCELLENT';
```

8.Retrieve the top 5 highest paid employees along with their departments.

```
SELECT
    E.EMPLOYEE_ID,
    E.FIRST_NAME,
    E.LAST_NAME,
    D.DEPARTMENT_NAME,
    E.SALARY
FROM
    EMPLOYEES E
JOIN
    DEPARTMENTS D ON E.DEPARTMENT_ID = D.DEPARTMENT_ID
ORDER BY
    E.SALARY DESC
LIMIT 5;
```

9.Show details of all employees who report directly to a specific manager (e.g., ManagerID = 101).

```
SELECT
    EMPLOYEE_ID,
```

```
FIRST_NAME,  
LAST_NAME,  
EMAIL,  
PHONE,  
DEPARTMENT_ID,  
SALARY  
FROM  
  EMPLOYEES  
WHERE  
  MANAGER_ID = 101;
```