Hope AI
MySQL Assignment

Thanigaivel G

Table Creation

Creating Employees Table

```
CREATE TABLE employees(
    employee_id INT PRIMARY KEY,
    name VARCHAR(50),
    age INT,
    salary INT,
    department_id INT
);
```

Inserting values into the Employees Table

```
INSERT INTO employees (employee_id, name, age, salary, department_id) VALUES (1, 'John', 30, 60000, 101);

INSERT INTO employees (employee_id, name, age, salary, department_id) VALUES (2, 'Emily', 25, 48000, 102);

INSERT INTO employees (employee_id, name, age, salary, department_id) VALUES (3, 'Michael', 40, 75000, 103);

INSERT INTO employees (employee_id, name, age, salary, department_id) VALUES (4, 'Sara', 35, 56000, 101);

INSERT INTO employees (employee_id, name, age, salary, department_id) VALUES (5, 'David', 28, 49000, 102);

INSERT INTO employees (employee_id, name, age, salary, department_id) VALUES (6, 'Robert', 45, 90000, 103);

INSERT INTO employees (employee_id, name, age, salary, department_id) VALUES (7, 'Sophia', 29, 51000, 102);
```

Create Departments Table

```
CREATE TABLE departments (
    department_id INT PRIMARY KEY,
    department_name VARCHAR(50)
);
```

Inserting values into the Departments Table

```
INSERT INTO departments (department_id, department_name) VALUES (101, 'HR');
INSERT INTO departments (department_id, department_name) VALUES (102, 'Finance');
INSERT INTO departments (department_id, department_name) VALUES (103, 'IT');
```

Create Sales Table

```
CREATE TABLE sales (
    sale_id INT PRIMARY KEY,
    customer_id INT,
    amount DECIMAL(10,2),
    sale_date DATE
);
```

Inserting Values into Sales Table

```
INSERT INTO sales (sale_id, customer_id, amount, sale_date) VALUES (1, 101, 4500.00,
to_date('2023-03-15','YYYY-MM-DD'));

INSERT INTO sales (sale_id, customer_id, amount, sale_date) VALUES (2, 102, 5500.00,
to_date('2023-03-16','YYYY-MM-DD'));

INSERT INTO sales (sale_id, customer_id, amount, sale_date) VALUES (3, 103, 7000.00,
to_date('2023-03-17','YYYY-MM-DD'));
```

```
INSERT INTO sales (sale_id, customer_id, amount, sale_date) VALUES (4, 104, 3000.00, to_date('2023-03-18','YYYY-MM-DD'));

INSERT INTO sales (sale_id, customer_id, amount, sale_date) VALUES (5, 105, 6000.00, to_date('2023-03-19','YYYY-MM-DD'));
```

Create Products Table

```
CREATE TABLE products (
    product_id INT PRIMARY KEY,
    product_name VARCHAR(50),
    price INT
);
```

Inserting values into Products Table

```
INSERT INTO products (product_id, product_name, price) VALUES (1, 'Laptop', 1000);
INSERT INTO products (product_id, product_name, price) VALUES (2, 'Mobile', 500);
INSERT INTO products (product_id, product_name, price) VALUES (3, 'Tablet', 300);
INSERT INTO products (product_id, product_name, price) VALUES (4, 'Headphones', 100);
INSERT INTO products (product_id, product_name, price) VALUES (5, 'Smartwatch', 200);
```

Create Orders Table

```
CREATE TABLE orders (
    order_id INT PRIMARY KEY,
    customer_name VARCHAR(50),
    order_date DATE,
    order_amount INT
);
```

Inserting Values into the Orders Table

INSERT INTO orders (order_id, customer_name, order_date, order_amount) VALUES (1, 'John', to_date('2023-05-01','YYYY-MM-DD'), 500);

INSERT INTO orders (order_id, customer_name, order_date, order_amount) VALUES (2, 'Emily', to_date('2023-05-02','YYYY-MM-DD'), 700);

INSERT INTO orders (order_id, customer_name, order_date, order_amount) VALUES (3, 'Michael', to_date('2023-05-03','YYYY-MM-DD'), 1200);

INSERT INTO orders (order_id, customer_name, order_date, order_amount) VALUES (4, 'Sara', to_date('2023-05-04','YYYY-MM-DD'), 450);

INSERT INTO orders (order_id, customer_name, order_date, order_amount) VALUES (5, 'David', to_date('2023-05-05','YYYY-MM-DD'), 900);

INSERT INTO orders (order_id, customer_name, order_date, order_amount) VALUES (6, 'John', to_date('2023-05-06','YYYY-MM-DD'), 600);

INSERT INTO orders (order_id, customer_name, order_date, order_amount) VALUES (7, 'Emily', to_date('2023-05-07','YYYY-MM-DD'), 750);

Data Querying

-- Question 1: Retrieve Employee Details

Retrieve all employees whose salary is greater than 60000.

```
select * from employees where salary > 60000 order by 1;
```

EMPLOYEE_ID	NAME	AGE	SALARY	DEPARTMENT_ID
3	Michael	40	75000	103
6	Robert	45	90000	103

2 rows returned in 0.01 seconds Download

-- Question 2: Find Total Sales Per Customer

Calculate the total sales amount for each customer from the sales table.

```
select customer_id, sum(amount) total_sales from sales group by customer_id order by 1
```

CUSTOMER_ID	TOTAL_SALES
101	4500
102	5500
103	7000
104	3000
105	6000

5 rows returned in 0.01 seconds Download

-- Question 3: Employee Salary in Finance Department

Retrieve the names and salaries of all employees working in the 'Finance' department.

```
select employee_id,name,salary
  from employees
 where department_id = (select department_id from departments where department_name =
'Finance')
order by 1
```

EMPLOYEE_ID	NAME	SALARY
2	Emily	48000
5	David	49000
7	Sophia	51000

3 rows returned in 0.01 seconds Download

-- Question 4: Total Sales on a Specific Date

Find the total sales amount made on '2023-03-17'.

```
select sale_date, sum(amount) total_sales
   from sales
 where sale_date = to_date('2023-03-17','YYYY-MM-DD')   group by sale_date
```

SALE_DATE	TOTAL_SALES
3/17/2023	7000

1 rows returned in 0.01 seconds Download

-- Question 5: High-Value Orders

Get the names of customers who have placed an order of more than 600.

```
select customer_name, order_amount
   from orders
 where order_amount > 600
```

CUSTOMER_NAME	ORDER_AMOUNT
David	900
Michael	1200
Emily	750
Emily	700

4 rows returned in 0.00 seconds Download

Scenario Based Questions

1) Employee Salary Analysis

Question:

Find the names and salaries of employees who earn more than the average salary in the company.

```
select employee_id,name, salary
  from employees
 where salary > (select round(avg(salary),2) from employees)
 order by 1
```

EMPLOYEE_ID	NAME	SALARY
3	Michael	75000
6	Robert	90000

2 rows returned in 0.01 seconds Download

2) Customer Orders without matching records

Question:

Retrieve a list of customer names who have not placed any orders.

```
select name from employees
    where name not in (select distinct customer_name from orders)
```

NAME
Robert
Sophia

2 rows returned in 0.02 seconds Download

3) Product Sales Summary

Question:

Display the total sales amount for each product.

Unable to write query, since product_id is not linked with the sales / orders table

4) Department Wise Employee Count

Question:

List each department name with the1 number of employees working in it.

```
select b.department_name, count(*)
  from employees a, departments b
 where a.department_id = b.department_id
group by b.department_name
```

DEPARTMENT_NAME	COUNT(*)
HR	2
IT	2
Finance	3

3 rows returned in 0.01 seconds Download

5) Top 3 Highest Sales

Question:

Find the top 3 highest sales transactions.

```
select * from (
  select amount from sales order by 1 desc
)
where rownum <= 3
```

AMOUNT
7000
6000
5500

3 rows returned in 0.00 seconds Download

6) Calculate Employee Salary Rank by Department

Question:

write query to display each employee's name, department name , salary and their salary rank within their respective department.

```
select  b.department_name, a.name, a.salary, rank() over (partition by a.department_id order by a.salary desc) salary_rank

 from employees a, departments b

where a.department_id = b.department_id
```

DEPARTMENT_NAME	NAME	SALARY	SALARY_RANK
HR	John	60000	1
HR	Sara	56000	2
Finance	Sophia	51000	1
Finance	David	49000	2
Finance	Emily	48000	3
IT	Robert	90000	1
IT	Michael	75000	2

7 rows returned in 0.00 seconds Download