**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);

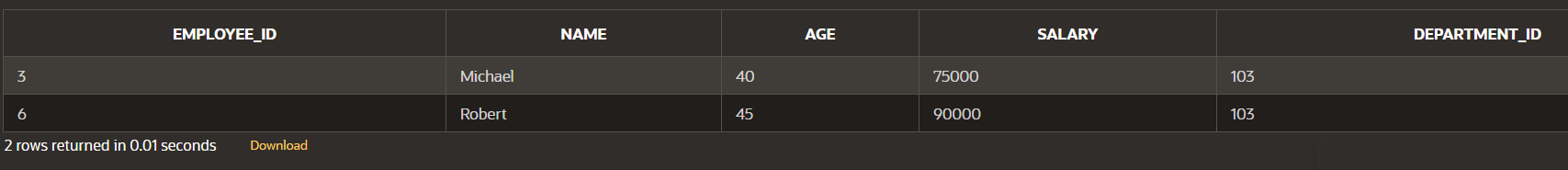
INSERT INTO orders (order\_id, customer\_name, order\_date, order\_amount) VALUES (7, 'Emily', to\_date('2023-05-07','YYYY-MM-DD'), 750);

**Data Querying**

-- Question 1: Retrieve Employee Details

Retrieve all employees whose salary is greater than 60000.

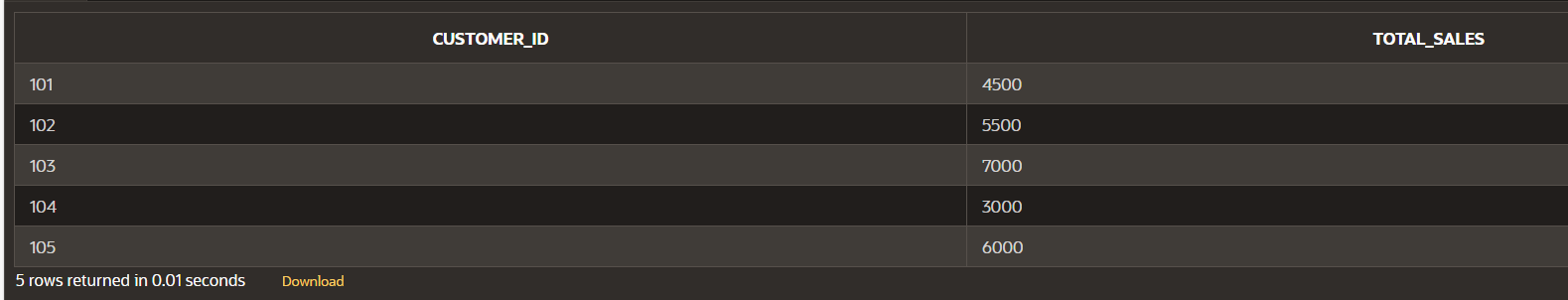
select \* from employees where salary > 60000 order by 1;



-- 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



-- Question 3: Employee Salary in Finance Department

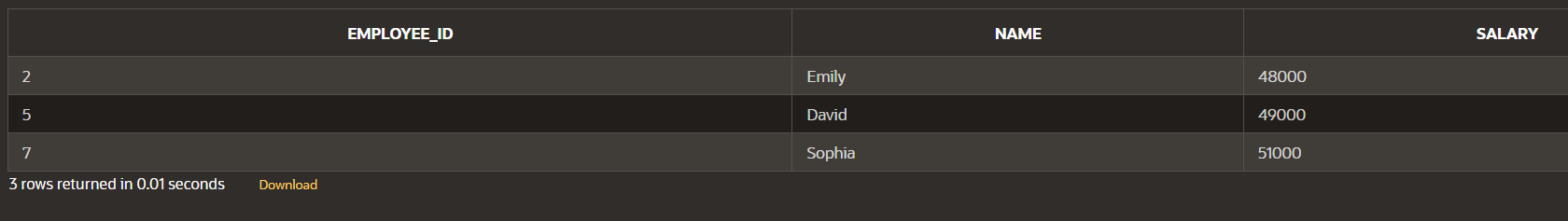
Retrieve the names and salaries of all employees working in the 'Finance' department.

select employee\_id,name,salary

from employees

where department\_id = (select department\_id from departments where department\_name = 'Finance')

order by 1



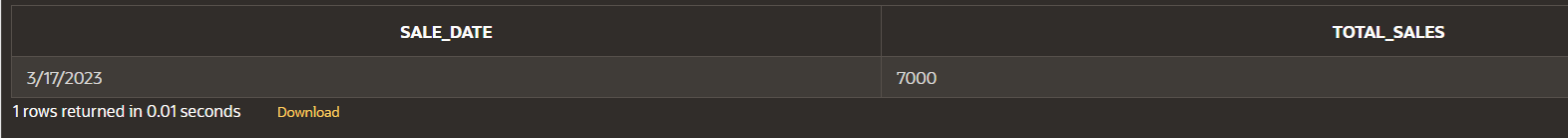
-- Question 4: Total Sales on a Specific Date

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



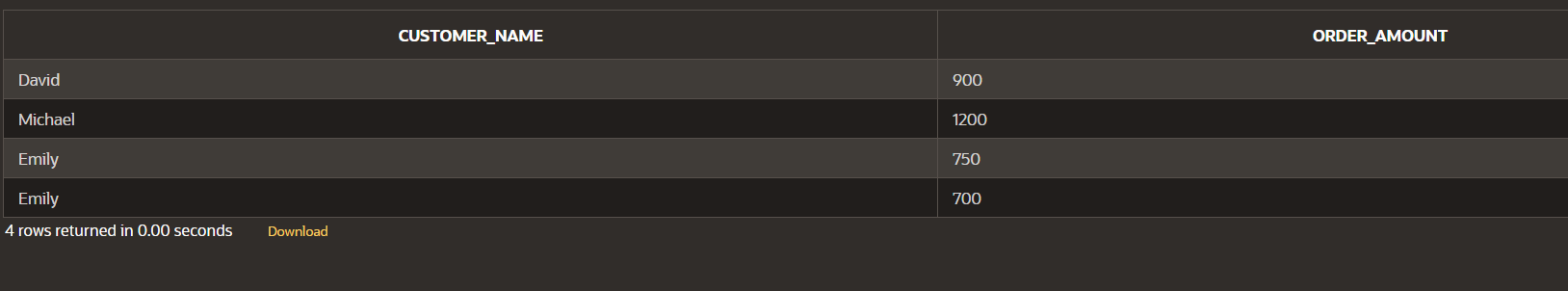
-- Question 5: High-Value Orders

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



**Scenario Based Questions**

1. Employee Salary Analysis

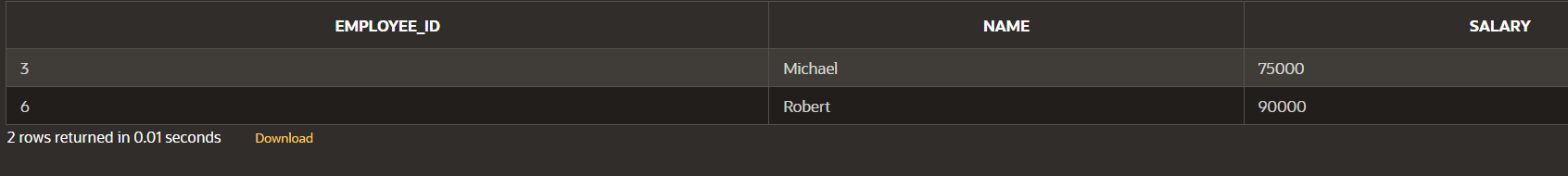
**Question:** Find the names and salaries of employees who earn more than the average salary in the company.

select employee\_id,name, salary

from employees

where salary > (select round(avg(salary),2) from employees)

order by 1

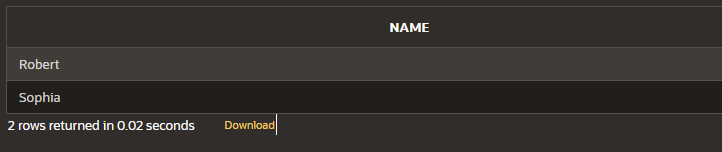


1. Customer Orders without matching records

**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)



1. 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

1. Department Wise Employee Count

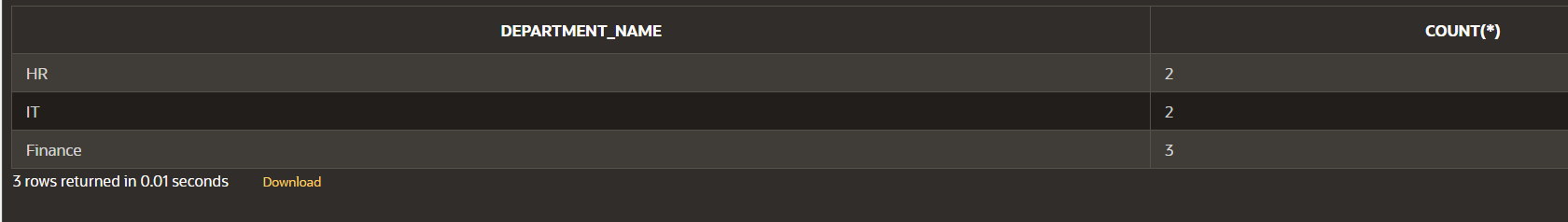
**Question:** List each department name with the1 number of employees working in it.

select b.department\_name, count(\*)

from employees a, departments b

where a.department\_id = b.department\_id

group by b.department\_name



1. Top 3 Highest Sales

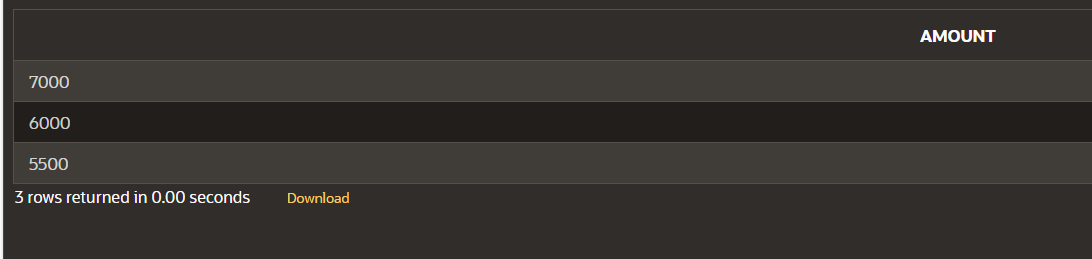
**Question:** Find the top 3 highest sales transactions.

select \* from (

select amount from sales order by 1 desc

)

where rownum <= 3



1. 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

