NAME – NIRBHIK MANDAL

**COGNIZANT WEEK 2 LEARNING**

**1. SQL Exercise - Advanced concepts**

***Exercise 1: Ranking and Window Functions* –**

**Schema SQL CODE –**

CREATE TABLE Products (

ProductID INT,

ProductName VARCHAR(50),

Category VARCHAR(50),

Price DECIMAL(10,2)

);

INSERT INTO Products VALUES

(1, 'Laptop', 'Electronics', 1000.00),

(2, 'Smartphone', 'Electronics', 800.00),

(3, 'Tablet', 'Electronics', 800.00),

(4, 'Headphones', 'Electronics', 200.00),

(5, 'TV', 'Electronics', 1200.00),

(6, 'Sofa', 'Furniture', 700.00),

(7, 'Dining Table', 'Furniture', 900.00),

(8, 'Chair', 'Furniture', 300.00),

(9, 'Bed', 'Furniture', 900.00),

(10, 'Bookshelf', 'Furniture', 400.00);

**QUERY SQL CODE –**

1. **Use ROW\_NUMBER() –**

SELECT

ProductID,

ProductName,

Category,

Price,

ROW\_NUMBER() OVER (PARTITION BY Category ORDER BY Price DESC) AS RowNum

FROM Products;

1. **Use RANK() and DENSE\_RANK()-**

SELECT

ProductID,

ProductName,

Category,

Price,

RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS RankNum

FROM Products;

SELECT

ProductID,

ProductName,

Category,

Price,

DENSE\_RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS DenseRankNum

FROM Products;

1. **Use PARTITION BY Category and ORDER BY Price DESC-**

SELECT \*

FROM (

SELECT

ProductID,

ProductName,

Category,

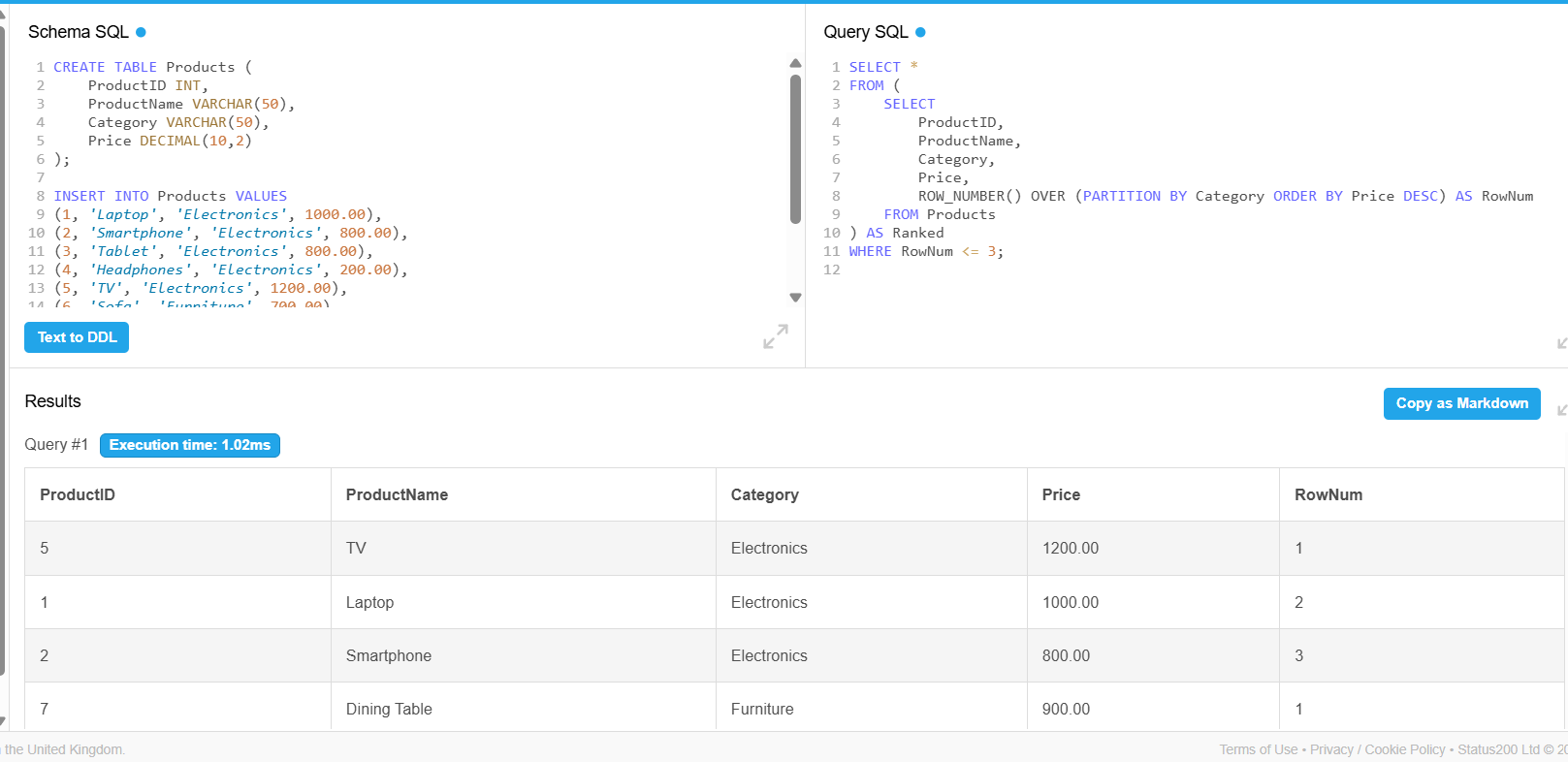
Price,

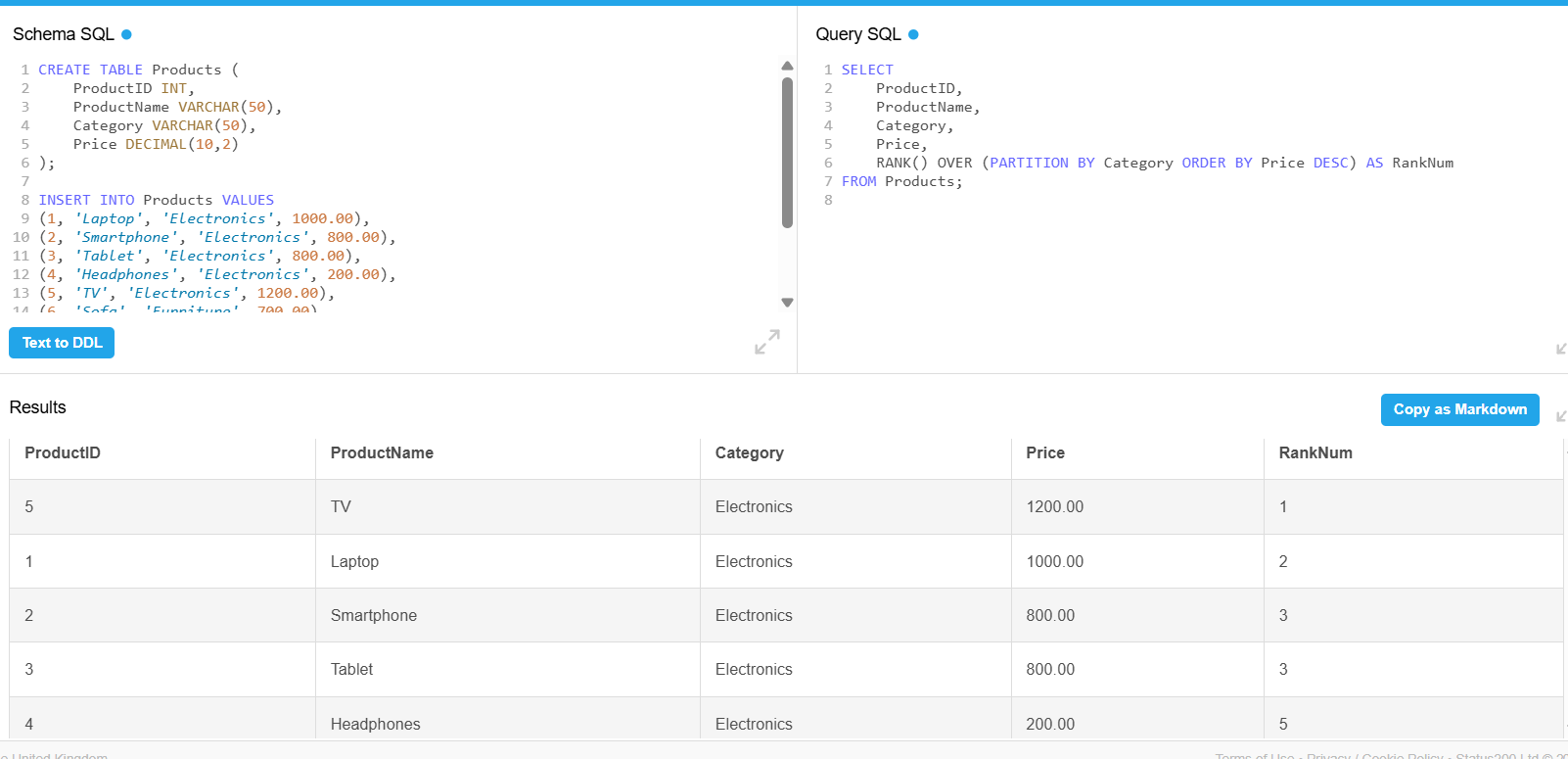
ROW\_NUMBER() OVER (PARTITION BY Category ORDER BY Price DESC) AS RowNum

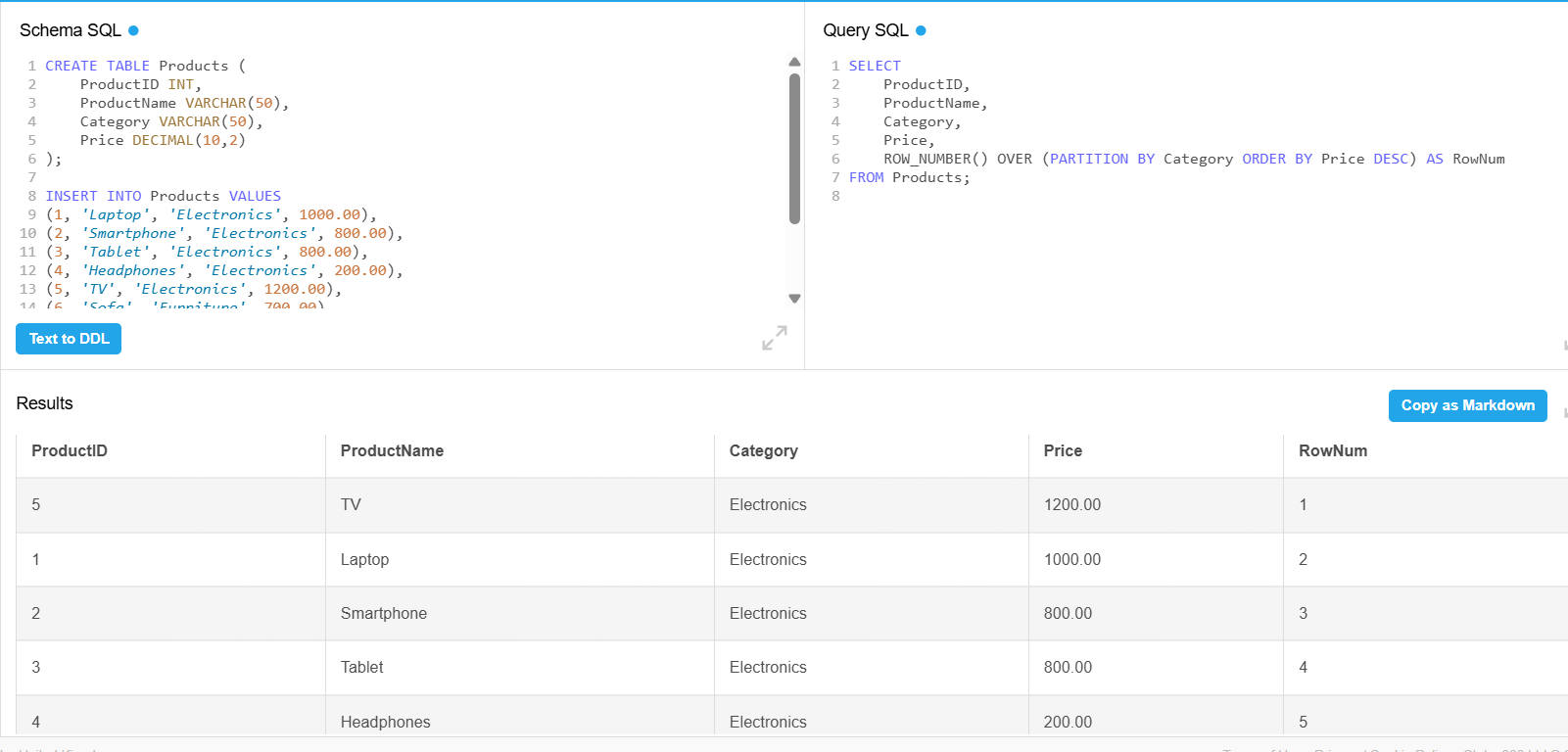
FROM Products

) AS Ranked

WHERE RowNum <= 3;

****

****

****

**4. SQL Exercise - Stored procedure**

**Exercise 1: Create a Stored Procedure –**

**Schema SQL CODE –**

CREATE TABLE Employees (

EmployeeID INT AUTO\_INCREMENT PRIMARY KEY,

FirstName VARCHAR(50),

LastName VARCHAR(50),

DepartmentID INT,

Salary DECIMAL(10,2),

JoinDate DATE

);

INSERT INTO Employees (FirstName, LastName, DepartmentID, Salary, JoinDate) VALUES

('Aunkita', 'Smith', 1, 60000.00, '2022-01-15'),

('Sonal', 'Kumar', 2, 55000.00, '2021-03-10'),

('Pranjal', 'Yadav', 1, 62000.00, '2023-06-01'),

('Nirbhik', 'Singh', 3, 70000.00, '2020-11-20');

**QUERY SQL CODE –**

SET @dept\_id = 1;

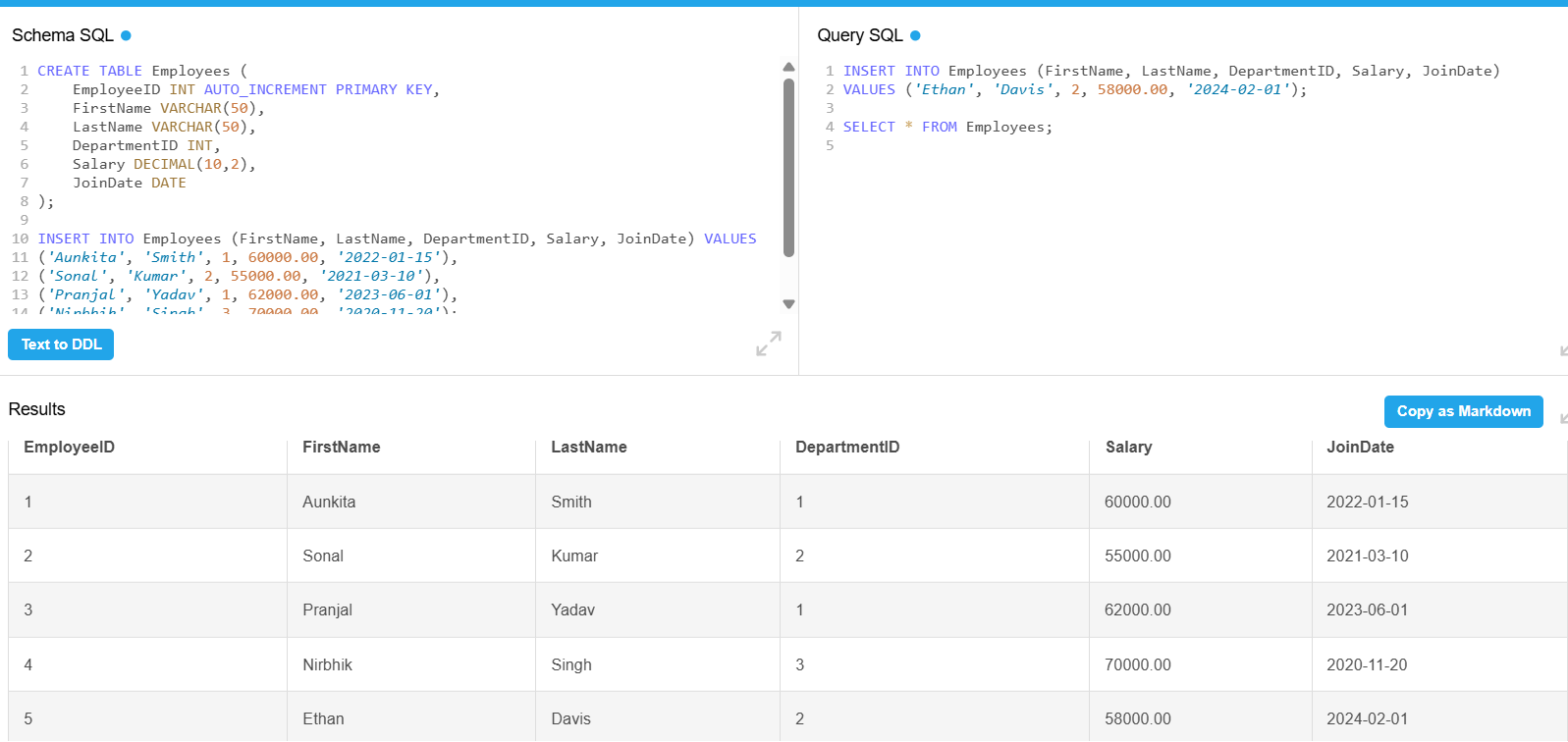
SELECT \* FROM Employees

WHERE DepartmentID = @dept\_id;

INSERT INTO Employees (FirstName, LastName, DepartmentID, Salary, JoinDate)

VALUES ('Ethan', 'Davis', 2, 58000.00, '2024-02-01');

SELECT \* FROM Employees;



**Exercise 5: Return Data from a Stored Procedure-**

**Schema SQL CODE -**

CREATE TABLE Employees (

EmployeeID INT AUTO\_INCREMENT PRIMARY KEY,

FirstName VARCHAR(50),

LastName VARCHAR(50),

DepartmentID INT,

Salary DECIMAL(10,2),

JoinDate DATE

);

INSERT INTO Employees (FirstName, LastName, DepartmentID, Salary, JoinDate) VALUES

('Aarav', 'Sharma', 1, 60000.00, '2022-01-15'),

('Priya', 'Verma', 2, 55000.00, '2021-03-10'),

('Rohan', 'Mehta', 1, 62000.00, '2023-06-01'),

('Sneha', 'Iyer', 3, 70000.00, '2020-11-20'),

('Karan', 'Patel', 1, 58000.00, '2024-02-01');

**QUERY SQL CODE –**

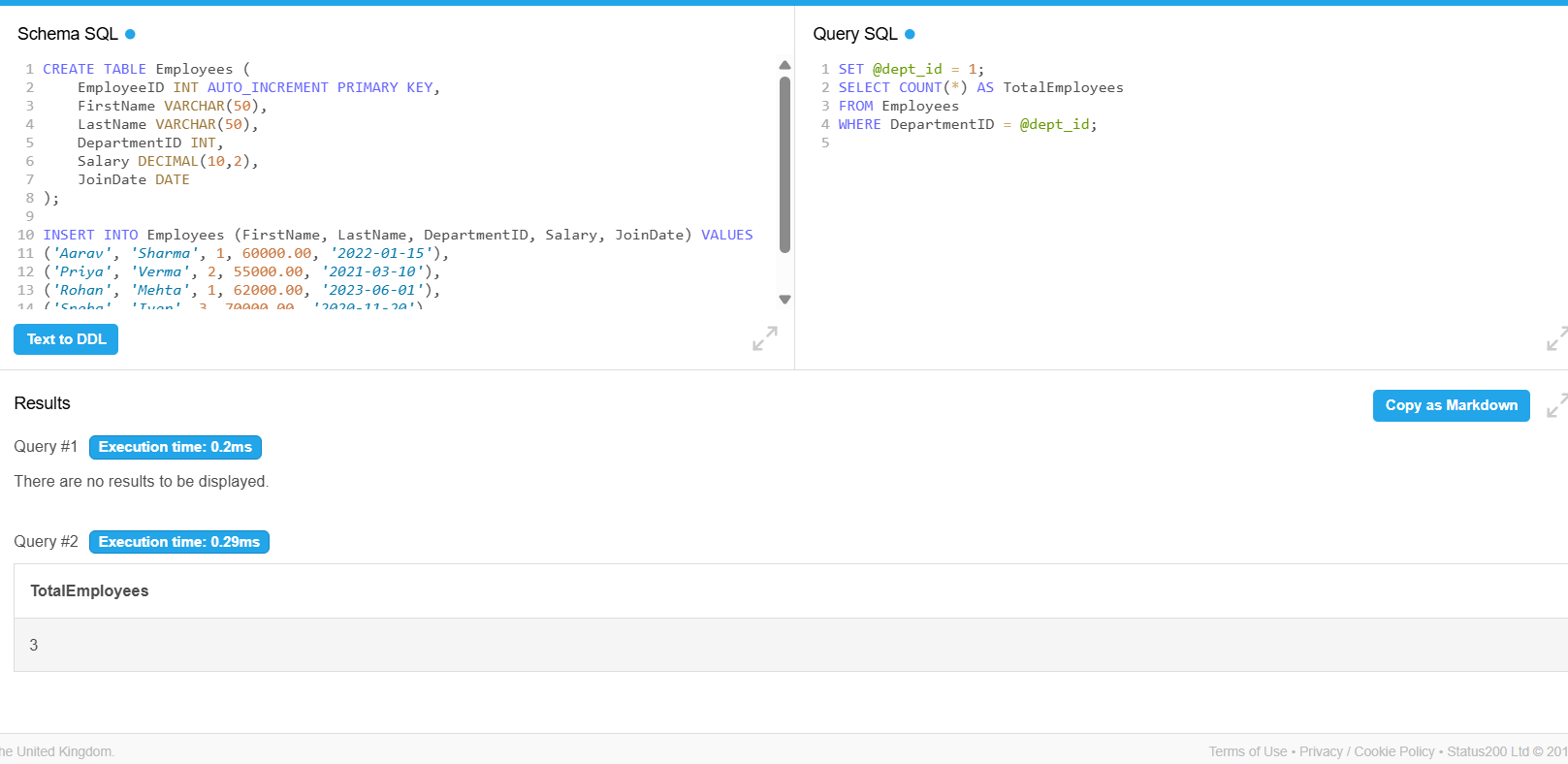
-- Simulate passing a department ID

SET @dept\_id = 1;

-- Return total number of employees in that department

SELECT COUNT(\*) AS TotalEmployees

FROM Employees

WHERE DepartmentID = @dept\_id;