# **Name- Sayandeep Dey (SupersetID:** **6363427)**

# **WEEK – 2 (Handson- Exercises)**

1. **Advance SQL Queries:**

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

**Code:**

1. **Create or Select a Database:**

CREATE DATABASE AdvancedSQL;

GO

USE AdvancedSQL;

GO

1. **ROW\_NUMBER():**

WITH RankedProducts AS (

SELECT

ProductID,

ProductName,

Category,

Price,

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

FROM Products

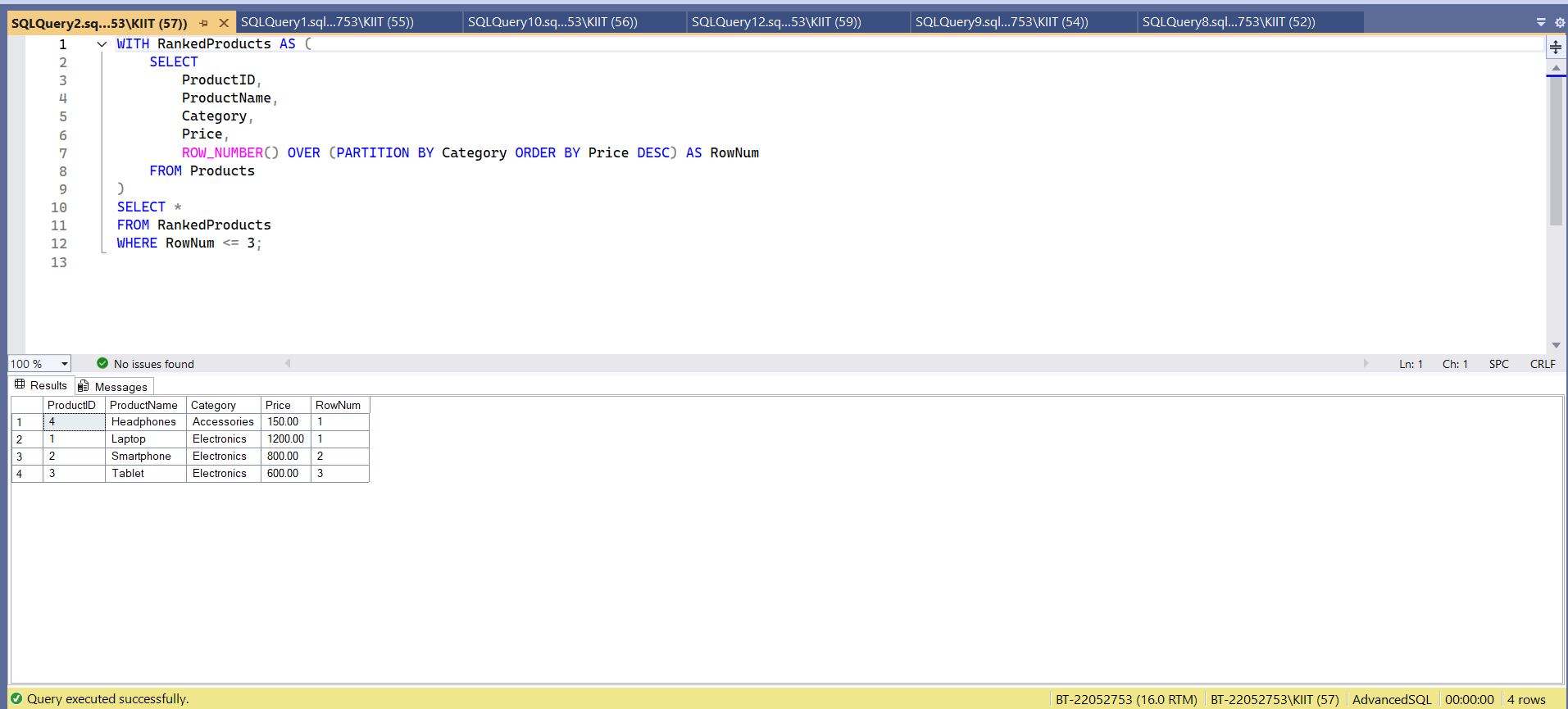
)

SELECT \*

FROM RankedProducts

WHERE RowNum <= 3;

**Output:**



1. **RANK():**

WITH RankedProducts AS (

SELECT

ProductID,

ProductName,

Category,

Price,

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

FROM Products

)

SELECT \*

FROM RankedProducts

WHERE RankNum <= 3;

**Output:**

**A screenshot of a computer

AI-generated content may be incorrect.**

1. **DENSE\_RANK():**

WITH RankedProducts AS (

SELECT

ProductID,

ProductName,

Category,

Price,

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

FROM Products

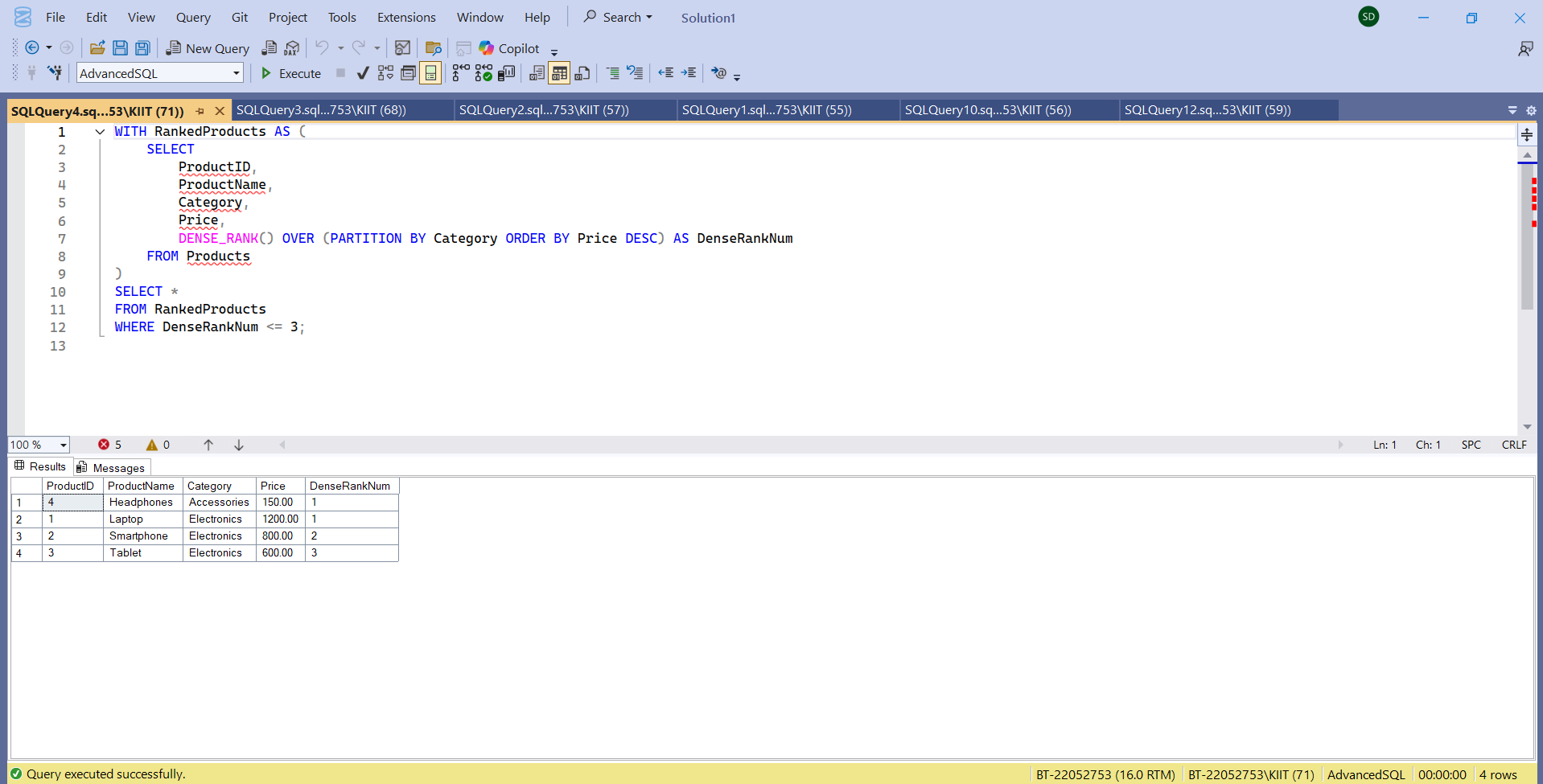
)

SELECT \*

FROM RankedProducts

WHERE DenseRankNum <= 3;

**Output:**



**5. SQL Exercise – Functions**

**Exercise 7: Return Data from a Scalar Function**

**Code:**

**CREATE DATABASE EmployeeManagementSystem;**

**After DB Created Successfully:**

USE EmployeeManagementSystem;

DROP TABLE IF EXISTS Employees;

DROP TABLE IF EXISTS Departments;

CREATE TABLE Departments (

DepartmentID INT PRIMARY KEY,

DepartmentName VARCHAR(100) NOT NULL

);

CREATE TABLE Employees (

EmployeeID INT PRIMARY KEY,

FirstName VARCHAR(50) NOT NULL,

LastName VARCHAR(50) NOT NULL,

DepartmentID INT,

Salary DECIMAL(10,2),

JoinDate DATE,

FOREIGN KEY (DepartmentID) REFERENCES Departments(DepartmentID)

);

INSERT INTO Departments (DepartmentID, DepartmentName) VALUES

(1, 'HR'),

(2, 'IT'),

(3, 'Finance');

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

(1, 'John', 'Doe', 1, 5000.00, '2020-01-15'),

(2, 'Jane', 'Smith', 2, 6000.00, '2019-03-22'),

(3, 'Bob', 'Johnson', 3, 5500.00, '2021-07-01');

**Output:**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Code:**

CREATE FUNCTION fn\_CalculateAnnualSalary (@EmpID INT)

RETURNS DECIMAL(12,2)

AS

BEGIN

DECLARE @AnnualSalary DECIMAL(12,2)

SELECT @AnnualSalary = Salary \* 12

FROM Employees

WHERE EmployeeID = @EmpID

RETURN @AnnualSalary

END;

**Output:**

**A computer screen shot of a computer

AI-generated content may be incorrect.**

SELECT dbo.fn\_CalculateAnnualSalary(1) AS AnnualSalary;

**A screenshot of a computer

AI-generated content may be incorrect.**

SELECT dbo.fn\_CalculateAnnualSalary(1) AS AnnualSalary;

SELECT dbo.fn\_CalculateAnnualSalary(2) AS AnnualSalary;

SELECT dbo.fn\_CalculateAnnualSalary(3) AS AnnualSalary;

**A screenshot of a computer

AI-generated content may be incorrect.**

# **(Extras)**

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

**Code:**

**Create a New Database**

CREATE DATABASE EmployeeDB1;

GO

USE EmployeeDB1;

GO

**Output:**

A screenshot of a computer

AI-generated content may be incorrect.

**Create Departments Table:**

**Create Employees Table:**

**Insert Sample Data into Departments:**

**Insert Sample Data into Employees:**

CREATE TABLE Departments (

DepartmentID INT PRIMARY KEY,

DepartmentName VARCHAR(100)

);

CREATE TABLE Employees (

EmployeeID INT PRIMARY KEY,

FirstName VARCHAR(50),

LastName VARCHAR(50),

DepartmentID INT FOREIGN KEY REFERENCES Departments(DepartmentID),

Salary DECIMAL(10,2),

JoinDate DATE

);

INSERT INTO Departments (DepartmentID, DepartmentName) VALUES

(1, 'HR'),

(2, 'Finance'),

(3, 'IT'),

(4, 'Marketing');

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

VALUES

(1, 'John', 'Doe', 1, 5000.00, '2020-01-15'),

(2, 'Jane', 'Smith', 2, 6000.00, '2019-03-22'),

(3, 'Michael', 'Johnson', 3, 7000.00, '2018-07-30'),

(4, 'Emily', 'Davis', 4, 5500.00, '2021-11-05');

**Output:**

**Created Successfully.**

**Code:**

CREATE PROCEDURE sp\_GetEmployeesByDepartment

@DepartmentID INT

AS

BEGIN

SELECT EmployeeID, FirstName, LastName, DepartmentID, Salary, JoinDate

FROM Employees

WHERE DepartmentID = @DepartmentID;

END;

**Output:**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Code for Output:**

EXEC sp\_GetEmployeesByDepartment @DepartmentID = 1;

**Output:**

**A screenshot of a computer

AI-generated content may be incorrect.**

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

**Code:**

CREATE PROCEDURE sp\_CountEmployeesByDepartment

@DepartmentID INT

AS

BEGIN

SELECT COUNT(\*) AS TotalEmployees

FROM Employees

WHERE DepartmentID = @DepartmentID;

END;

A screenshot of a computer

AI-generated content may be incorrect.

**Output:**

EXEC sp\_CountEmployeesByDepartment @DepartmentID = 1;

A screenshot of a computer

AI-generated content may be incorrect.

EXEC sp\_CountEmployeesByDepartment @DepartmentID = 5;

**A screenshot of a computer

AI-generated content may be incorrect.**

**Exercise 7: Return Data from a Scalar Function:**

**Code:**

CREATE PROCEDURE sp\_UpdateEmployeeSalary

@EmployeeID INT,

@NewSalary DECIMAL(10, 2)

AS

BEGIN

UPDATE Employees

SET Salary = @NewSalary

WHERE EmployeeID = @EmployeeID;

END;

**Output:**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Output:**

EXEC sp\_UpdateEmployeeSalary 1, 5500.00;

**A screenshot of a computer

AI-generated content may be incorrect.**

SELECT \* FROM Employees WHERE EmployeeID = 1;

A screenshot of a computer

AI-generated content may be incorrect.

SELECT \* FROM Employees;

A screenshot of a computer

AI-generated content may be incorrect.