**1.Advanced SQL Exercises for Online Retail Store**

Exercise 1: Ranking and Window Functions

Code:

SELECT \*

FROM (

SELECT \*,

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

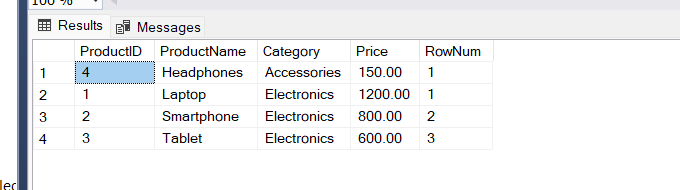
FROM Products

) AS RankedProducts

WHERE RowNum <= 3;

Output:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ProductID | ProductName | Category | Price | RowNum |
| 4 | Headphones | Accessories | 150.00 | 1 |
| 1 | Laptop | Electronics | 1200.00 | 1 |
| 2 | Smartphone | Electronics | 800.00 | 2 |
| 3 | Tablet | Electronics | 600.00 | 3 |



**4. SQL Exercise - Stored procedure**

**Employee Management System SQL Exercises**

Exercise 1: Create a Stored Procedure

Code:

CREATE PROCEDURE sp\_GetEmployeesByDepartment

@DeptID INT

AS

BEGIN

SELECT

E.EmployeeID,

E.FirstName,

E.LastName,

D.DepartmentName,

E.Salary,

E.JoinDate

FROM Employees E

INNER JOIN Departments D ON E.DepartmentID = D.DepartmentID

WHERE E.DepartmentID = @DeptID;

END;

CREATE PROCEDURE sp\_InsertEmployee

@FirstName VARCHAR(50),

@LastName VARCHAR(50),

@DepartmentID INT,

@Salary DECIMAL(10,2),

@JoinDate DATE

AS

BEGIN

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

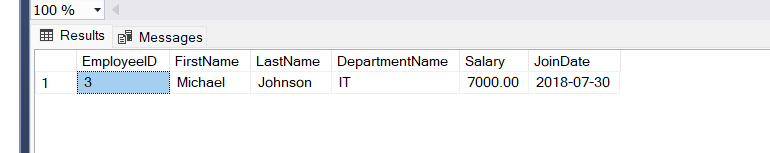
VALUES (@FirstName, @LastName, @DepartmentID, @Salary, @JoinDate);

END;

Output:

EXEC sp\_GetEmployeesByDepartment @DeptID = 3;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EmployeeID | FirstName | LastName | DepartmentName | Salary | JoinDate |
| 3 | Michael | Johnson | IT | 7000.00 | 2018-07-30 |



//Insert new data

EXEC sp\_InsertEmployee

@FirstName = 'Alice',

@LastName = 'Walker',

@DepartmentID = 2,

@Salary = 6200.00,

@JoinDate = '2022-04-10';

Select \* from Employees;

EmployeeID FirstName LastName DepartmentID Salary JoinDate

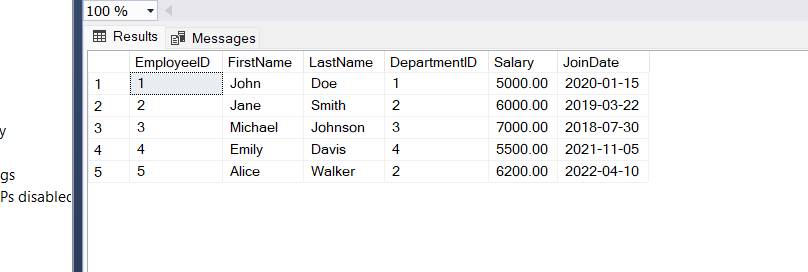
1 John Doe 1 5000.00 2020-01-15

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

5 Alice Walker 2 6200.00 2022-04-10



Exercise 5: Return Data from a Stored Procedure

Code:

-- creating the procedure

CREATE PROCEDURE sp\_GetEmployeeCountByDepartment

@DeptID INT

AS

BEGIN

SELECT COUNT(\*) AS TotalEmployees

FROM Employees

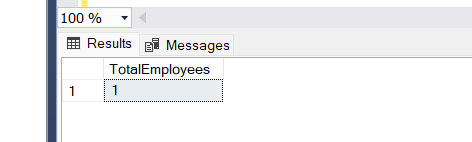
WHERE DepartmentID = @DeptID;

END;

GO

EXEC sp\_GetEmployeeCountByDepartment @DeptID = 3;

Output:



|  |
| --- |
| TotalEmployees |
| 1 |