**Week2-HANDSON**

**Module 3-Advance SQL**

**Exercise 1: Ranking and Window Functions (SQL Exercise - Index)**

**Step1:Create table Categories & Products**

CREATE TABLE Categories (

CategoryID INT PRIMARY KEY,

CategoryName VARCHAR(100)

);

CREATE TABLE Products (

ProductID INT PRIMARY KEY,

ProductName VARCHAR(100),

Price DECIMAL(10, 2),

CategoryID INT,

FOREIGN KEY (CategoryID) REFERENCES Categories(CategoryID)

);

**Step2:Insert data into tables**

INSERT INTO Categories (CategoryID, CategoryName) VALUES

(1, 'Electronics'),

(2, 'Books'),

(3, 'Clothing');

INSERT INTO Products (ProductID, ProductName, Price, CategoryID) VALUES

(1, 'Smartphone', 1800.00, 1),

(2, 'Laptop', 2000.00, 1),

(3, 'Tablet', 650.00, 1),

(4, 'Smartwatch', 620.00, 1),

(5, 'Novel A', 10.00, 2),

(6, 'Novel B', 20.00, 2),

(7, 'Collector Edition', 22.00, 2),

(8, 'T-shirt', 21.00, 3),

(9, 'Jacket', 50.00, 3),

(10, 'Sneakers', 40.00, 3),

(11, 'Cap', 15.00, 3);

**Step3: Ranking Query**

SELECT

c.CategoryName,

p.ProductName,

p.Price,

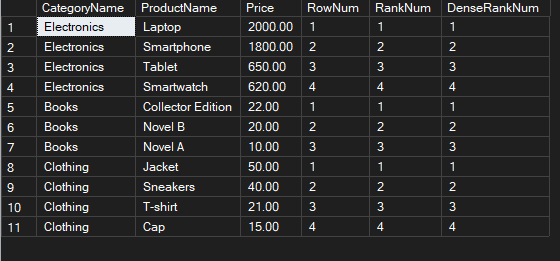
ROW\_NUMBER() OVER (PARTITION BY c.CategoryID ORDER BY p.Price DESC) AS RowNum,

RANK() OVER (PARTITION BY c.CategoryID ORDER BY p.Price DESC) AS RankNum,

DENSE\_RANK() OVER (PARTITION BY c.CategoryID ORDER BY p.Price DESC) AS DenseRankNum

FROM Products p

JOIN Categories c ON p.CategoryID = c.CategoryID;



**Exercise 2: Create a Stored Procedure**

**Step1: Create table Employees**

CREATE TABLE Employees (

EmployeeID INT IDENTITY(1,1) PRIMARY KEY,

FirstName VARCHAR(50),

LastName VARCHAR(50),

DepartmentID INT FOREIGN KEY REFERENCES Departments(DepartmentID),

Salary DECIMAL(10,2),

JoinDate DATE

);

**Step2: Create Insert Procedure**

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;

**Setp3: Execute Procedure**

EXEC sp\_InsertEmployee

@FirstName = 'Adhi',

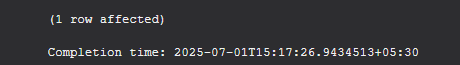
@LastName = 'S',

@DepartmentID = 3,

@Salary = 50000.00,

@JoinDate = '2024-08-18';

**Output:**

****

**Exercise 3: Return Data from a Stored Procedure**

**Step1: create Get Employees by Department Procedure**

CREATE PROCEDURE sp\_GetEmployeesByDepartment

@DepartmentID INT

AS

BEGIN

SELECT

e.EmployeeID,

e.FirstName,

e.LastName,

e.Salary,

e.JoinDate,

d.DepartmentName

FROM Employees e

INNER JOIN Departments d ON e.DepartmentID = d.DepartmentID

WHERE e.DepartmentID = @DepartmentID;

END;

**Step2: Execute Procedure**

EXEC sp\_GetEmployeesByDepartment @DepartmentID = 3;

Output:



**Module4-Nunit & MOQ testing**

Ex1-Hands-on in this document

Step1-Create folder in vs code and execute below in terminal

dotnet new sln -n CustomerApp

dotnet new classlib -n CustomerCommLib

dotnet new nunit -n CustomerComm.Tests

dotnet sln CustomerApp.sln add CustomerCommLib/CustomerCommLib.csproj

dotnet sln CustomerApp.sln add CustomerComm.Tests/CustomerComm.Tests.csproj

cd CustomerComm.Tests

dotnet add reference ../CustomerCommLib/CustomerCommLib.csproj

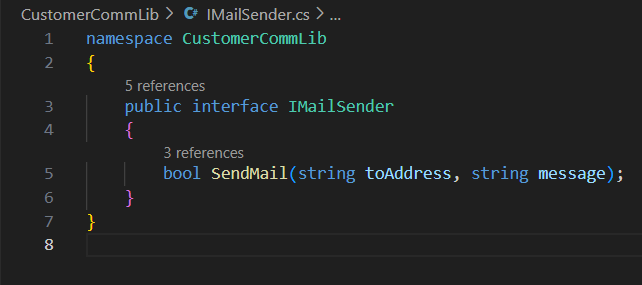
cd CustomerComm.Tests

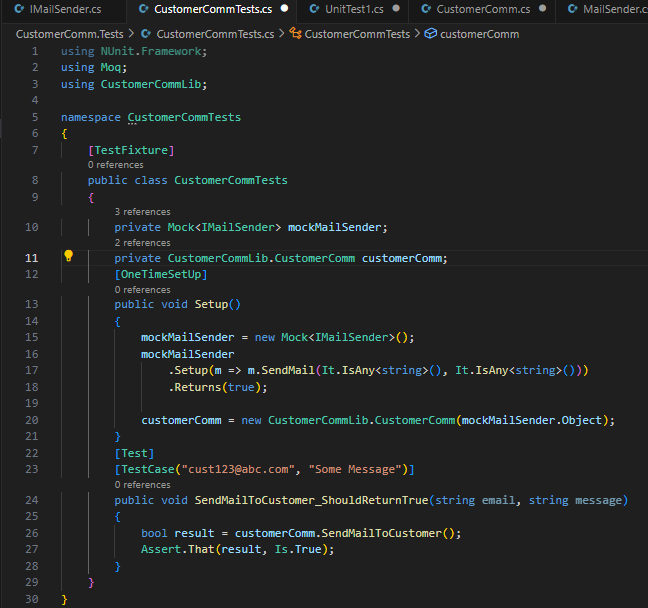
dotnet add package Moq

dotnet add package NUnit

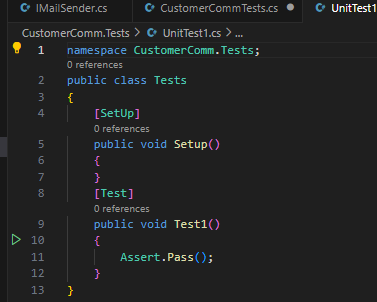
dotnet add package NUnit3TestAdapter

dotnet add package Microsoft.NET.Test.Sdk

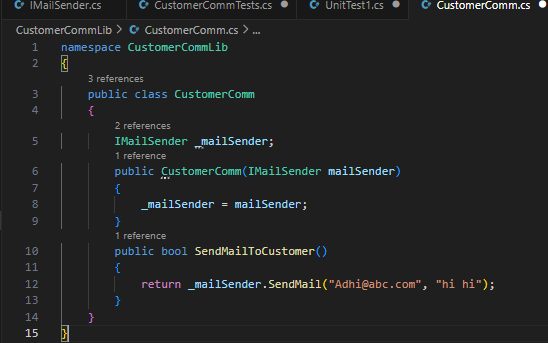
step2-create ImailSender inside CustomerCommLib

Step3-create CustomerCommTests inside CustomercommTests

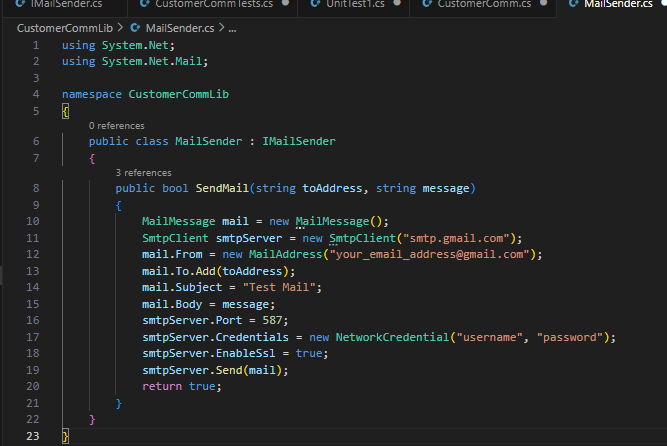
Step4-create Test inside Customercomm.test



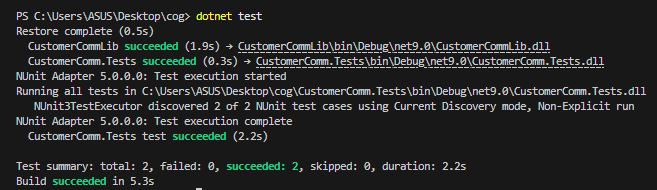
Step5-create CustomerComm inside CustomerCommLib



Step6- create MailSender inside CustomerCommLib



Output- “dotnet test” in terminal



Ex2- Write Testable Code with Moq

Step1- Create folder in vs code and execute below in terminal

dotnet new classlib -n CalcLibrary

dotnet new nunit -n CalcLibrary.Tests

dotnet new sln -n CalculatorApp

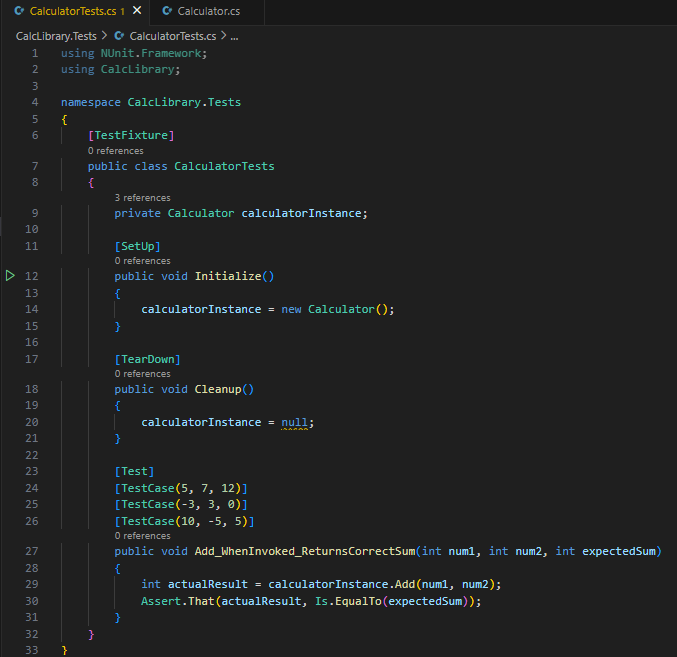
dotnet sln CalculatorApp.sln add CalcLibrary/CalcLibrary.csproj

dotnet sln CalculatorApp.sln add CalcLibrary.Tests/CalcLibrary.Tests.csproj

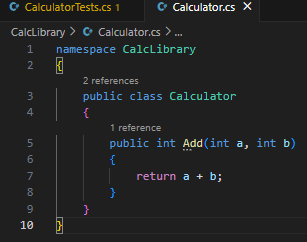
cd CalcLibrary.Tests

dotnet add reference ../CalcLibrary/CalcLibrary.csproj

step2- create CalculatorTests inside CacLibrary.tests



Step3- create Calculator inside CalcLibrary



Output- “dotnet test” in terminal

