**Exercise 1: Create a Stored Procedure**

-- Step 1: Drop table and procedure if they exist

IF OBJECT\_ID('dbo.SalesData', 'U') IS NOT NULL

DROP TABLE dbo.SalesData;

IF OBJECT\_ID('dbo.GetSalesRanking', 'P') IS NOT NULL

DROP PROCEDURE dbo.GetSalesRanking;

GO

-- Step 2: Create table

CREATE TABLE SalesData (

SalesID INT IDENTITY(1,1) PRIMARY KEY,

SalesPerson VARCHAR(50),

Region VARCHAR(50),

SalesAmount INT

);

GO

-- Step 3: Insert sample data

INSERT INTO SalesData (SalesPerson, Region, SalesAmount)

VALUES

('Aniketh', 'North', 5000),

('Surya', 'North', 7000),

('Shashi', 'North', 7000),

('Vidhya', 'South', 4000),

('Nandini', 'South', 6000),

('Jyothi', 'South', 4000),

('Vikram', 'East', 8000),

('Arjun', 'East', 8000),

('Sandeep', 'East', 3000),

('Akhila', 'West', 9000);

GO

-- Step 4: Create stored procedure (start in new batch)

CREATE PROCEDURE GetSalesRanking

AS

BEGIN

SELECT

SalesID,

SalesPerson,

Region,

SalesAmount,

RANK() OVER (PARTITION BY Region ORDER BY SalesAmount DESC) AS RankInRegion,

DENSE\_RANK() OVER (PARTITION BY Region ORDER BY SalesAmount DESC) AS DenseRankInRegion,

ROW\_NUMBER() OVER (PARTITION BY Region ORDER BY SalesAmount DESC) AS RowNumInRegion,

NTILE(4) OVER (ORDER BY SalesAmount DESC) AS Quartile,

LAG(SalesAmount, 1) OVER (PARTITION BY Region ORDER BY SalesAmount DESC) AS PrevSales,

LEAD(SalesAmount, 1) OVER (PARTITION BY Region ORDER BY SalesAmount DESC) AS NextSales,

SUM(SalesAmount) OVER (PARTITION BY Region) AS TotalRegionSales,

AVG(SalesAmount \* 1.0) OVER (PARTITION BY Region) AS AvgRegionSales

FROM SalesData

ORDER BY Region, SalesAmount DESC;

END;

GO

