Challenge 4 Stored Procedures and Functions

This challenge consists of **three exercises.** Its' purpose is to confirm your understanding of procedures, passing parameters, scaler and table functions. Remember to indent your code properly and that all key words should be in UPPER case. Good luck!

Exercise 1: Create and call a Stored Procedure

1. Create a procedure called **usp_AddOne** that requires two parameters

   Parameter 1: **INPUT** parameter, **INT** datatype

   Parameter 2: **OUTPUT** parameter, **INT** datatype. Set this to the value of the **INPUT** parameter plus one.

2. Create an anonymous block that will execute the procedure and then print the value of the output parameter(should be one more than what you sent to the function).

3. Don't forget to place a comment at the top with author, date, description for all procedures and functions.

Exercise 2: Create and call a Scalar function

1. Create a Function called **ufn_ConcatenateTwoString**

   - It should take in two **NCHAR** parameters of size **25** and concatenates them together with a **one** space between them and **RETURN**s that value.
   - The returned value should be in the form of a declared variable of a type **NVARCHAR**

2. Test the function by running it in an anonymous block that uses a PRINT statement

- **NOTE:** You will need to declare three variable in your anonymous block to do this
  - **Example:**
    - Parameter 1: "FirstName"
    - Parameter 2: "LastName"
    - Returns: "FirstName LastName"

Exercise 3: Create and call a Table function

1. Create a Table Function called **ufn_GetMovieRating_*FirstName_LastName*** (where ***FirstName_LastName*** is replaced with your first and last name)

   - It should take in one **NCHAR** parameter of size **50** that will be the name of a movie in the database
   - The return columns should include the name of the movie, its' RatingId and Description

2. Test the table function using a **SELECT** statement

   - **NOTE:** The function should ignore the case of the passed parameter.
     - **Example:** a result set should be returned regardless if the parameter passed is the move name **rocky**, **ROCKY** *or* **rOCky**