

Guidelines and instructions:

- This is a graded assignment, you can collaborate to understand the concepts, but plagiarism is not allowed. You are directed to complete all tasks individually. Any case of plagiarism would result in **0 marks**.
- Please follow the coding conventions
 - Indent your code properly
 - Use meaningful variable and method names, mind the usage of *PascalCase* and *camelCase* notation
 - Comment your code properly
- You are requested to
 - Create a new branch named **Assignment-02** in the repository you've created in the last assignment
 - If you have not created your repository then create one private repository on GitHub on this pattern: **Rollnumber-ES** (e.g., **BITF20M001-ES**)
 - Send GitHub link of your assignment at this email: bsef20m021@pucit.edu.pk
 - Subject line: '<Roll Number>-Assignment-02' (e.g. BITF20M001-Assignment-02)
- The submission deadline of this assignment is **Thursday, 5 October, 2023 10:14:59 AM**
- This assignment should not take more than 90 minutes to complete
- Please note that the assignments submitted after the deadline will not be entertained

Happy coding 😊

Please write the code for the following statements:

1. String concatenation
Write a method that asks the user for their first name and last name as separate inputs. Then, concatenate the two strings to create a full name. Finally, display the full name as output.
2. Substring fetching
You are given a string containing a sentence. Write a method that extracts and displays the last 5 characters of the sentence as a substring.
3. String interpolation
Write a method that defines two variables: **'temperature'** (double) and **'city'** (string). Ask the user for the current temperature and the name of their city. Then, use string interpolation to display a message like **"The temperature in [city] is [temperature] degrees Celsius."**
4. Array declaration and initialization
Create an integer array named **'numbers'** that can hold 5 elements. Initialize the array with the values 1, 2, 3, 4, and 5. Then, print all the elements of the array to the console.
5. Array iteration
 - a. Using for loop
Given an array of strings called **'fruits'** containing the names of fruits, write a method to iterate through the array using a for loop and print each fruit's name on a separate line.
 - b. Using foreach loop
Define a string array called **'colors'** containing the names of colors. Write a method to iterate through the array using foreach loop and display each color's name followed by a comma and a space (e.g., "Red, Blue, Green, ").

6. Sum of array elements
Create an array of integers named **'scores'** with 10 elements. Initialize the array with random test scores (e.g., 85, 92, 78, 95, ...). Write a method to calculate and display the sum of all the test scores in the array. (Use do while loop)
7. Finding the maximum value
Given an integer array **'values'**, write a method to find and display the maximum value in the array using a for loop. (Use while loop)
8. Array reversal
Write a method that takes an integer array **'numbers'** and reverses its elements in place (i.e., without creating a new array). After the reversal, print the reversed array to the console. (Use foreach loop)
9. Boxing
 - a. Integer
Write a method that declares an integer variable **'x'** with a value of 42. Perform boxing to convert the integer into an object. Then, unbox the **object** and store the result in a new integer variable **'y'**. Finally, display the value of **'y'** to the console.
 - b. Double
Create a method that declares a double variable **'doubleValue'** with a value of 3.14159. Perform boxing to convert the double into an object. Then, unbox the object and store the result in a new double variable **'unboxedValue'**. Display the value of **'unboxedValue'** to the console.
10. Unboxing
 - a. Define an array of integers called **'numbers'** with a few elements. Write a program that uses a loop to iterate through the array. In each iteration, perform the following:
 - i. Box the current integer value.
 - ii. Unbox the boxed object and store it in a new integer variable.
 - iii. Calculate the square of the unboxed integer.
 - iv. Display both the original integer and its squared value in each iteration.
 - b. Use a generic **List** to store values of different value types (e.g., int, double, char). Perform boxing and unboxing operations while adding and retrieving elements from the list. Display the elements and their types to ensure successful operations.
11. Dynamic variables
 - a. Use the **dynamic** keyword to declare a variable named **'myVariable'**. Assign an integer value (e.g., 42) to this variable, display it on the console. Then assign string value (e.g., "Hello, Dynamic!") to this variable and display it on the console.
 - b. Declare a **'dynamic'** variable name **'myVariable2'** and assign it an integer value and use **GetType()** method to display type of the variable on the console, then assign this variable a double value and use **GetType()** method of to print type of the variable on the console, do the same activity with **DateTime** and **String** values.