Part01

- **Problem**: Write a program to:
  - Accept a string input from the user.
  - Convert it to an integer using both `int.Parse` and `Convert.ToInt32`.
  - Handle potential exceptions using a `try-catch` block.
- **Question**: What is the difference between `int.Parse` and `Convert.ToInt32` when handling `null` inputs?

- **Problem**: Write a program that:
  - Prompts the user to input a number.
  - Uses `int.TryParse` to check if the input is a valid integer.
  - If valid, print the number; otherwise, print an error message.
- **Question**: Why is `TryParse` recommended over `Parse` in user-facing applications?

- **Problem**: Implement a program to:
  - Declare an `object` variable.
  - Assign it different data types (e.g., `int`, `string`, `double`).
  - Print the `GetHashCode()` of each assignment.
- **Question**: Explain the real purpose of the `GetHashCode()` method.

- **Problem**: Demonstrate how changing one reference affects another when both point to the same object. Use the following steps:
  - Create an `object` and assign it a value.
  - Create a second reference to the same object.
  - Modify the value of the object using one reference and print the value using the other.
- **Question**: What is the significance of reference equality in .NET?

- **Problem**: Write a program to:
  - Declare a string and modify it by concatenating additional text "Hi Willy".
  - Print the `GetHashCode()` before and after modification.
- **Question**: Why string is immutable in C# ?

- **Problem**: Create a program to:

- o Use `StringBuilder` to append text to a string "Hi Willy".
- o Print the `GetHashCode()` of the `StringBuilder` instance before and after the modification.
- **Question**: How does `StringBuilder` address the inefficiencies of string concatenation?

- **Question**: Why is `StringBuilder` faster for large-scale string modifications?

- **Problem**: Create a program to:
  - o Accept two integer inputs from the user.
  - o Display the sum using all three methods "Sum is input1+input2":
    - Concatenation (+ operator)
    - Composite formatting (`string.Format`)
    - String interpolation (`$`)
- **Question**: Which string formatting method is most used and why?

- **Problem**: Create a program using `StringBuilder` to:
  - o Append text.
  - o Replace a substring.
  - o Insert a string at a specific position.
  - o Remove a portion of text.
- **Question**: Explain how `StringBuilder` is designed to handle frequent modifications compared to strings.

Part02

1- LinkedIn article about string immutability

2- What's Enum data type, when is it used? And name three common built_in enums used frequently?

3- what are scenarios to use string Vs StringBuilder?

Part03 Bonus

4- self study report

5- what meant by user defined constructor and its role in initialization

6- compare between Array and Linked List