Temperature Conversion System Using System Exceptions

Title

Temperature Conversion Utility with Exception Handling

Question No : 1 / 1

Problem Statement

Write a C# console application that reads a temperature value and a conversion type from the user.
The conversion type indicates whether to convert the temperature from **Fahrenheit to Celsius** or from **Celsius to Fahrenheit**.

The program should output the converted temperature rounded to **two decimal places**.

If the user provides **invalid input** (non-numeric temperature or invalid conversion type), the program should handle the error gracefully and display an appropriate error message.

Use **SystemException** to catch the exception message.
Write the solution within the **Program.cs** file.

Requirements

1. Input

- A numeric value representing the temperature.
- A string representing the conversion type:
    - **F** → Fahrenheit to Celsius
    - **C** → Celsius to Fahrenheit

2. Output

- Converted temperature rounded to two decimal places.
- Error message for invalid input.

3. Conversion Formula

- Fahrenheit to Celsius
  ```
  celsius = (temperature - 32) * 5 / 9
  ```
- Celsius to Fahrenheit
  ```
  fahrenheit = (temperature * 9 / 5) + 32
  ```

Input Format

- First line: Temperature value of type **double**
- Second line: Conversion type (**C** or **F**) of type **string**

Output Format

- If input is **F**
- `Temperature in Celsius: {value}`
- If input is **C**
- `Temperature in Fahrenheit: {value}`
- If conversion type is invalid
- `Invalid conversion type. Please enter 'F' or 'C'.`
- If temperature input is invalid
- `Error: Invalid input provided.`
- `Exception Message: {errorMessage}`

Test Cases

Test Case ID	Input	Expected Output
TC01	`100, F`	Temperature in Celsius: 37.78
TC02	`30, C`	Temperature in Fahrenheit: 86.00
TC03	`abc`	Error: Invalid input provided. Exception Message: Input string was not in a correct format.
TC04	`10, g`	Invalid conversion type. Please enter 'F' or 'C'.
TC05	`-40, F`	Temperature in Celsius: -40.00

Sample Input 1

```
100
F
```

Sample Output 1

```
Temperature in Celsius: 37.78
```

Sample Input 2

```
30
C
```

Sample Output 2

```
Temperature in Fahrenheit: 86.00
```

Sample Input 3

```
abc
```

Sample Output 3

```
Error: Invalid input provided.
Exception Message: Input string was not in a correct format.
```

Sample Input 4

```
10
g
```

Sample Output 4

```
Invalid conversion type. Please enter 'F' or 'C'.
```

C# Solution (Program.cs)

```csharp
class InvalidInputException : Exception
{
    public InvalidInputException(string message) :
base(message){}
}
class InvalidTypeException : Exception
{
    public InvalidTypeException(string message) :
base(message){}
}


class Program
{
    static void Main()
    {
        try
        {
            Console.Write("Enter temperature value: ");
            string input = Console.ReadLine()??"";
```

```csharp
if (!double.TryParse(input, out double value))
{
    throw new InvalidInputException("Error: Invalid
input provided.\nException Message: Input string was not in
a correct format.");
}


Console.Write("Enter conversion type (C/F): ");
string conversion = (Console.ReadLine() ??
"").Trim().ToLower();


if (string.IsNullOrWhiteSpace(conversion))
{
    throw new InvalidTypeException("Conversion
type cannot be empty.");
}


if (conversion != "c" && conversion != "f")
{
```

```csharp
            throw new InvalidTypeException("Invalid
conversion type. Please enter 'F' or 'C'.");

        }


        if (conversion == "f")

        {

            double result = (value - 32) * 5 / 9;

            Console.WriteLine("Temperature in Celsius:
{0:F}",result);

        }

        else

        {

            double result = (value * 9 / 5) + 32;

            Console.WriteLine("Temperature in Fahrenheit:
{0:F}",result);

        }

    }

    catch (InvalidInputException ex)

    {

        Console.WriteLine(ex.Message);
```

```csharp
        }
        catch (InvalidTypeException ex)
        {
            Console.WriteLine(ex.Message);
        }
    }
}
```