```
using System;
namespace backendDay2
{
    internal class Program
    {
        static void Main(string[] args)
        {
            bool useAgain = true;
            int calculatorOption = 0;
            int continueUsing = 0;
            int confirmExit = 1;
            bool valid;

            do
            {
                valid = false;
                Console.WriteLine("---------\t Advance Calculator \t---------");
                Console.WriteLine("1. Basic Calculator");
                Console.WriteLine("2. Continuous Calculator");
                Console.WriteLine("3. Scientific Calculator");
                Console.WriteLine("4. Exit");
                Console.Write("Please select an option: ");

                while (!valid)
                {
                    if (!int.TryParse(Console.ReadLine(), out calculatorOption)
|| calculatorOption < 1 || calculatorOption > 4)
                    {
                        Console.Write("Invalid input. Please enter a number from
1 to 4: ");
                    }
                    else
                    {
                        valid = true;
                    }
                }

                switch (calculatorOption)
                {
                    case 1:
                        do
                        {
                            Console.WriteLine("\nWelcome to the Basic
Calculator!");
                            double num1 = 0, num2 = 0;
```

```
                            char operation;
                            valid = false;
                            continueUsing = 0;

                            while (!valid)
                            {
                                Console.Write("Enter first number: ");
                                if (double.TryParse(Console.ReadLine(), out
num1))
                                    valid = true;
                                else
                                    Console.WriteLine("Invalid input. Try
again.");
                            }

                            valid = false;
                            while (!valid)
                            {
                                Console.Write("Enter second number: ");
                                if (double.TryParse(Console.ReadLine(), out
num2))
                                    valid = true;
                                else
                                    Console.WriteLine("Invalid input. Try
again.");
                            }

                            Console.Write("Choose operation (+, -, *, /): ");
                            operation = Console.ReadKey().KeyChar;
                            Console.WriteLine();

                            switch (operation)
                            {
                                case '+':
                                    Console.WriteLine($"Result: {num1 + num2}");
                                    break;
                                case '-':
                                    Console.WriteLine($"Result: {num1 - num2}");
                                    break;
                                case '*':
                                    Console.WriteLine($"Result: {num1 * num2}");
                                    break;
                                case '/':
                                    if (num2 != 0)
```

```
                                    Console.WriteLine($"Result: {num1 /
num2}");
                                else
                                    Console.WriteLine("❌ Cannot divide by
zero.");
                                break;
                            default:
                                Console.WriteLine("❌ Invalid operation.");
                                break;
                        }

                        // Ask to continue
                        Console.Write("Do you want to calculate again? Type 1
for YES, 0 for NO: ");
                        while (!int.TryParse(Console.ReadLine(), out
continueUsing) || (continueUsing != 0 && continueUsing != 1))
                        {
                            Console.Write("Invalid input. Please enter 1 or
0: ");
                        }

                    } while (continueUsing == 1);

                    Console.WriteLine("\nThank you for using the Basic
Calculator!\n");

                    useAgain = true;
                    break;

                case 2:
                    do
                    {
                        double num1 = 0, num2 = 0;
                        valid = false;
                        continueUsing = 0;
                        int count = 0;

                        Console.WriteLine("Welcome to the Continuous
Calculator!");
                        Console.WriteLine("In this calculator you can
continuously perform basic math operations and input number!");
                        while (!valid)
                        {
                            Console.Write("Enter first number: ");
```

```
                                if (double.TryParse(Console.ReadLine(), out
num1))
                                    valid = true;
                                else
                                    Console.WriteLine("Invalid input. Try
again.");
                            }

                            while(count < 10)
                            {
                                Console.Write("\nEnter an operation (+, -, *, /)
or type 'exit' to stop: ");
                                string operation =
Console.ReadLine().Trim().ToLower();

                                if (operation == "exit")
                                {
                                    Console.WriteLine("Exiting Continuous
Calculator.");
                                    break;
                                }

                                valid = false;
                                while (!valid)
                                {
                                    Console.Write("Enter first number: ");
                                    if (double.TryParse(Console.ReadLine(), out
num2))
                                        valid = true;
                                    else
                                        Console.WriteLine("Invalid input. Try
again.");
                                }

                                switch (operation)
                                {
                                    case "+":
                                        Console.WriteLine($"Result: {num1 +
num2}");
                                        break;
                                    case "-":
                                        Console.WriteLine($"Result: {num1 -
num2}");
                                        break;
                                    case "*":
```

```
                                    Console.WriteLine($"Result: {num1 *
num2}");
                                    break;
                                case "/":
                                    if (num2 != 0)
                                        Console.WriteLine($"Result: {num1 /
num2}");
                                    else
                                        Console.WriteLine("❌ Cannot divide
by zero.");
                                    break;
                                default:
                                    Console.WriteLine("❌ Invalid
operation.");
                                    break;
                            }
                        }

                        Console.Write("Do you want to calculate again? Type 1
for YES, 0 for NO: ");
                        while (!int.TryParse(Console.ReadLine(), out
continueUsing) || (continueUsing != 0 && continueUsing != 1))
                        {
                            Console.Write("Invalid input. Please enter 1 or
0: ");
                        }

                    } while (continueUsing == 1);

                    Console.WriteLine("\nThank you for using the Continuous
Calculator!\n");
                    useAgain = true;
                    break;

                case 3:
                    do {
                        valid = false;
                        continueUsing = 0;
                        double num1 = 0, num2 = 0;
                        double inputNumber = 0;
                        string[] operations = { "add", "subtract",
"multiply", "divide", "square", "cube", "sqrt" };
                        Console.WriteLine("\nWelcome to Scientific
Calculator!");
                        Console.WriteLine("Available Operations:");
```

```
                            foreach (string op in operations)
                            {
                                Console.WriteLine("- "+ op);
                            }
                            Console.Write("Enter the operation you want to
perform: ");
                            string selectedOperation =
Console.ReadLine().ToLower();

                            switch (selectedOperation)
                            {
                                case "add":
                                    valid = false;
                                    while (!valid)
                                    {
                                        Console.Write("Enter first number: ");
                                        if (double.TryParse(Console.ReadLine(),
out num1))
                                            valid = true;
                                        else
                                            Console.WriteLine("Invalid input. Try
again.");
                                    }

                                    valid = false;
                                    while (!valid)
                                    {
                                        Console.Write("Enter second number: ");
                                        if (double.TryParse(Console.ReadLine(),
out num2))
                                            valid = true;
                                        else
                                            Console.WriteLine("Invalid input. Try
again.");
                                    }
                                    Console.WriteLine($"Result: {num1 + num2}");
                                    break;
                                case "subtract":
                                    valid = false;
                                    while (!valid)
                                    {
                                        Console.Write("Enter first number: ");
                                        if (double.TryParse(Console.ReadLine(),
out num1))
                                            valid = true;
```

```
                                    else
                                        Console.WriteLine("Invalid input. Try
again.");
                                }

                                valid = false;
                                while (!valid)
                                {
                                    Console.Write("Enter second number: ");
                                    if (double.TryParse(Console.ReadLine(),
out num2))
                                        valid = true;
                                    else
                                        Console.WriteLine("Invalid input. Try
again.");
                                }
                                Console.WriteLine($"Result: {num1 - num2}");
                                break;
                            case "multiply":
                                valid = false;
                                while (!valid)
                                {
                                    Console.Write("Enter first number: ");
                                    if (double.TryParse(Console.ReadLine(),
out num1))
                                        valid = true;
                                    else
                                        Console.WriteLine("Invalid input. Try
again.");
                                }

                                valid = false;
                                while (!valid)
                                {
                                    Console.Write("Enter second number: ");
                                    if (double.TryParse(Console.ReadLine(),
out num2))
                                        valid = true;
                                    else
                                        Console.WriteLine("Invalid input. Try
again.");
                                }
                                Console.WriteLine($"Result: {num1 * num2}");
                                break;
                            case "divide":
```

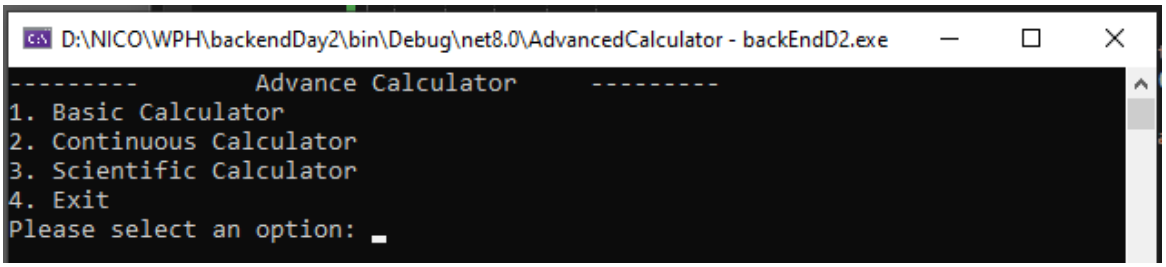
```
                                    valid = false;
                                    while (!valid)
                                    {
                                        Console.Write("Enter first number: ");
                                        if (double.TryParse(Console.ReadLine(),
out num1))
                                            valid = true;
                                        else
                                            Console.WriteLine("Invalid input. Try
again.");
                                    }

                                    valid = false;
                                    while (!valid)
                                    {
                                        Console.Write("Enter second number: ");
                                        if (double.TryParse(Console.ReadLine(),
out num2))
                                            valid = true;
                                        else
                                            Console.WriteLine("Invalid input. Try
again.");
                                    }
                                    if (num2 != 0)
                                        Console.WriteLine($"Result: {num1 /
num2}");
                                    else
                                        Console.WriteLine("❌ Cannot divide by
zero.");
                                    break;
                                case "square":
                                    valid = false;
                                    while (!valid)
                                    {
                                        Console.Write("Enter a number: ");
                                        if (double.TryParse(Console.ReadLine(),
out inputNumber))
                                            valid = true;
                                        else
                                            Console.WriteLine("Invalid input.
Please enter a valid number.");
                                    }
                                    Console.WriteLine($"Result: {inputNumber}² =
{Math.Pow(inputNumber, 2)}");
                                    break;
```

```
                                case "cube":
                                    valid = false;
                                    while (!valid)
                                    {
                                        Console.Write("Enter a number: ");
                                        if (double.TryParse(Console.ReadLine(),
out inputNumber))
                                            valid = true;
                                        else
                                            Console.WriteLine("Invalid input.
Please enter a valid number.");
                                    }
                                    Console.WriteLine($"Result: {inputNumber}^3 =
{Math.Pow(inputNumber, 3)}");
                                    break;
                                case "sqrt":
                                    valid = false;
                                    while (!valid)
                                    {
                                        Console.Write("Enter a number: ");
                                        if (double.TryParse(Console.ReadLine(),
out inputNumber))
                                            valid = true;
                                        else
                                            Console.WriteLine("Invalid input.
Please enter a valid number.");
                                    }
                                    if (inputNumber >= 0)
                                        Console.WriteLine($"Result:
√{inputNumber} = {Math.Sqrt(inputNumber)}");
                                    else
                                        Console.WriteLine("❌ Cannot compute
square root of a negative number.");
                                    break;
                                default:
                                    Console.WriteLine("❌ Invalid scientific
operation.");
                                    break;
                            }
                            Console.Write("Do you want to calculate again? Type 1
for YES, 0 for NO: ");
                            while (!int.TryParse(Console.ReadLine(), out
continueUsing) || (continueUsing != 0 && continueUsing != 1))
                            {
```

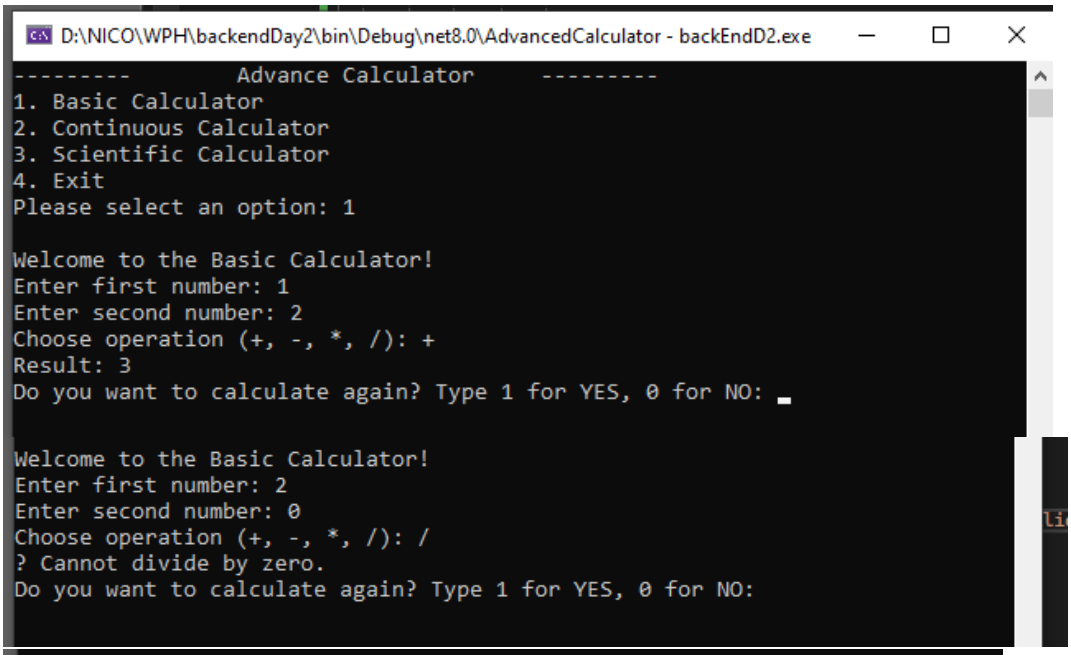
```
                                Console.Write("Invalid input. Please enter 1 or
0: ");
                            }

                        } while (continueUsing == 1);

                        Console.WriteLine("\nThank you for using the Scientific
Calculator!\n");
                        useAgain = true;
                        break;

                    case 4:
                        Console.Write("Are you sure you want to exit Advance
Calculator? Type 1 for YES and 0 for NO: ");
                        while (!int.TryParse(Console.ReadLine(), out confirmExit)
|| (confirmExit != 0 && confirmExit != 1))
                        {
                            Console.Write("Invalid input. Please enter 1 or 0:
");
                        }

                        if (confirmExit == 1)
                        {
                            Console.WriteLine("Thank you for using Advance
Calculator!");
                            useAgain = false;
                        }
                        else
                        {
                            useAgain = true;
                        }
                        break;

                    default:
                        Console.WriteLine("Please choose a valid option :)");
                        useAgain = true;
                        break;
                }
                Console.WriteLine();
            } while (useAgain);
        }
    }
}
```

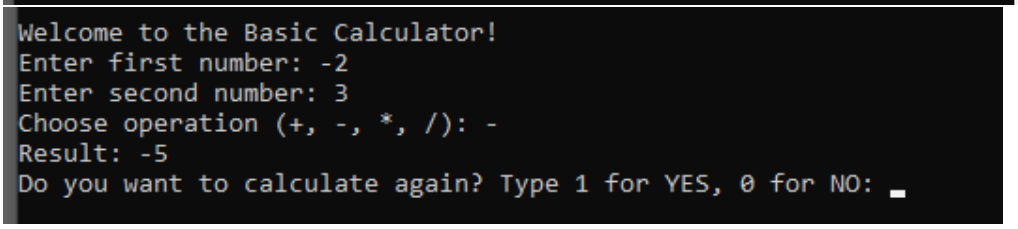
DOCUMENTATION

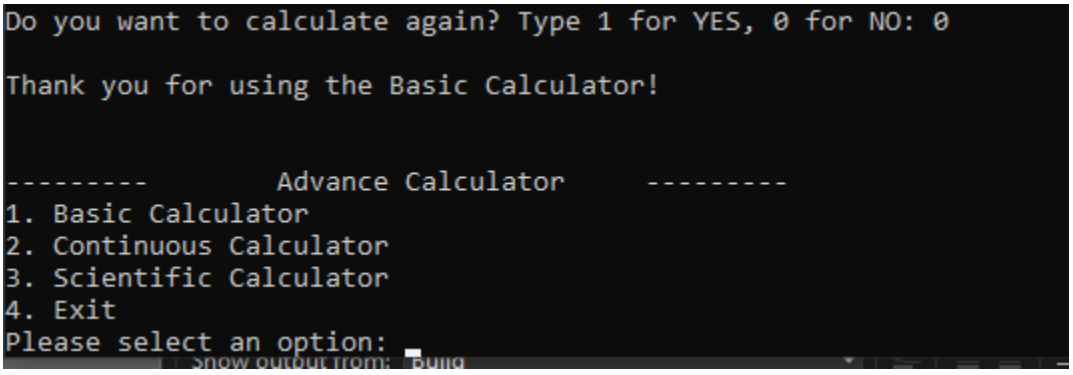
Landing Page



BASIC CALCULATOR OPERATIONS







Error handling

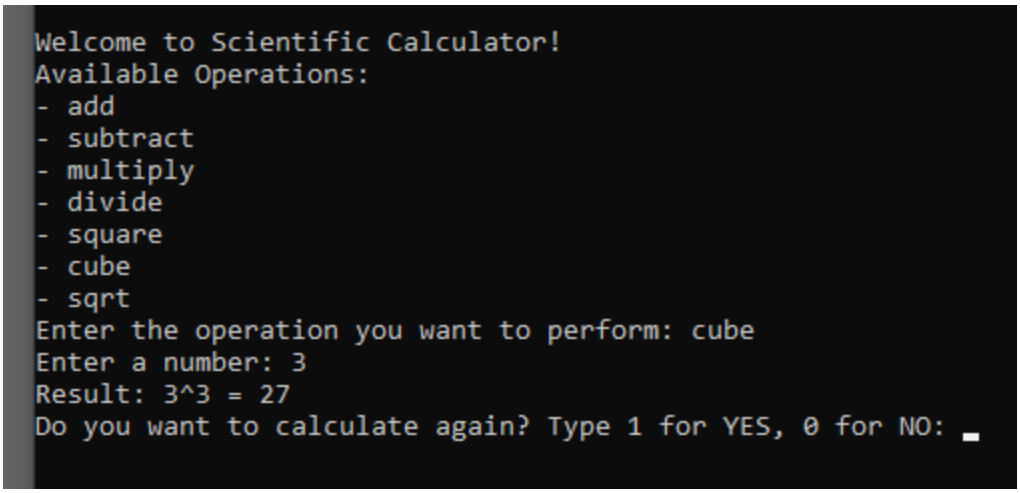
```
Welcome to the Basic Calculator!
Enter first number: s
Invalid input. Try again.
```

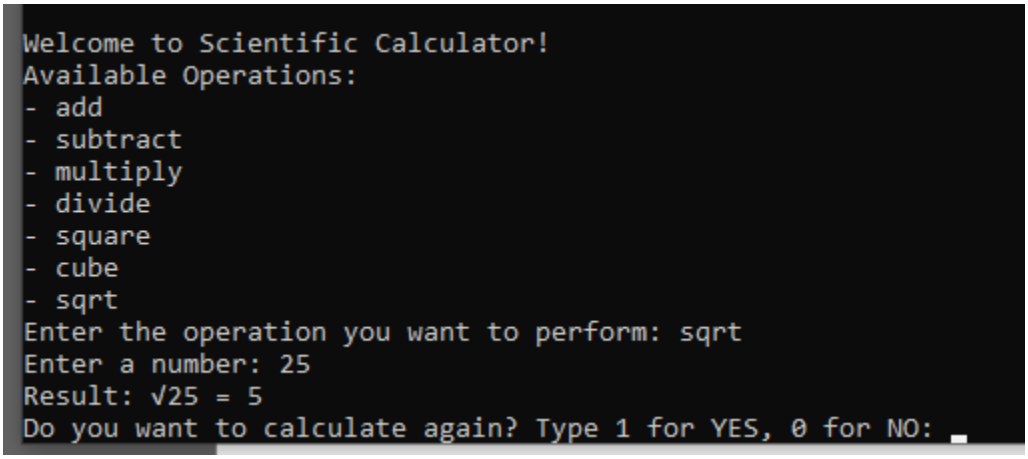
CONTINUOUS CALCULATOR OPERATIONS

```
Please select an option: 2
Welcome to the Continuous Calculator!
In this calculator you can continuously perform basic math operations and in
put number!
Enter first number: 1

Enter an operation (+, -, *, /) or type 'exit' to stop: +
Enter next number: 3
Result: 4
Current Result: 4

Enter an operation (+, -, *, /) or type 'exit' to stop: /
Enter next number: 2
Result: 2
Current Result: 2

Enter an operation (+, -, *, /) or type 'exit' to stop: *2
Enter next number: 4
? Invalid operation.

Enter an operation (+, -, *, /) or type 'exit' to stop: *
Enter next number: 2
Result: 4
Current Result: 4

Enter an operation (+, -, *, /) or type 'exit' to stop: exit
Exiting Continuous Calculator.
```

SCIENTIFIC CALCULATOR OPERATIONS

```
Welcome to Scientific Calculator!
Available Operations:
- add
- subtract
- multiply
- divide
- square
- cube
- sqrt
Enter the operation you want to perform: square
Enter a number: 2
Result: 2² = 4
Do you want to calculate again? Type 1 for YES, 0 for NO: 1
```





RELFECTION

The program Advance Calculator features all different kinds of loops, conditional statements, and the tryparse syntax that is not familiar for me. I used the do-while loops so users can use the calculators repeatedly when they are prompted. I used the while loops to help validate the inputs and let them try again to input if it is not a valid input. Finally, I used the foreach loop in the scientific calculator's operations. This ensures that if more operations are added in the future, they will automatically appear in the options list which improves maintainability.

The flow statements especially the switch syntax, helps me manage the selections of the 'menu' of the program as well as the operations of the calculators and makes the code cleaner and easier to extend. I used the if – else statements for further validation of output like checking if the divisor is 0.

The program features a main menu displayed in every outer loop iteration. It enables users to:

- Choose the calculator type,
- Perform calculations,
- Return to the menu or exit.

The clear separation of cases and the consistent prompt format give a user-friendly experience and make it easy to explore different calculators.

Instead of using exception handling, the program uses:

- double.TryParse() and int.TryParse()
  These methods safely attempt conversion and return a bool indicating success or failure.
- while (!valid) loops to repeat prompts until the input is correct.

Every time a division is about to occur:

```
if (num2 != 0)
   Console.WriteLine($"Result: {num1 / num2}");
else
   Console.WriteLine(" ✕ Cannot divide by zero.");
```

This check ensures the program never attempts an invalid division, preventing runtime errors or undefined results.

In the Continuous Calculator, the first input is stored in num1 and for every subsequent operation, the result replaces num1, allowing it to act as the first number for the next calculation. It also uses a count variable to limit operations to a maximum of 10. Users can exit early by typing "exit" during operation input, which is checked using:

The challenging and interesting parts for me is the overall creation of the program because it is quite overwhelming cause of its size and the logic behind it. I also find it hard to validate the output since I need to use the tryParse which I'm not familiar with.

The most challenging and interesting part for me was the overall creation of the program, especially because of its large size and the logic behind each part. It was sometimes overwhelming to organize all the nested loops and make sure each calculator worked as expected. I also found it difficult to validate the inputs at first since I had to use TryParse, which was new to me.

Despite the complexity and all the different requirements for each calculator mode, I found this activity to be fun and rewarding. It helped me understand control structures, loops, and user input handling in a deeper way. It also showed me the importance of writing clean, readable, and modular code, especially when working with menus and user interactions.