Task 1:

In this part, you will create a program that is a simple calculator. Your calculator will perform addition, subtraction, multiplication and division of two values. It will accept floating-point values, and it will check for division by zero.

In this software using for add, subtract, multiply, division of two numbers. It's only working for number.

Working Procedure:

This is a C# desktop base application. In this software have five buttons they are: Add, Subtract, Multiply, Division and Quit.

User need to input 2 numbers in this textbox. Then when I click add button then the software first checks the input value is number or not. If the input value is not number then it shows invalid input below the textbox. If the input numbers is valid then it adds two numbers and then it show in result textbox. Also like this after check the input validation it calculate subtract, multiply and division. Only Infinity error message show when any number divided by 0. All the calculation part I use try catch for errors handling.

If I click the quit button then it clear all the textbox number and all error message.

Algorithm:

In this program take a variable and get input of this variable. If I click Add button then it first checks the input. If input is ok then it add two numbers and give a result in result textbox. If I click subtract button then it first the input is ok or not. If the input is ok then it subtracts two number and shoe the result in result textbox. If I click multiply then it also check input. If the input is ok then it multiply two numbers and show the result in result textbox. If I click division button then it checks input is ok or not. If the input is ok then it divides two numbers and gets a result. If the result is infinity then it show an error if not then it show the result in result textbox. If I click quit then it clear all the textbox.

Exception Handling:

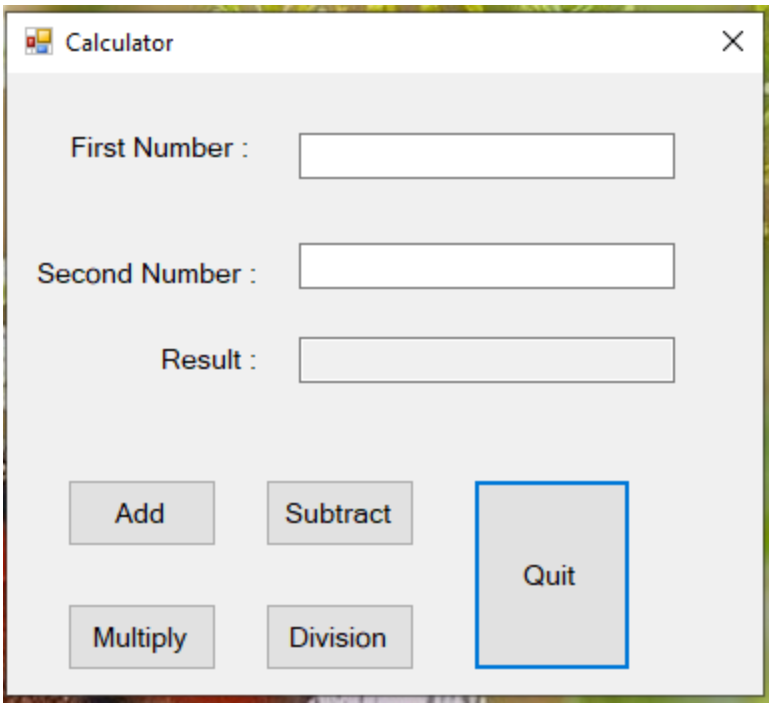
If I click without input any value in textbox then it show an error message. If the number is strings then it shows an error message. In this program I use try catch for errors handling.

Code:

```
// Addition action
private void button1_Click(object sender, EventArgs e)
{

    // check validation
    int er = 0;
    label5.Visible = false;
    label4.Visible = false;
    label6.Visible = false;
    string i = textBoxNum1.Text;
    string k = textBoxNum2.Text;
    int j = 0;
    if (!int.TryParse(i, out j))
    {
        label4.Text = "Invalide Input";
        label4.Visible = true;
        er++;

    }
    else if (!int.TryParse(k, out j))
    {
        label5.Text = "Invalide Input";
        label5.Visible = true;
        er++;

    }

    else if (textBoxNum1.Text == "")
    {
        er++;
        label4.Text = "Please insert number..";
        label4.Visible = true;
    }
    else if (textBoxNum2.Text=="")
    {
        er++;
        label5.Text = "Please Insert number..";
        label5.Visible = true;
    }
    if (er > 0)
    {
```

```
    }

    // convert data type string to double
    double num1 = Convert.ToDouble(textBoxNum1.Text);
    double num2 = Convert.ToDouble(textBoxNum2.Text);
    double result = 0;



    result = num1 + num2; // add two  numbers
    textBoxResult.Text = result.ToString();



}


// Action Subtract
private void button2_Click(object sender, EventArgs e)
{
    // Check validation
    int er = 0;
    label5.Visible = false;
    label4.Visible = false;
    label6.Visible = false;
    string i = textBoxNum1.Text;
    string k = textBoxNum2.Text;
    int j = 0;
    if (!int.TryParse(i, out j))
    {
        label4.Text = "Invalide Input";
        label4.Visible = true;
        er++;

    }
    else if (!int.TryParse(k, out j))
    {
        label5.Text = "Invalide Input";
        label5.Visible = true;
        er++;

    }
```

Output :


Calculator
First Number :
Second Number :
Result :
Add
Subtract
Multiply
Division
Quit