Building a Calculator in C#
Open Up a Windows form:

![New Project dialog box showing Visual Studio installed templates and My Templates sections.](image)
Name your form:
Add a text box for the display:
Create buttons:

- The number buttons should have their text be the actual numbers. We will use this text in the program.
- The function buttons names are not important.
Copy/Paste, make it look nice:

- Copy and paste the buttons to keep size and placement consistent.
- Alter the text values to correspond to the buttons.
- Adjust the size of the form and placement of the elements.
- Change the background color of the form.
The Finished Design:
The First Piece of Code:

```csharp
namespace Calculator
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void btn0_Click(object sender, EventArgs e)
        {
            tb1.Text = tb1.Text + btn0.Text;
        }
    }
}
```

I add the text from the button’s text field to the textbox using concatenation.

This allows the population of the text box by the button presses.
private void btn2_Click(object sender, EventArgs e)
{
    tbl.Text = tbl.Text + btn2.Text;
}

private void btn3_Click(object sender, EventArgs e)
{
    tbl.Text = tbl.Text + btn3.Text;
}

private void btn4_Click(object sender, EventArgs e)
{
    tbl.Text = tbl.Text + btn4.Text;
}

private void btn5_Click(object sender, EventArgs e)
{
    tbl.Text = tbl.Text + btn5.Text;
}

private void btn6_Click(object sender, EventArgs e)
{
    tbl.Text = tbl.Text + btn6.Text;
}

private void btn7_Click(object sender, EventArgs e)
{
    tbl.Text = tbl.Text + btn7.Text;
}

private void btn8_Click(object sender, EventArgs e)
{
    tbl.Text = tbl.Text + btn8.Text;
}

private void btn9_Click(object sender, EventArgs e)
{
Let’s Program the Clear Button:
This will clear the text box:

```csharp
private void btn_clear_Click(object sender, EventArgs e)
{
    tb1.Clear();
}
```
Declaring the first variable:

- We need a variable to hold the first value that the user inputs.
- Let’s use a double for the datatype.
- A double has two advantages:
  - It allows for big numbers.
  - It allows for a decimal point.
Initialize the variable to zero:

```csharp
private void btn_clear_Click(object sender, EventArgs e)
{
    tbl.Clear();
}

double total1 = 0;
```
Let’s program the Plus button:

- This is our first math function button
- This will not actually do the calculation
- Calculations will be saved for the equals button.
This code does several things:

- Allows for more than two numbers to be acted upon
- Converts the string to a number (Parse)
- Clears the text box for the next input

```csharp
    double total1 = 0;

    private void btn_add_Click(object sender, EventArgs e)
    {
        total1 = total1 + double.Parse(tb1.Text);
        tb1.Clear();
    }
```
Adding a number to itself is a common task in programming, so a shortcut is commonly used: `+=`
Create a 2\textsuperscript{nd} variable to hold the next input:

```csharp
double total1 = 0;
double total2 = 0;

private void btn_add_Click(object sender, EventArgs e)
{
    total1 += double.Parse(tb1.Text);
    tb1.Clear();
}
```
Let’s Program the Equal Button:

- This button will do the work. This will develop over several stages.
This adds the two numbers together
Again, parsing converts strings to numbers
It resets the first variable to zero for adding several numbers together
Programming the minus button:

- To include other functions such as subtract, multiply and divide, how will the calculator know that we have chosen a different function?
- We require conditional statements
- A boolean (bool) is a datatype that has only two states.....true/false
Let’s create some boolean variables with conditions we can test for, and initialize them to false:

```csharp
    total2 = total1 + double.Parse(tb1.Text);
    tb1.Text = total2.ToString();
    total1 = 0;

    bool plusButtonClicked = false;
    bool minusButtonClicked = false;
    bool multiplyButtonClicked = false;
    bool divideButtonClicked = false;

    private void btn_sub_Click(object sender, EventArgs e)
    {
    }
    }
```
Integrating the boolean into the minus button:

```c#
bool plusButtonClicked = false;
bool minusButtonClicked = false;
bool multiplyButtonClicked = false;
bool divideButtonClicked = false;

private void btn_sub_Click(object sender, EventArgs e)
{
    total1 = total1 + double.Parse(txtDisplay.Text);
    txtDisplay.Clear();

    plusButtonClicked = false;
    minusButtonClicked = true;
    multiplyButtonClicked = false;
    divideButtonClicked = false;
}
```
The Multiply and Divide Buttons:

```csharp
private void btn_mul_Click(object sender, EventArgs e)
{
    total1 += double.Parse(tb1.Text);
    tb1.Clear();

    plusButtonClicked = false;
    minusButtonClicked = false;
    multiplyButtonClicked = true;
    divideButtonClicked = false;
}

private void btn_div_Click(object sender, EventArgs e)
{
    total1 += double.Parse(tb1.Text);
    tb1.Clear();

    plusButtonClicked = false;
    minusButtonClicked = false;
    multiplyButtonClicked = false;
    divideButtonClicked = true;
}
```
Utilizing the boolean in the equals button:

- We need to test the conditional boolean statements in the equals button and then supply the code for the correct function.
- This requires if/else if statements so that we can test for all conditions.
private void btn_tot_Click(object sender, EventArgs e)
{
    if (plusButtonClicked == true)
    {
        total2 = total1 + double.Parse(tb1.Text);
        tb1.Text = total2.ToString();
        total1 = 0;
    }
    else if (minusButtonClicked == true)
    {
        total2 = total1 - double.Parse(tb1.Text);
        tb1.Text = total2.ToString();
        total1 = 0;
    }
    else if (multiplyButtonClicked == true)
    {
        total2 = total1 * double.Parse(tb1.Text);
        tb1.Text = total2.ToString();
        total1 = 0;
    }
    else if (divideButtonClicked == true)
    {
        total2 = total1 / double.Parse(tb1.Text);
        tb1.Text = total2.ToString();
        total1 = 0;
    }
}
The Finished Product: