

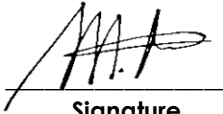
Faculty of Information Technology									
<p>I declare that I am familiar with, and will abide to the Examination rules of CTU</p>  <p>Signature</p>	<p>SUBJECT NAME: SOFTWARE DESIGN AND DEVELOPMENT SUBJECT CODE: PRG521</p>								
	<p>FORMATIVE ASSESSMENT Duration: Sep 15 – 04 Oct Date 02 October 2023 Total Marks: 20 Total pages: 8</p>					<p>Examiner: Miss Faith Muwishi Moderator: Mr. Newton</p>			
	<p>Student number</p>								
	2	0	2	3	2	7	5	9	
	<p>Surname: Poponi</p>				<p>Initials: M.P</p>				/

Table of Contents

Question 1.....	3
Full code.....	3
Output.....	5
Submission button with Exception handling.....	5
Clear button	6
Exit button.....	6
Exception messages (displayed in message box) saved in text file	6
Results message (displayed in textbox) saved in Tax Calculation file	8

Question 1

Full code

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Sales_Tax_Calculation_Form_20232759
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void numericUpDown1_ValueChanged(object sender, EventArgs e)
        {
        }

        private void btnSubmit_Click(object sender, EventArgs e)
        {
            try
            {
                // Get the selected sales tax rate from the numericUpDown
                int taxPercentage = (int)numUpDownTax.Value;

                // Parse the amount from the TextBox
                decimal amount = decimal.Parse(txtAmount.Text);

                // Calculate tax and total
                decimal taxAmount = (amount * taxPercentage) / 100;
                decimal totalAmount = amount + taxAmount;

                // Display a MessageBox with the results
                MessageBox.Show($"Tax on {amount:C} at {taxPercentage}% is {taxAmount:C}\nThe Total is: {totalAmount:C}", "Tax Calculator");
            }
            catch { }
        }
    }
}
```

```
// Save the results to the "TaxCalculation.txt" file
string resultMessage = $"Tax Calculation:
{DateTime.Now}\n\nTax on {amount:C} at {taxPercentage}% is
{taxAmount:C}\nThe Total is: {totalAmount:C}\n\n";

File.AppendAllText(@"C:\Users\popmz\source\repos\Sales_Tax_Calculation_Form
_20232759\TaxCalculation.txt", resultMessage);
}
catch (FormatException ex)
{
    // Save the exception message to the text file
    string exceptionMessage = $"{DateTime.Now}:
{ex.Message}\n";

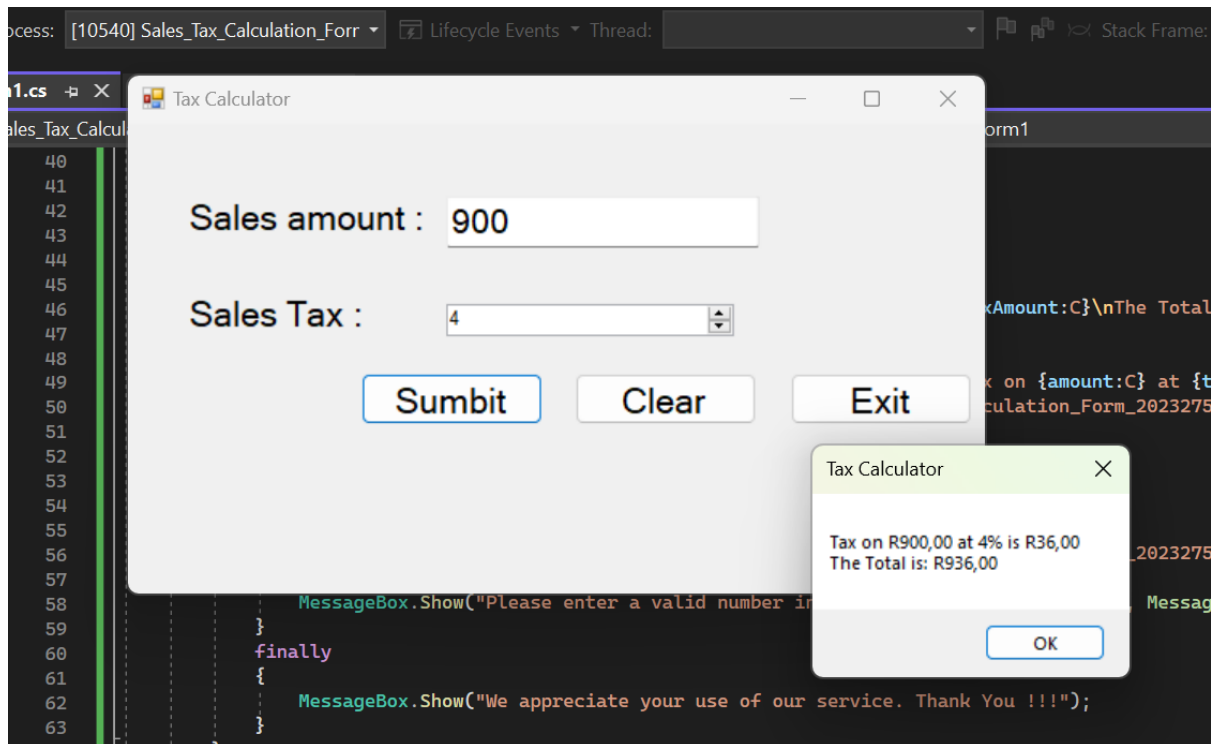
File.AppendAllText(@"C:\Users\popmz\source\repos\Sales_Tax_Calculation_Form
_20232759\Exceptionfile.txt", "Exceptions \n\n" + exceptionMessage);

    MessageBox.Show("Please enter a valid number in the Amount
field.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
}
finally
{
    MessageBox.Show("We appreciate your use of our service.
Thank You !!!");
}
}

private void btnClear_Click(object sender, EventArgs e)
{
    // Clear the Amount TextBox and the numeric up-down control
    txtAmount.Clear();
    numUpDownTax.Value = 0;
}

private void btnExit_Click(object sender, EventArgs e)
{
    // Close the application
    Application.Exit();
}
}
```

Output



Submission button with Exception handling

```
private void btnSubmit_Click(object sender, EventArgs e)
{
    try
    {
        // Get the selected sales tax rate from the numericUpDown control
        int taxPercentage = (int)numUpDownTax.Value;

        // Parse the amount from the TextBox
        decimal amount = decimal.Parse(txtAmount.Text);

        // Calculate tax and total
        decimal taxAmount = (amount * taxPercentage) / 100;
        decimal totalAmount = amount + taxAmount;

        // Display a MessageBox with the results

        MessageBox.Show($"Tax on {amount:C} at {taxPercentage}% is {taxAmount:C}\nThe Total is: {totalAmount:C}", "Tax Calculator");

        // Save the results to the "TaxCalculation.txt" file
        string resultMessage = $"Tax Calculation: {DateTime.Now}\n\nTax on {amount:C} at {taxPercentage}% is {taxAmount:C}\nThe Total is: {totalAmount:C}\n\n";
    }
    catch { }
    finally
    {
        MessageBox.Show("We appreciate your use of our service. Thank You !!!");
    }
}
```

```
File.AppendAllText(@"C:\Users\popmz\source\repos\Sales_Tax_Calculation_Form_20232759\TaxCalculation.txt",
resultMessage);
```

```
    }
    catch (FormatException ex)
    {
        // Save the exception message to the text file
        string exceptionMessage = $"{DateTime.Now}: {ex.Message}\n";
```

```
File.AppendAllText(@"C:\Users\popmz\source\repos\Sales_Tax_Calculation_Form_20232759\Exceptionfile.txt", "Ex
ceptions \n\n" + exceptionMessage);
```

```
        MessageBox.Show("Please enter a valid number in the Amount field.", "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
```

```
    }
    finally
    {
        MessageBox.Show("We appreciate your use of our service. Thank You !!!");
    }
}
```

Clear button

```
private void btnClear_Click(object sender, EventArgs e)
{
    // Clear the Amount TextBox and the numeric up-down control
    txtAmount.Clear();
    numUpDownTax.Value = 0;
}
```

Exit button

```
private void btnExit_Click(object sender, EventArgs e)
{
    // Close the application
    Application.Exit();
}
```

Exception messages (displayed in message box) saved in text file

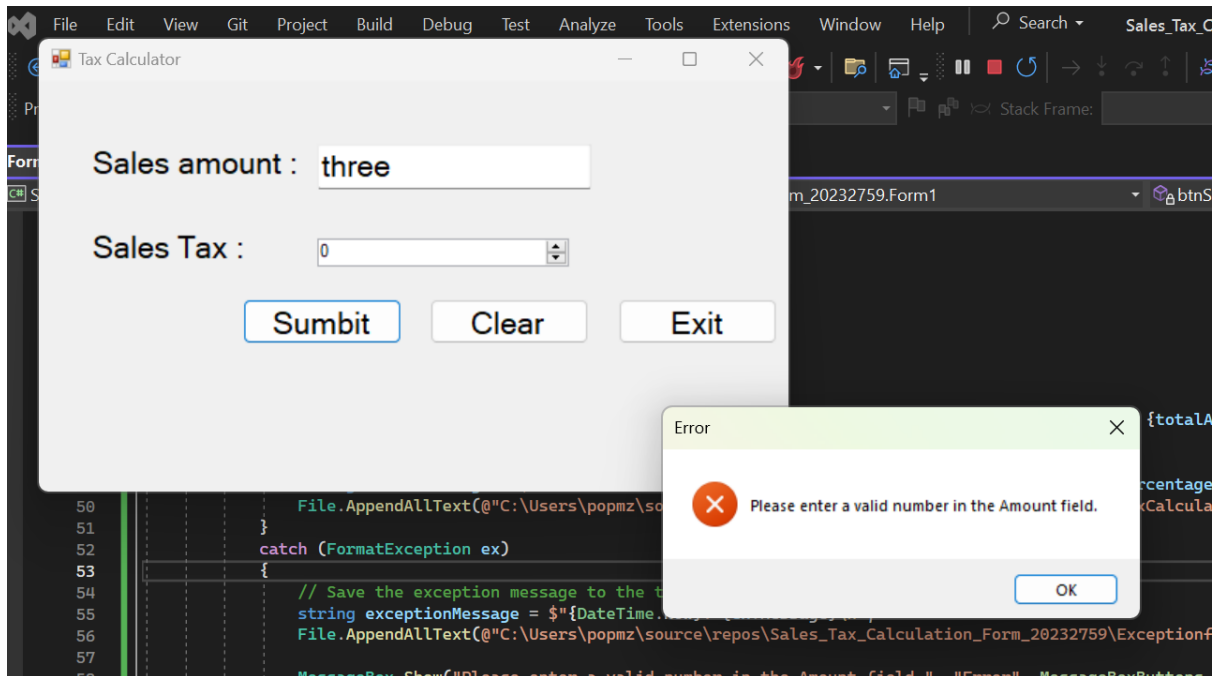
```
catch (FormatException ex)
{
    // Save the exception message to the text file
    string exceptionMessage = $"{DateTime.Now}: {ex.Message}\n";
```

```
File.AppendAllText(@"C:\Users\popmz\source\repos\Sales_Tax_Calculation_Form_20232759\Exceptionfile.txt", "Ex
ceptions \n\n" + exceptionMessage);
```

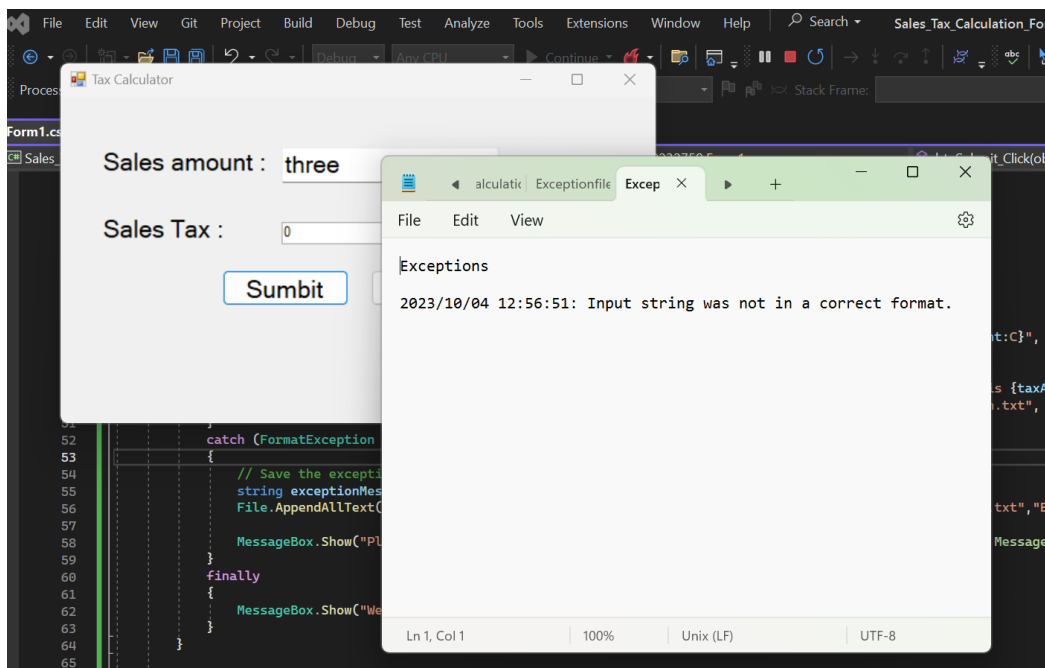
```
        MessageBox.Show("Please enter a valid number in the Amount field.", "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
```

```
    }
    finally
```

```
{  
    MessageBox.Show("We appreciate your use of our service. Thank You !!!");  
}
```



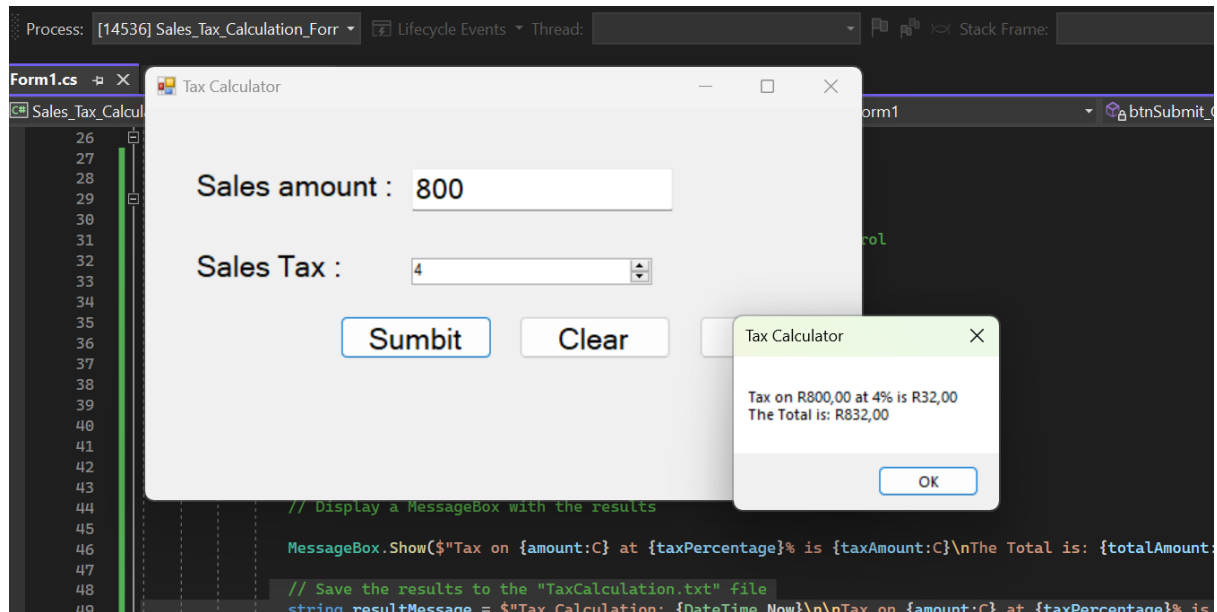
Text file (Exceptionfile.txt)



Results message (displayed in textbox) saved in Tax Calculation file

```
// Save the results to the "TaxCalculation.txt" file
string resultMessage = $"Tax Calculation: {DateTime.Now}\n\nTax on {amount:C} at {taxPercentage}% is {taxAmount:C}\n\nThe Total is: {totalAmount:C}\n\n";
```

```
File.AppendAllText(@"C:\Users\popmz\source\repos\Sales_Tax_Calculation_Form_20232759\TaxCalculation.txt",
resultMessage);
}
```



Text file (TaxCalculation file)

