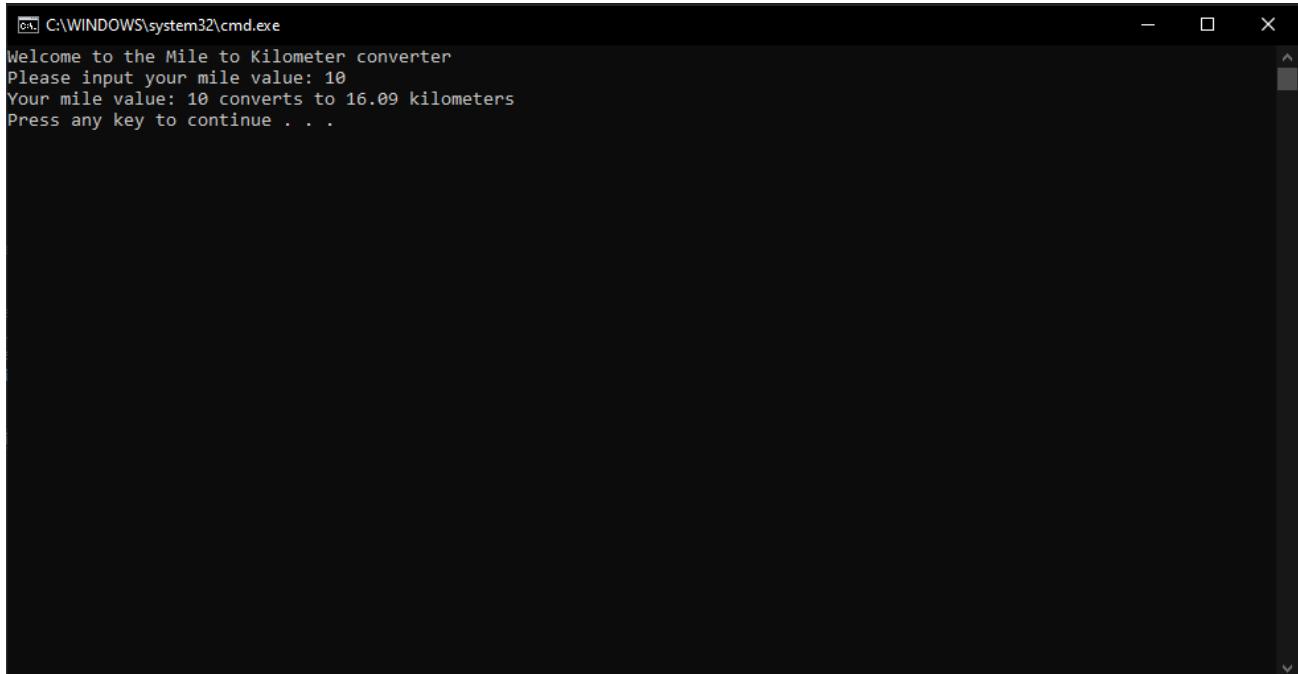
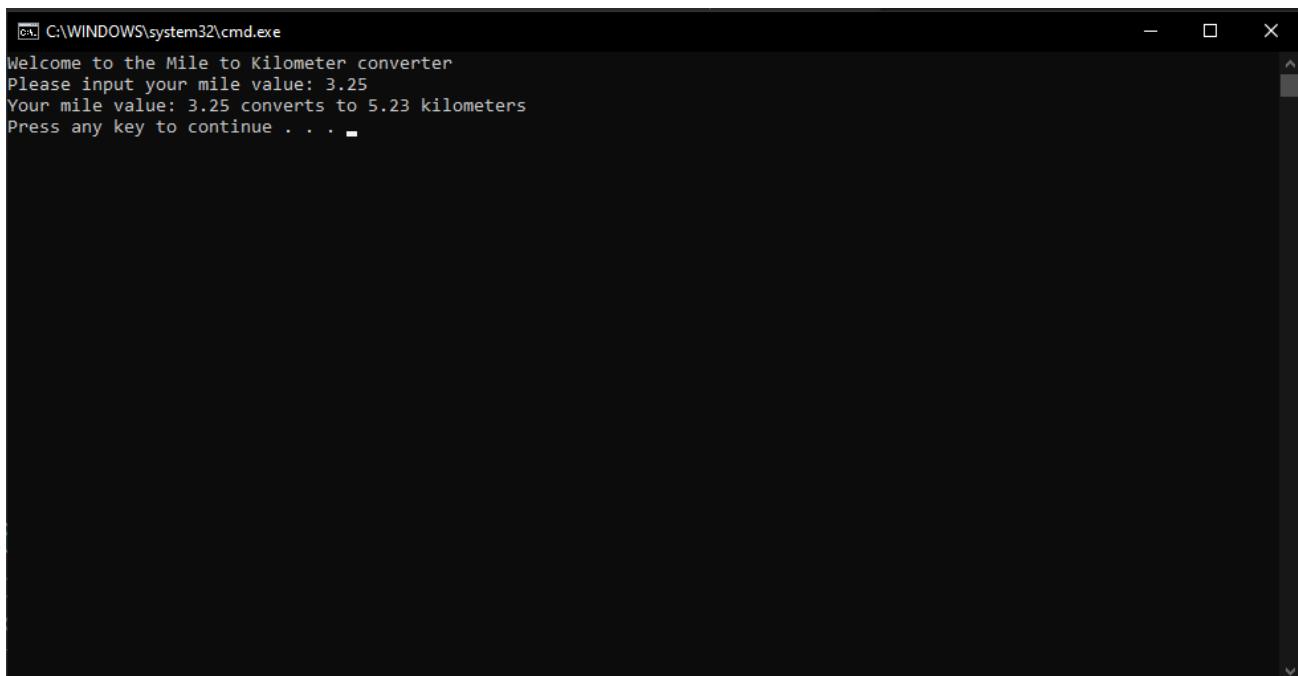


1. Write a C# Console application that converts a mile into its equivalent metric kilometer measurement. The program asks the user to input the value of miles to be converted and displays the original miles and the converted value . Test your code with inputs (1) 10 miles, (2) 3.25 miles.



```
C:\WINDOWS\system32\cmd.exe
Welcome to the Mile to Kilometer converter
Please input your mile value: 10
Your mile value: 10 converts to 16.09 kilometers
Press any key to continue . . .
```



```
C:\WINDOWS\system32\cmd.exe
Welcome to the Mile to Kilometer converter
Please input your mile value: 3.25
Your mile value: 3.25 converts to 5.23 kilometers
Press any key to continue . . .
```

```
class Program
{
    static void Main(string[] args)
    {
        double mile, km;
        Console.WriteLine("Welcome to the Mile to Kilometer converter");
        Console.Write("Please input your mile value: ");
        string mileS = Console.ReadLine();
        mile = Convert.ToDouble(mileS);
        km = mile * 1.60934;
        km=Math.Round(km, 2);
        Console.WriteLine("Your mile value: " + mile + " converts to " + km + " kilometers");
    }
}
```

2. Design a C# Windows Forms Application that solves Problem 1.

Mile to Kilometer Converter

Mile(s)  
10

Convert

Kilometer(s)  
16.09

Clear

Exit

This screenshot shows the initial state of the Windows Forms application. The title bar reads "Mile to Kilometer Converter". The main interface has two text input fields: one for "Mile(s)" containing "10" and another for "Kilometer(s)" containing "16.09". A large blue-bordered "Convert" button is positioned between them. On the right side, there are two buttons: "Clear" and "Exit".

Mile to Kilometer Converter

Mile(s)  
3.25

Convert

Kilometer(s)  
5.23

Clear

Exit

This screenshot shows the application after a conversion. The "Mile(s)" field now contains "3.25" and the "Kilometer(s)" field contains "5.23". The "Convert" button remains in its original position. The "Clear" and "Exit" buttons are still on the right side.

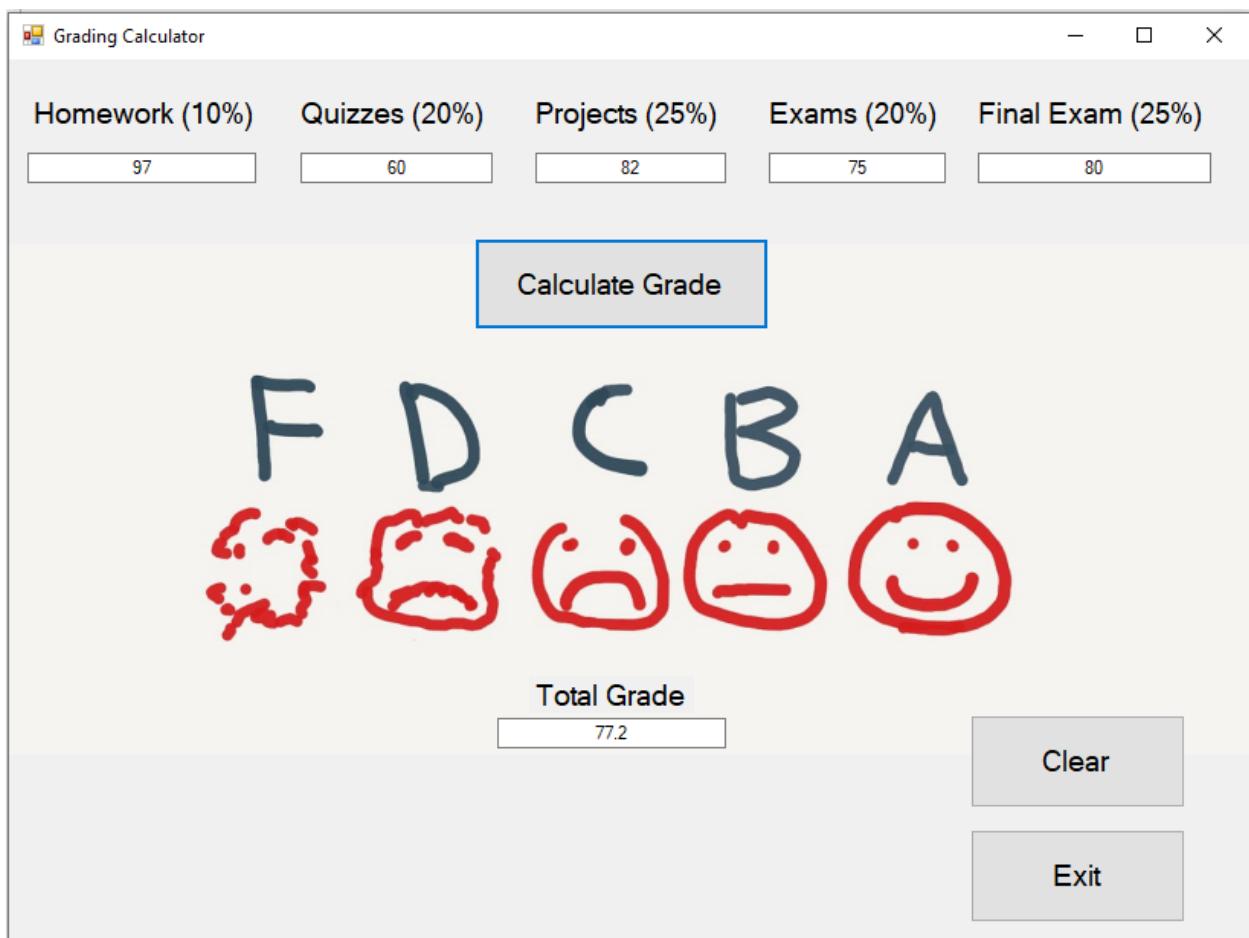
```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        double mile, km;
        string mileS = textBoxMile.Text;
        mile = Convert.ToDouble(mileS);
        km = mile * 1.60934;
        km = Math.Round(km, 2);
        string kilometer = Convert.ToString(km);
        textBoxKM.Text = kilometer;
    }

    private void buttonClear_Click(object sender, EventArgs e)
    {
        textBoxMile.Clear();
        textBoxKM.Clear();
    }

    private void buttonExit_Click(object sender, EventArgs e)
    {
        Application.Exit();
    }
}
```

Write a C# program (Your choice of Windows Forms application , or, Console Application) that computes a weighted total grade (with precision to 1 decimal digit).



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

```
}

private void button1_Click(object sender, EventArgs e)
{
    double hw, quiz, project, exam, finalexam, sum;
    string hwS = textBoxHW.Text;
    hw = Convert.ToDouble(hwS) * 0.10;
    string quizS = textBoxQuiz.Text;
    quiz = Convert.ToDouble(quizS) * 0.20;
    string projectS = textBoxProject.Text;
    project = Convert.ToDouble(projectS) * 0.25;
    string examS = textBoxExam.Text;
    exam = Convert.ToDouble(examS) * 0.20;
    string feS = textBoxFinalExam.Text;
    finalexam = Convert.ToDouble(feS) * 0.25;
    sum = hw + quiz + project + exam + finalexam;
    Math.Round(sum, 1);
    string sumfinal = Convert.ToString(sum);
    textBoxTotalGrade.Text = sumfinal;
}

private void button2_Click(object sender, EventArgs e)
{
    textBoxExam.Clear();
    textBoxFinalExam.Clear();
    textBoxHW.Clear();
    textBoxProject.Clear();
    textBoxQuiz.Clear();
    textBoxTotalGrade.Clear();
}

private void buttonExit_Click(object sender, EventArgs e)
{
    Application.Exit();
}

}
```