Student's name	Gundo Netsianda		
Student number	202301484		
Name of Campus	Braamfontein		
Year of Study	2025	**Semester**	1

Programme	Diploma in Information Technology		
Module Name	Programming 3A	**Module Code**	
Lecturer			
Due date	16 May 2025	**Date submitted**	13 May 2025

KEEP A COPY

Please note that it is your responsibility to retain copies of your assessments.

A CheckforPlagiarism report **MUST** be attached to each assignment submission.

DECLARATION BY STUDENT

I, the undersigned declare that:

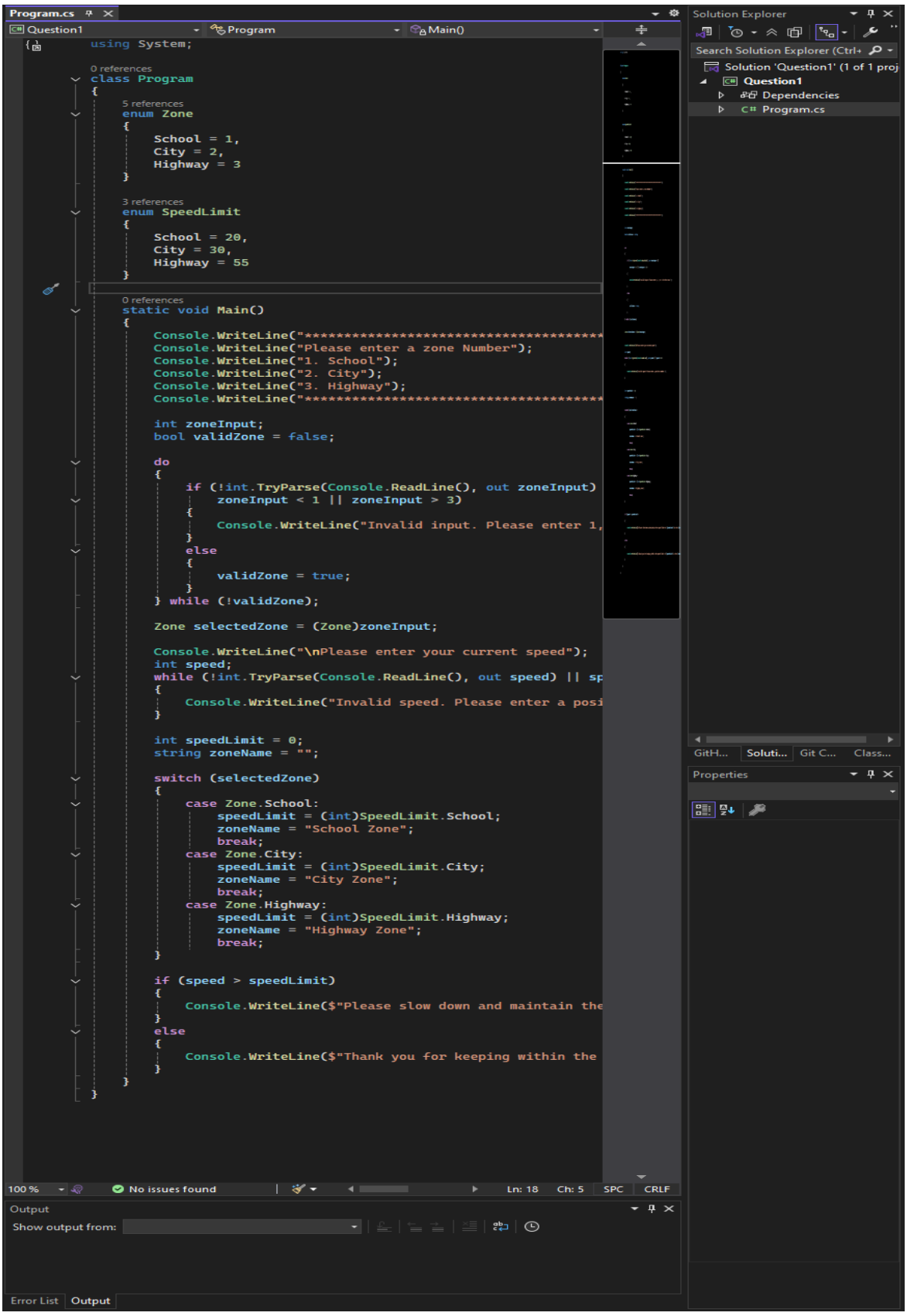
- I have retained a copy of this assessment.
- I understand what plagiarism is and are aware of the Damelin's policy in this regard.
- The work hereby submitted is my original work, gathered and utilised to fulfil the requirements of this assignment except for source material explicitly acknowledged.
- I have not used work previously pro duced by another student or any other persons to hand in as my own.
- I have not allowed, and will not allo w, anyone to copy my work with the intention of passing it off as their own work.

Signature of Student G.N **Date: 13 May 2025**

Table of Contents

Question 1

Code:



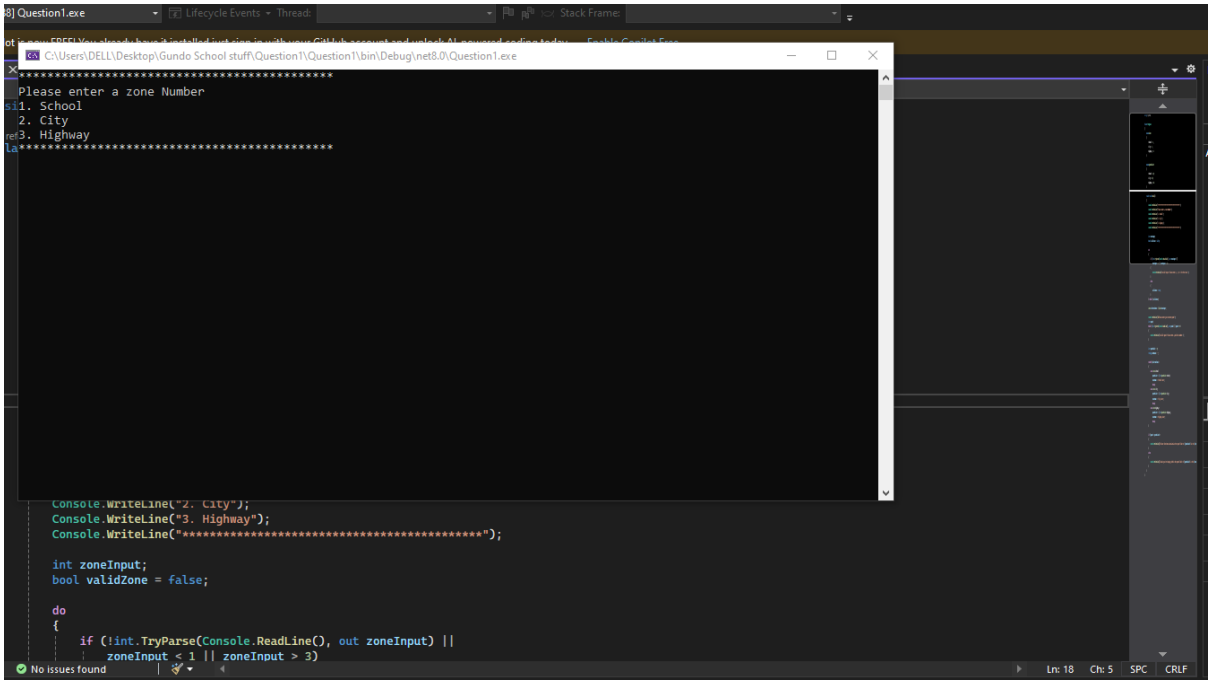
Source Code:

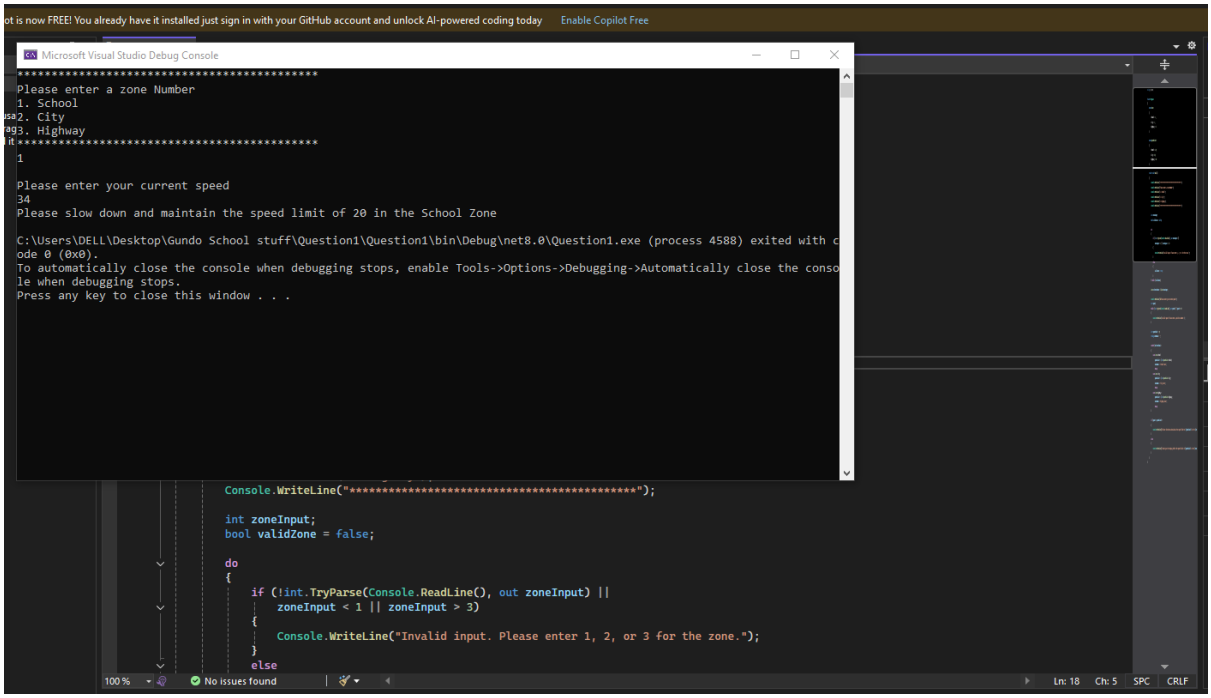
```
using System;

class Program
{
   enum Zone
   {
      School = 1,
      City = 2,
      Highway = 3
   }

   enum SpeedLimit
   {
      School = 20,
      City = 30,
      Highway = 55
   }

   static void Main()
   {
      Console.WriteLine("*****************************************");
      Console.WriteLine("Please enter a zone Number");
      Console.WriteLine("1. School");
      Console.WriteLine("2. City");
      Console.WriteLine("3. Highway");
      Console.WriteLine("*****************************************");

      int zoneInput;
      bool validZone = false;
```

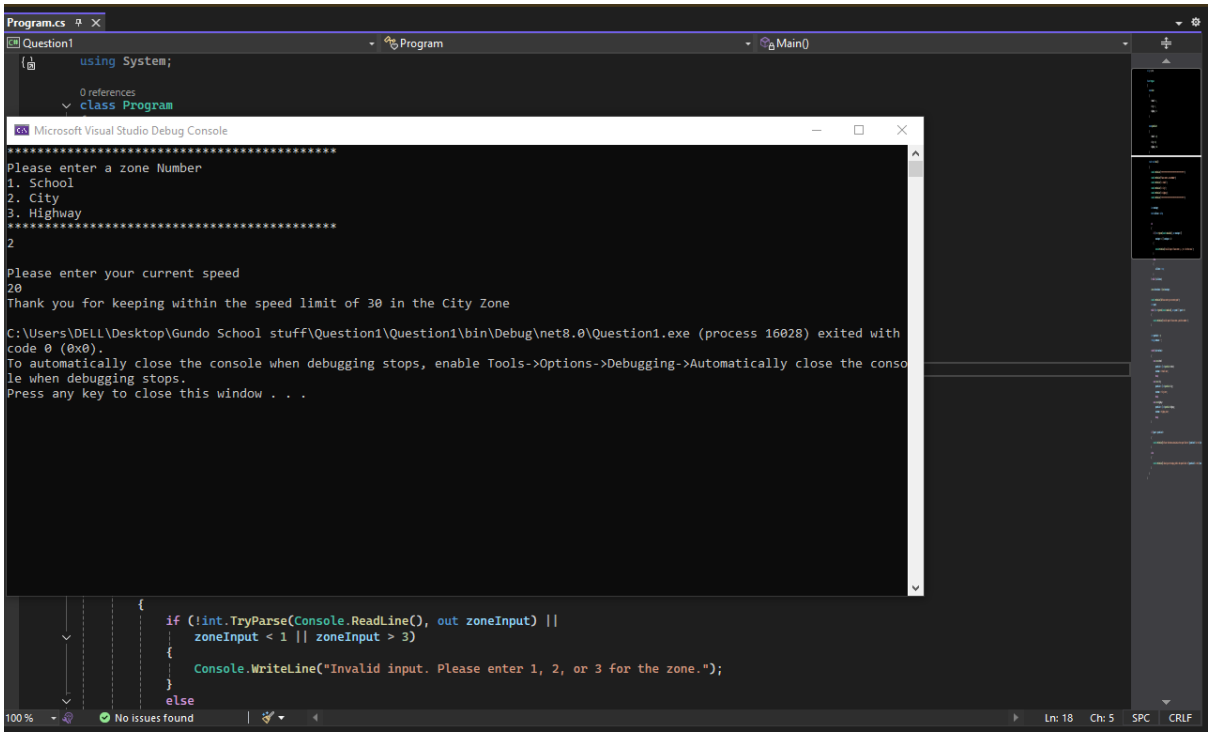
```
do
{
   if (!int.TryParse(Console.ReadLine(), out zoneInput) ||
      zoneInput < 1 || zoneInput > 3)
   {
      Console.WriteLine("Invalid input. Please enter 1, 2, or 3 for the zone.");
   }
   else
   {
      validZone = true;
   }
} while (!validZone);

Zone selectedZone = (Zone)zoneInput;

Console.WriteLine("\nPlease enter your current speed");
int speed;
while (!int.TryParse(Console.ReadLine(), out speed) || speed < 0)
{
   Console.WriteLine("Invalid speed. Please enter a positive number.");
}

int speedLimit = 0;
string zoneName = "";

switch (selectedZone)
{
   case Zone.School:
      speedLimit = (int)SpeedLimit.School;
      zoneName = "School Zone";
      break;
```

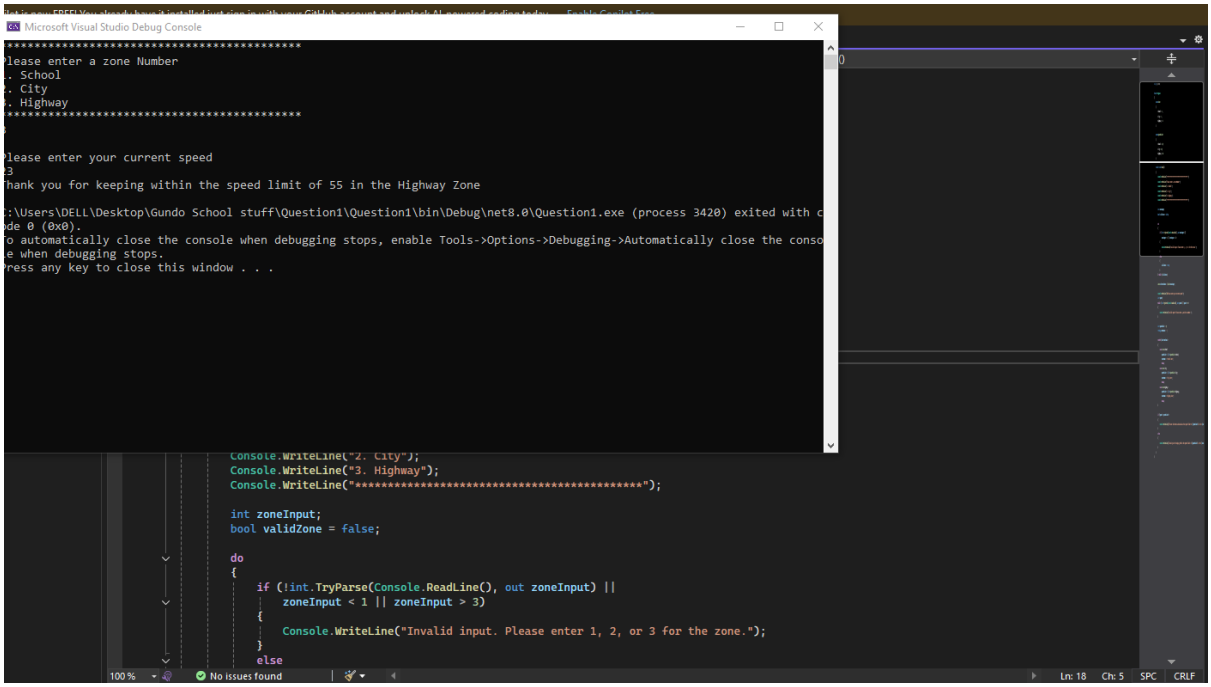
```
            case Zone.City:
                speedLimit = (int)SpeedLimit.City;
                zoneName = "City Zone";
                break;
            case Zone.Highway:
                speedLimit = (int)SpeedLimit.Highway;
                zoneName = "Highway Zone";
                break;
        }

        if (speed > speedLimit)
        {
            Console.WriteLine($"Please slow down and maintain the speed limit of {speedLimit} in the {zoneName}");
        }
        else
        {
            Console.WriteLine($"Thank you for keeping within the speed limit of {speedLimit} in the {zoneName}");
        }
    }
}
```

Output:


C:\Users\DELL\Desktop\Gundo School stuff\Question1\Question1\bin\Debug\net8.0\Question1.exe
****
Please enter a zone Number
1. School
2. City
3. Highway
****
Console.WriteLine("2. City");
Console.WriteLine("3. Highway");
Console.WriteLine("**********");
int zoneInput;
bool validZone = false;
do
{
if (!int.TryParse(Console.ReadLine(), out zoneInput) ||
zoneInput < 1 || zoneInput > 3)
No issues found
Ln: 18
Ch: 5
SPC
CRLF

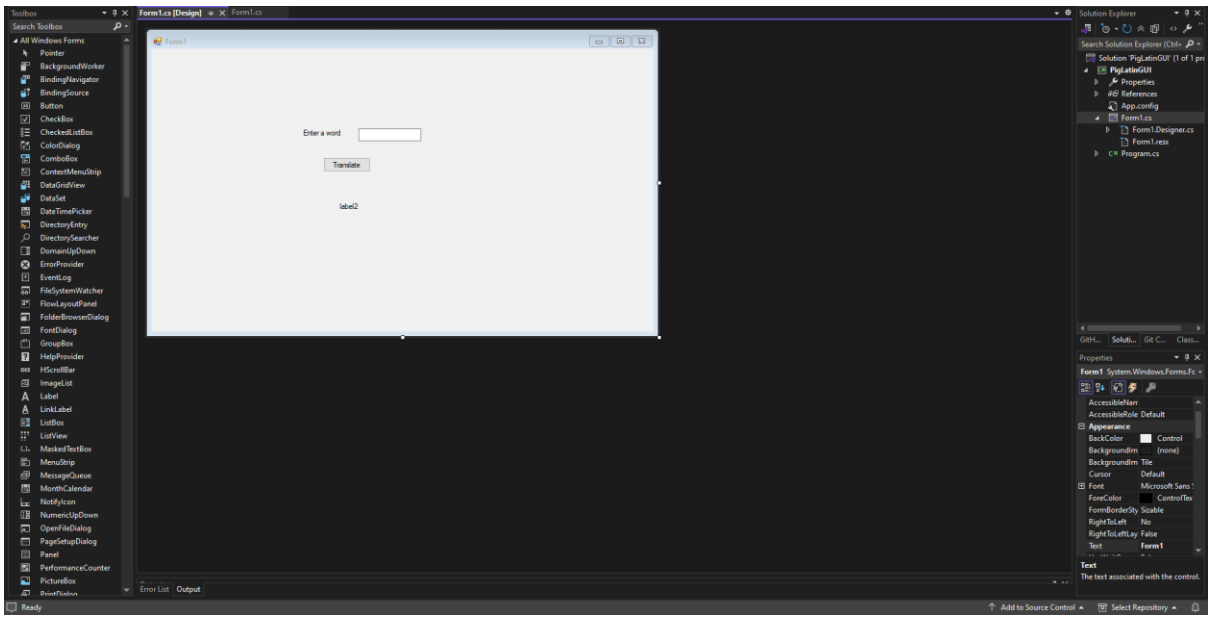

ot is now FREE! You already have it installed just sign in with your GitHub account and unlock AI-powered coding today
Enable Copilot Free
Microsoft Visual Studio Debug Console
****
Please enter a zone Number
1. School
2. City
3. Highway
****
1
Please enter your current speed
34
Please slow down and maintain the speed limit of 20 in the School Zone
C:\Users\DELL\Desktop\Gundo School stuff\Question1\Question1\bin\Debug\net8.0\Question1.exe (process 4588) exited with c
ode 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
Console.WriteLine("**********");
int zoneInput;
bool validZone = false;
do
{
if (!int.TryParse(Console.ReadLine(), out zoneInput) ||
zoneInput < 1 || zoneInput > 3)
{
Console.WriteLine("Invalid input. Please enter 1, 2, or 3 for the zone.");
}
else
100 %
No issues found
Ln: 18
Ch: 5
SPC
CRLF

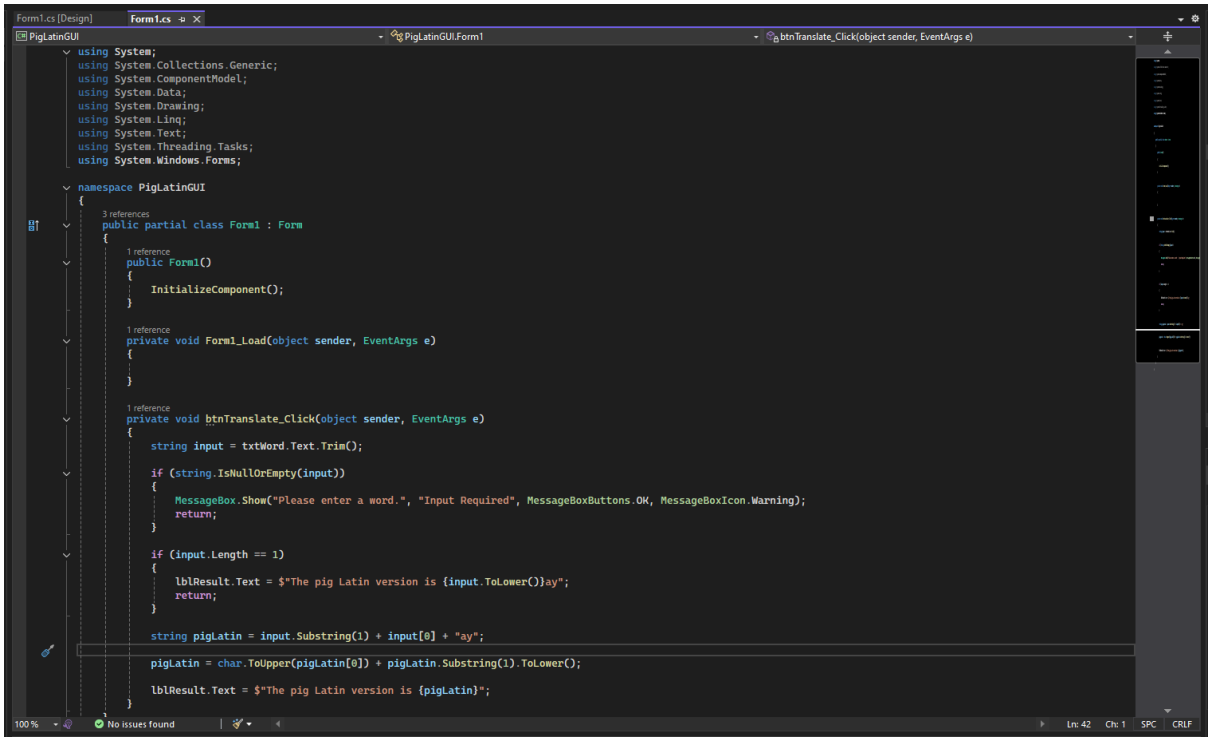

Program.cs
Question1
Program
Main()
using System;
0 references
class Program
Microsoft Visual Studio Debug Console
****
Please enter a zone Number
1. School
2. City
3. Highway
****
2
Please enter your current speed
20
Thank you for keeping within the speed limit of 30 in the City Zone
C:\Users\DELL\Desktop\Gundo School stuff\Question1\Question1\bin\Debug\net8.0\Question1.exe (process 16028) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
{
if (!int.TryParse(Console.ReadLine(), out zoneInput) ||
zoneInput < 1 || zoneInput > 3)
{
Console.WriteLine("Invalid input. Please enter 1, 2, or 3 for the zone.");
}
else
100 %
No issues found
Ln: 18
Ch: 5
SPC
CRLF


Microsoft Visual Studio Debug Console
Console.WriteLine("2. City");
Console.WriteLine("3. Highway");
Console.WriteLine("**********");
int zoneInput;
bool validZone = false;
do
{
if (!int.TryParse(Console.ReadLine(), out zoneInput) ||
zoneInput < 1 || zoneInput > 3)
{
Console.WriteLine("Invalid input. Please enter 1, 2, or 3 for the zone.");
}
else
100 %
No issues found
Ln: 18
Ch: 5
SPC
CRLF

Question 2

code





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

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

        private void Form1_Load(object sender, EventArgs e)
        {

        }

        private void btnTranslate_Click(object sender, EventArgs e)
        {
            string input = txtWord.Text.Trim();

            if (string.IsNullOrEmpty(input))
            {
                MessageBox.Show("Please enter a word.", "Input Required", MessageBoxButtons.OK, MessageBoxIcon.Warning);
                return;
            }

            if (input.Length == 1)
            {
                lblResult.Text = $"The pig Latin version is {input.ToLower()}ay";
                return;
            }

            string pigLatin = input.Substring(1) + input[0] + "ay";

            pigLatin = char.ToUpper(pigLatin[0]) + pigLatin.Substring(1).ToLower();

            lblResult.Text = $"The pig Latin version is {pigLatin}";
        }
```

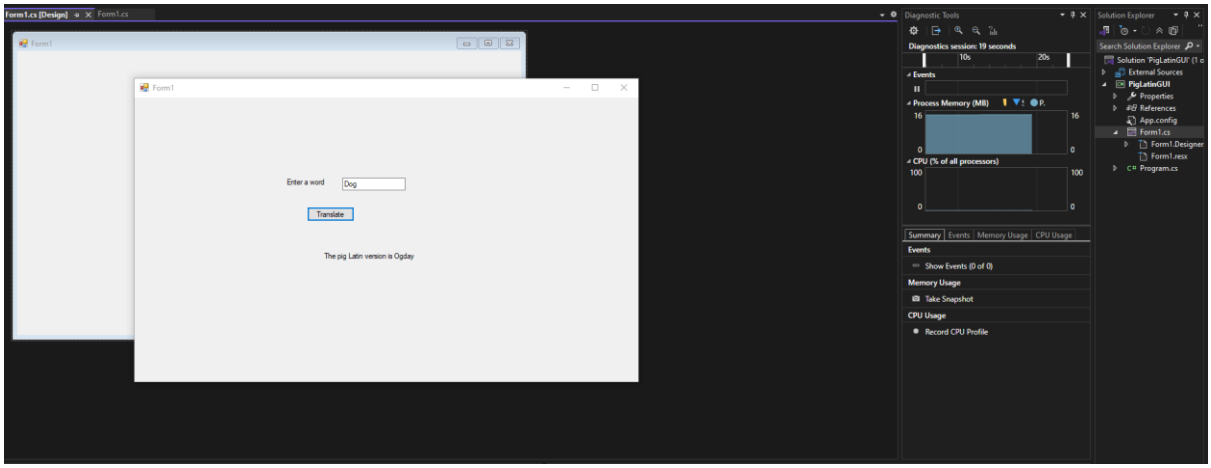
Source Code:

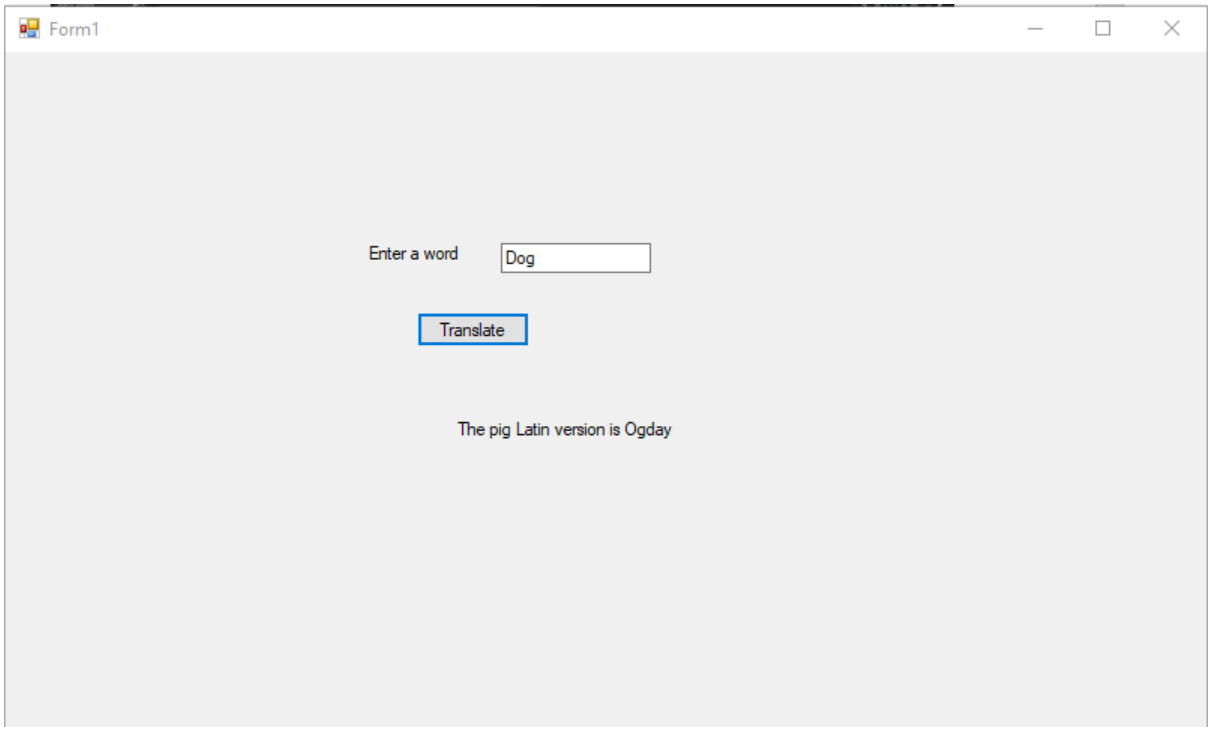
```
using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;
```

```
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

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

        private void Form1_Load(object sender, EventArgs e)
        {

        }

        private void btnTranslate_Click(object sender, EventArgs e)
        {
            string input = txtWord.Text.Trim();

            if (string.IsNullOrEmpty(input))
            {
                MessageBox.Show("Please enter a word.", "Input Required", MessageBoxButtons.OK, MessageBoxIcon.Warning);
                return;
            }
```

```
        if (input.Length == 1)
        {
            lblResult.Text = $"The pig Latin version is {input.ToLower()}ay";
            return;
        }
        string pigLatin = input.Substring(1) + input[0] + "ay";

        pigLatin = char.ToUpper(pigLatin[0]) + pigLatin.Substring(1).ToLower();

        lblResult.Text = $"The pig Latin version is {pigLatin}";
    }
  }
}
```

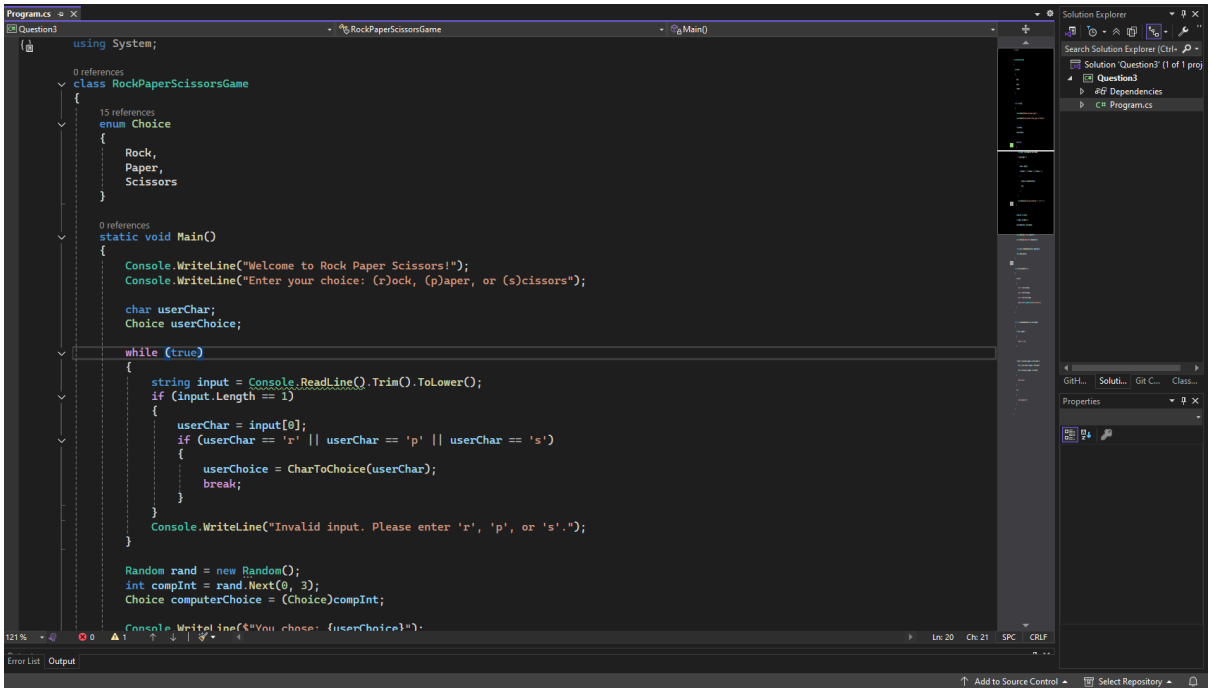
Output:



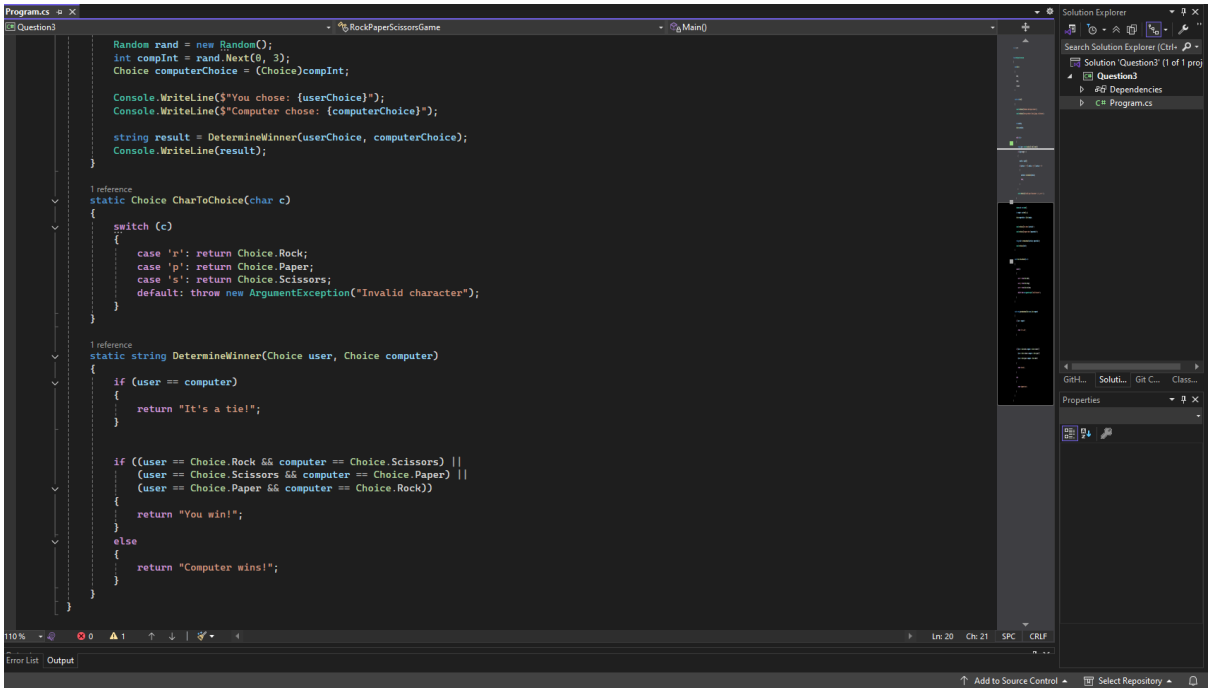

Form1
Enter a word
Dog
Translate
The pig Latin version is Ogday

Question 3

Code:



```
using System;

class RockPaperScissorsGame
{
    enum Choice
    {
        Rock,
        Paper,
        Scissors
    }

    static void Main()
    {
        Console.WriteLine("Welcome to Rock Paper Scissors!");
        Console.WriteLine("Enter your choice: (r)ock, (p)aper, or (s)cissors");

        char userChar;
        Choice userChoice;

        while (true)
        {
            string input = Console.ReadLine().Trim().ToLower();
            if (input.Length == 1)
            {
                userChar = input[0];
                if (userChar == 'r' || userChar == 'p' || userChar == 's')
                {
                    userChoice = CharToChoice(userChar);
                    break;
                }
            }
            Console.WriteLine("Invalid input. Please enter 'r', 'p', or 's'.");
        }

        Random rand = new Random();
        int compInt = rand.Next(0, 3);
        Choice computerChoice = (Choice)compInt;
```



```
        Random rand = new Random();
        int compInt = rand.Next(0, 3);
        Choice computerChoice = (Choice)compInt;

        Console.WriteLine($"You chose: {userChoice}");
        Console.WriteLine($"Computer chose: {computerChoice}");

        string result = DetermineWinner(userChoice, computerChoice);
        Console.WriteLine(result);
    }

    static Choice CharToChoice(char c)
    {
        switch (c)
        {
            case 'r': return Choice.Rock;
            case 'p': return Choice.Paper;
            case 's': return Choice.Scissors;
            default: throw new ArgumentException("Invalid character");
        }
    }

    static string DetermineWinner(Choice user, Choice computer)
    {
        if (user == computer)
        {
            return "It's a tie!";
        }

        if ((user == Choice.Rock && computer == Choice.Scissors) ||
            (user == Choice.Scissors && computer == Choice.Paper) ||
            (user == Choice.Paper && computer == Choice.Rock))
        {
            return "You win!";
        }
        else
        {
            return "Computer wins!";
        }
    }
}
```

Source Code:

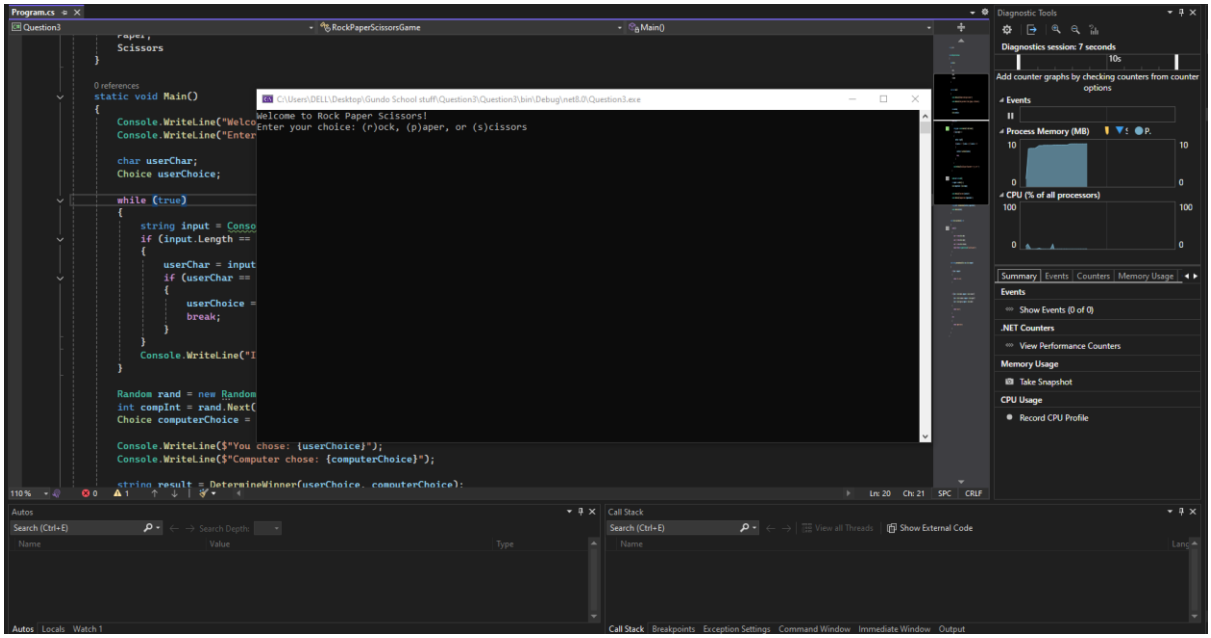
```
using System;


class RockPaperScissorsGame
```

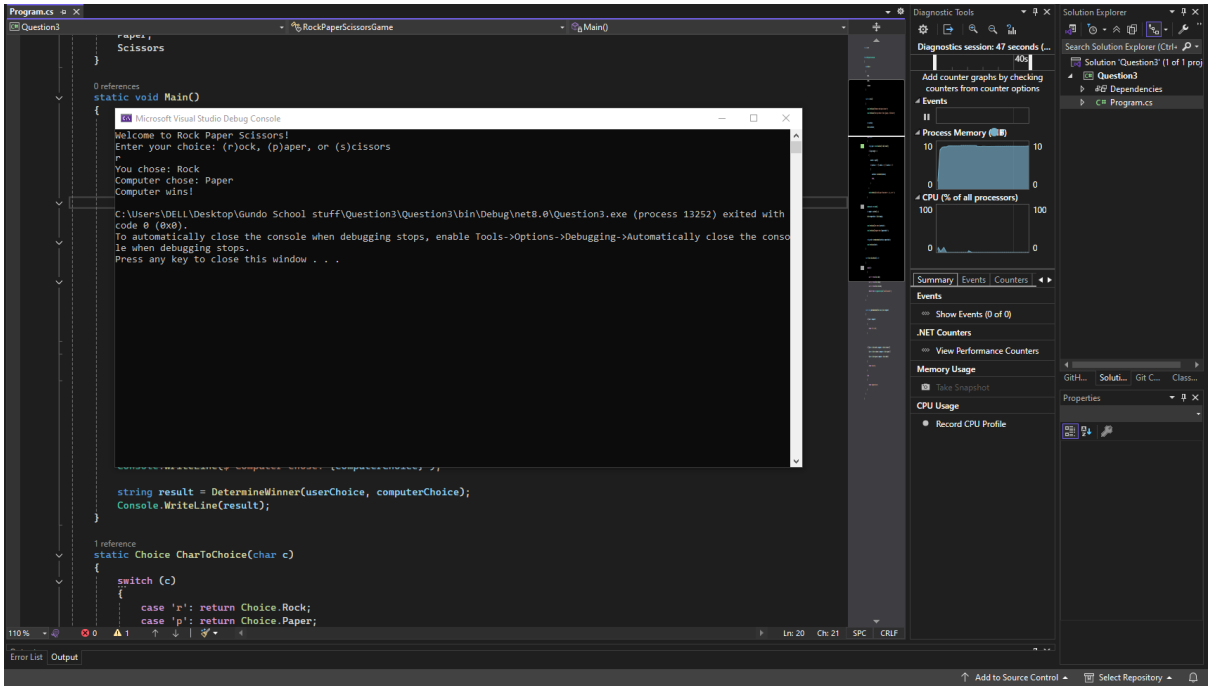
```
{
    enum Choice
    {
        Rock,
        Paper,
        Scissors
    }

    static void Main()
    {
        Console.WriteLine("Welcome to Rock Paper Scissors!");
        Console.WriteLine("Enter your choice: (r)ock, (p)aper, or (s)cissors");

        char userChar;
        Choice userChoice;

        while (true)
        {
            string input = Console.ReadLine().Trim().ToLower();
            if (input.Length == 1)
            {
                userChar = input[0];
                if (userChar == 'r' || userChar == 'p' || userChar == 's')
                {
                    userChoice = CharToChoice(userChar);
                    break;
                }
            }
            Console.WriteLine("Invalid input. Please enter 'r', 'p', or 's'.");
        }
```

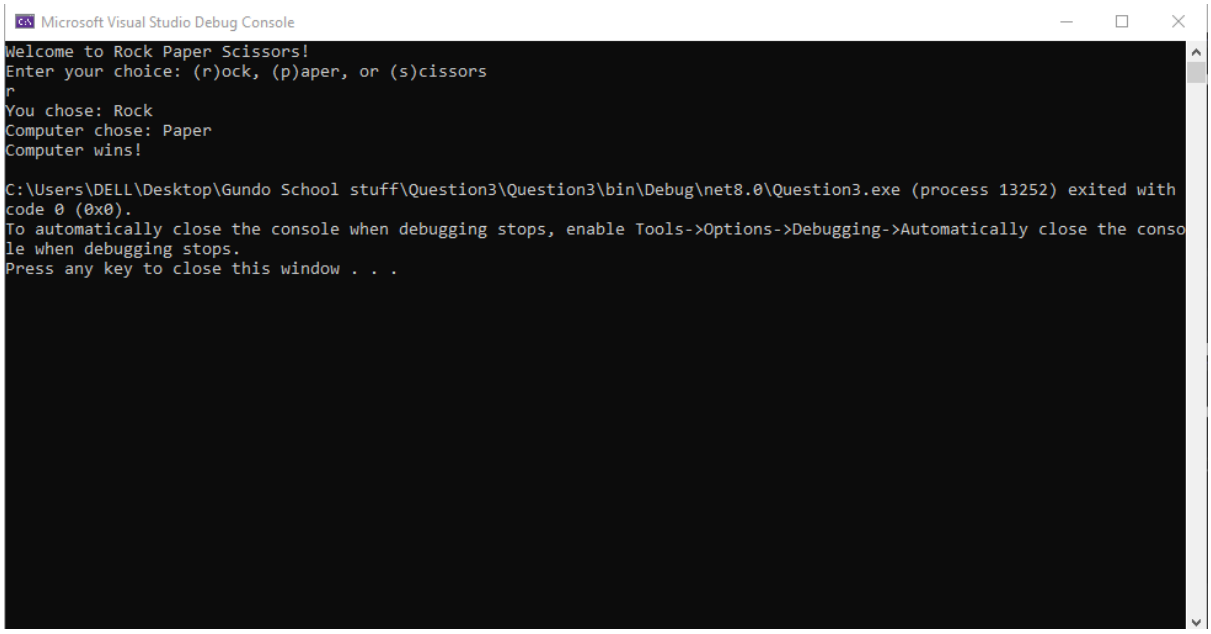
```
    Random rand = new Random();
    int compInt = rand.Next(0, 3);
    Choice computerChoice = (Choice)compInt;

    Console.WriteLine($"You chose: {userChoice}");
    Console.WriteLine($"Computer chose: {computerChoice}");

    string result = DetermineWinner(userChoice, computerChoice);
    Console.WriteLine(result);
}

static Choice CharToChoice(char c)
{
    switch (c)
    {
        case 'r': return Choice.Rock;
        case 'p': return Choice.Paper;
        case 's': return Choice.Scissors;
        default: throw new ArgumentException("Invalid character");
    }
}

static string DetermineWinner(Choice user, Choice computer)
{
    if (user == computer)
    {
        return "It's a tie!";
    }


    if ((user == Choice.Rock && computer == Choice.Scissors) ||
```

```
            (user == Choice.Scissors && computer == Choice.Paper) ||
            (user == Choice.Paper && computer == Choice.Rock))
        {
            return "You win!";
        }
        else
        {
            return "Computer wins!";
        }
    }
}
```

Output:

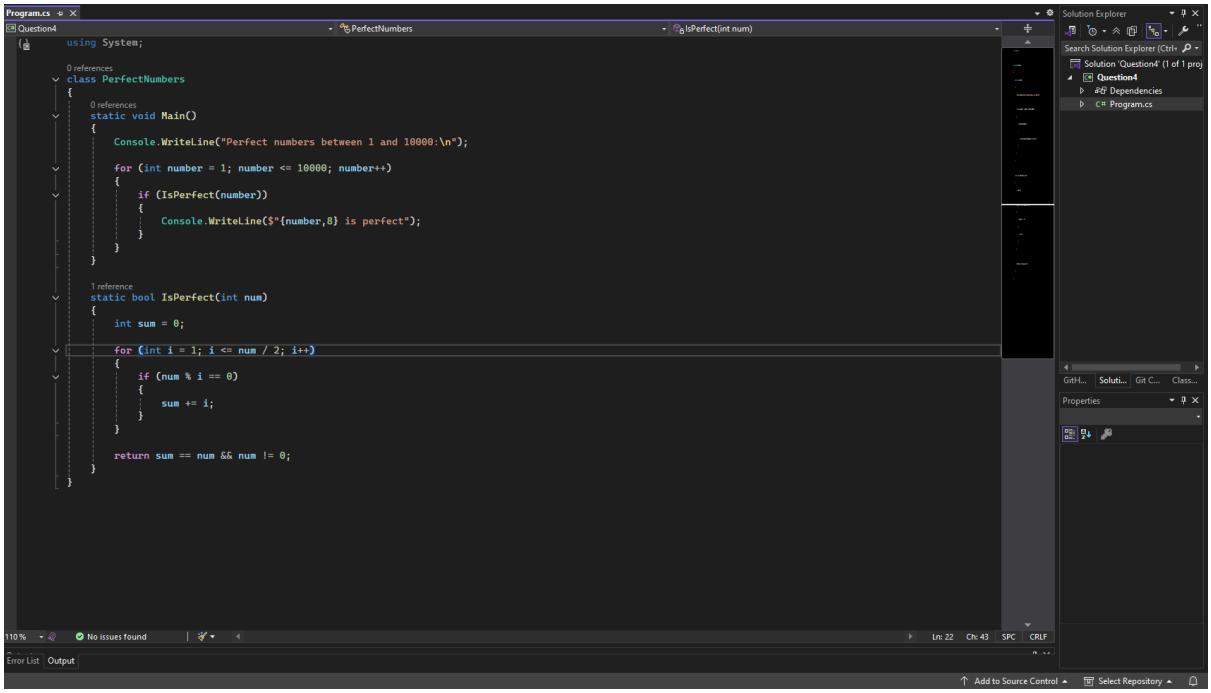



Program.cs
Question3
RockPaperScissorsGame
Main()
Scissors
}
0 references
static void Main()
{
Microsoft Visual Studio Debug Console
Welcome to Rock Paper Scissors!
Enter your choice: (r)ock, (p)aper, or (s)cissors
r
You chose: Rock
Computer chose: Paper
Computer wins!
C:\Users\DELL\Desktop\Gundo School stuff\Question3\Question3\bin\Debug\net8.0\Question3.exe (process 13252) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
string result = DetermineWinner(userChoice, computerChoice);
Console.WriteLine(result);
}
1 reference
static Choice CharToChoice(char c)
{
switch (c)
{
case 'r': return Choice.Rock;
case 'p': return Choice.Paper;
Diagnostic Tools
Diagnostics session: 47 seconds (...
Add counter graphs by checking counters from counter options
Events
Process Memory
CPU (% of all processors)
Summary
Events
Counters
Show Events (0 of 0)
.NET Counters
View Performance Counters
Memory Usage
Take Snapshot
CPU Usage
Record CPU Profile
Solution Explorer
Search Solution Explorer (Ctrl+
Solution 'Question3' (1 of 1 proj
Question3
Dependencies
Program.cs
Properties
Ln: 20
Ch: 21
SPC
CRLF
Error List
Output
Add to Source Control
Select Repository

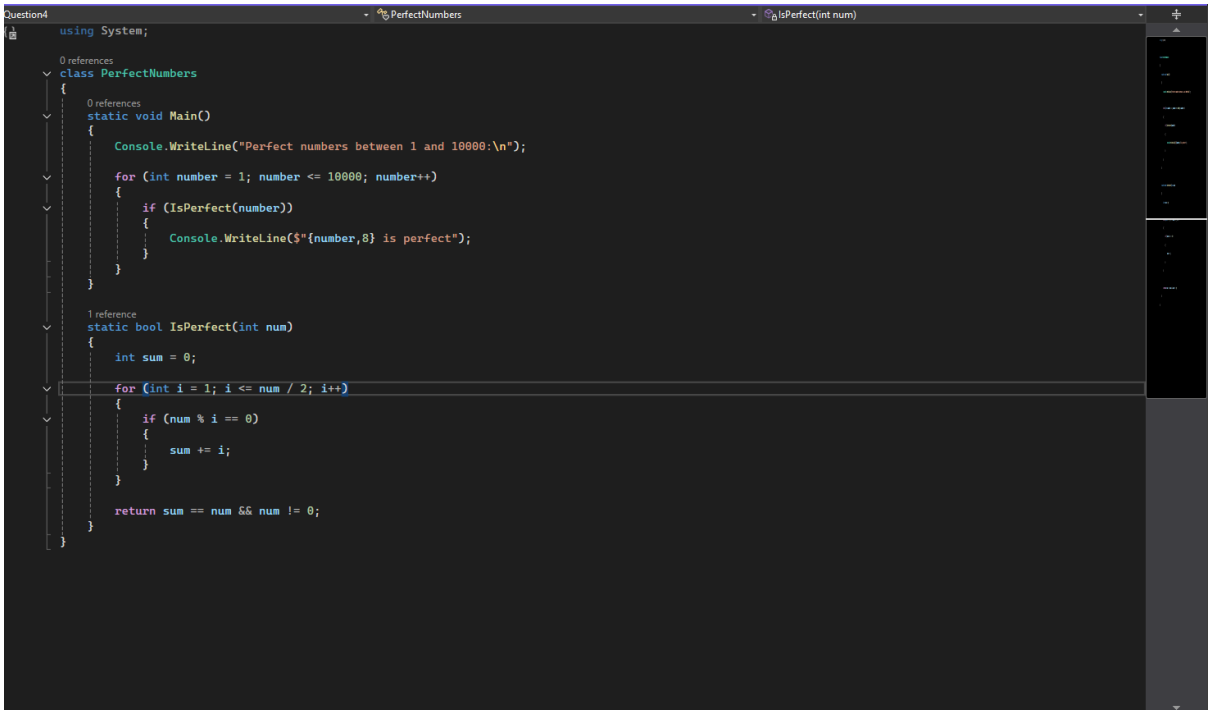

Microsoft Visual Studio Debug Console
Welcome to Rock Paper Scissors!
Enter your choice: (r)ock, (p)aper, or (s)cissors
r
You chose: Rock
Computer chose: Paper
Computer wins!
C:\Users\DELL\Desktop\Gundo School stuff\Question3\Question3\bin\Debug\net8.0\Question3.exe (process 13252) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .

Question 4

Code:



```
using System;

0 references
class PerfectNumbers
{
    0 references
    static void Main()
    {
        Console.WriteLine("Perfect numbers between 1 and 10000:\n");

        for (int number = 1; number <= 10000; number++)
        {
            if (IsPerfect(number))
            {
                Console.WriteLine($"{number,8} is perfect");
            }
        }
    }

    1 reference
    static bool IsPerfect(int num)
    {
        int sum = 0;

        for (int i = 1; i <= num / 2; i++)
        {
            if (num % i == 0)
            {
                sum += i;
            }
        }

        return sum == num && num != 0;
    }
}
```



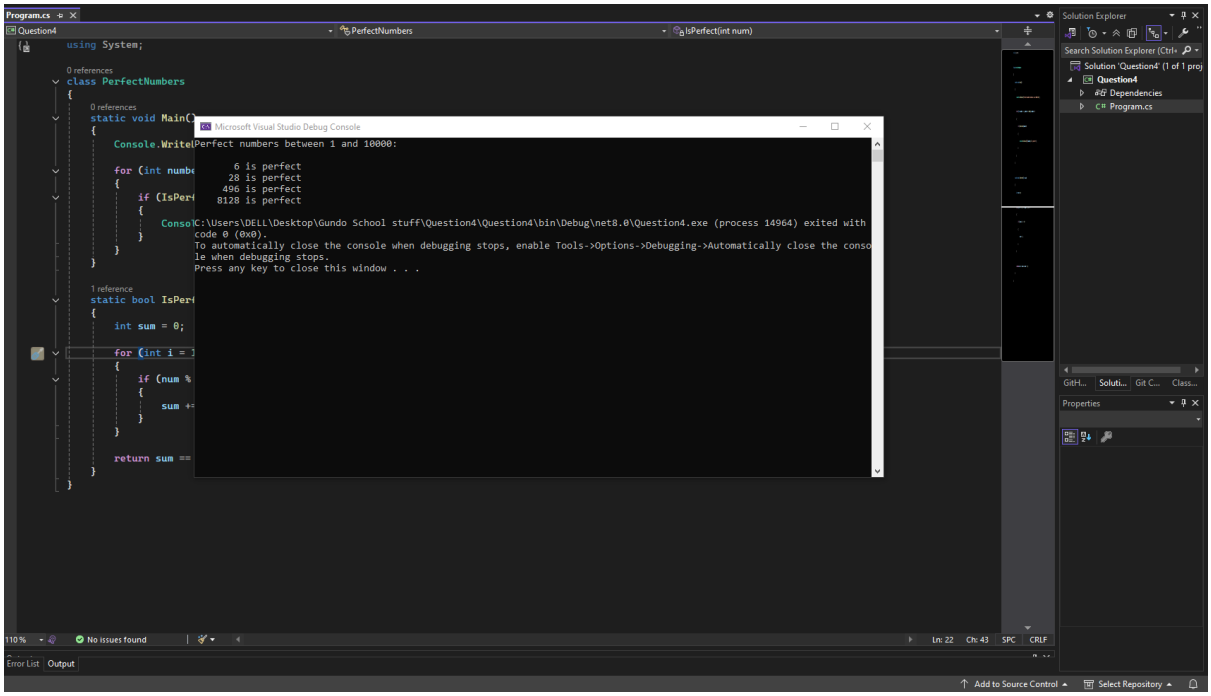
```
using System;

0 references
class PerfectNumbers
{
    0 references
    static void Main()
    {
        Console.WriteLine("Perfect numbers between 1 and 10000:\n");

        for (int number = 1; number <= 10000; number++)
        {
            if (IsPerfect(number))
            {
                Console.WriteLine($"{number,8} is perfect");
            }
        }
    }

    1 reference
    static bool IsPerfect(int num)
    {
        int sum = 0;

        for (int i = 1; i <= num / 2; i++)
        {
            if (num % i == 0)
            {
                sum += i;
            }
        }

        return sum == num && num != 0;
    }
}
```

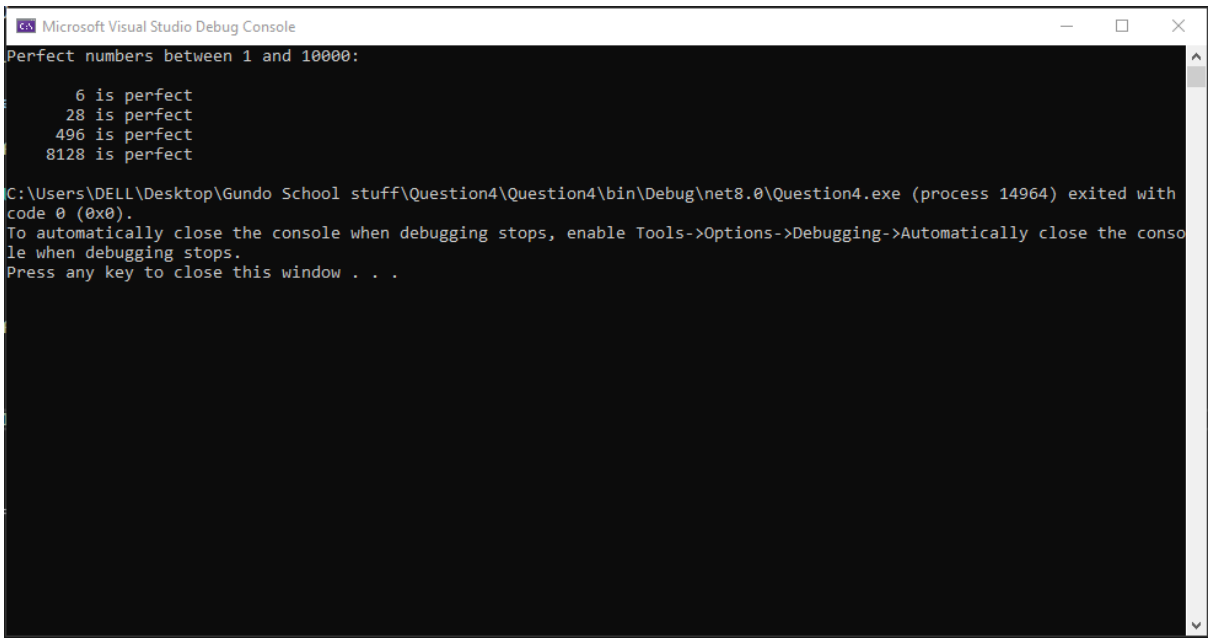
Source Code:

```
using System;


class PerfectNumbers
{
   static void Main()
   {
      Console.WriteLine("Perfect numbers between 1 and 10000:\n");


      for (int number = 1; number <= 10000; number++)
      {
         if (IsPerfect(number))
         {
            Console.WriteLine($"{number,8} is perfect");
         }
      }
   }


   static bool IsPerfect(int num)
   {
      int sum = 0;


      for (int i = 1; i <= num / 2; i++)
      {
         if (num % i == 0)
         {
            sum += i;
         }
      }


      return sum == num && num != 0;
```

```
    }
}
```

OUTPUT:


Microsoft Visual Studio Debug Console
Perfect numbers between 1 and 10000:
6 is perfect
28 is perfect
496 is perfect
8128 is perfect
C:\Users\DELL\Desktop\Gundo School stuff\Question4\Question4\bin\Debug\net8.0\Question4.exe (process 14964) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .


Microsoft Visual Studio Debug Console
Perfect numbers between 1 and 10000:
6 is perfect
28 is perfect
496 is perfect
8128 is perfect
C:\Users\DELL\Desktop\Gundo School stuff\Question4\Question4\bin\Debug\net8.0\Question4.exe (process 14964) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .