HOME WORK 25/08/2023

NAME: LÊ HỮU HOÀN

CLASS: C# Development (3)

Exercise 1: Dice

Use object-oriented programming method to create a C# program with the following requirements:

-Users enter a number of sides of Dice

-The program will generate a random number from 1 to the number of sides

What classes should we create?

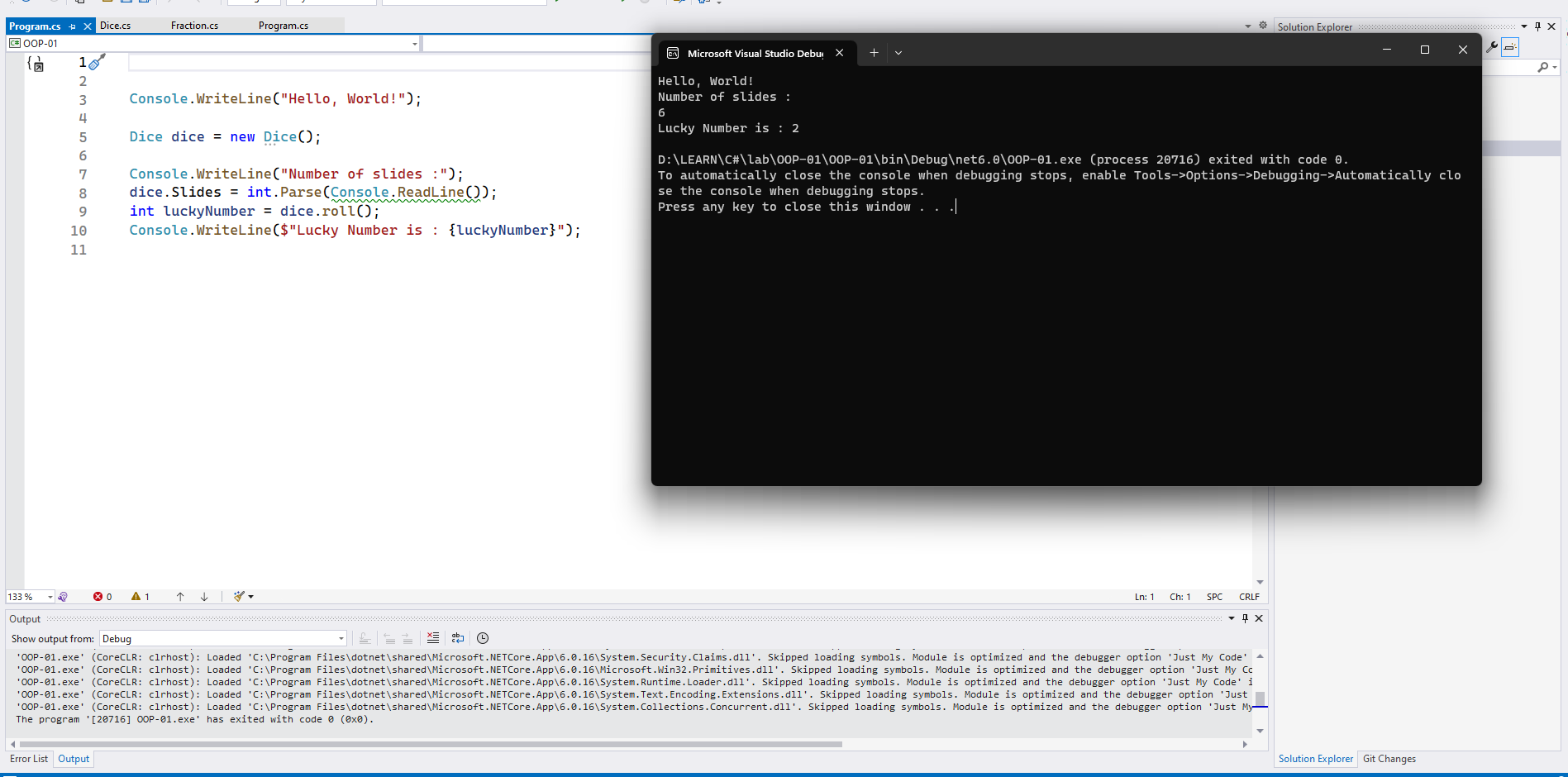
- Class: Dice

o Fields: slides

o Methods: roll (generate a random number from 1 to sides)

- A class to run the program

CONSOLE :



Code:

|  |
| --- |
| Dice dice = new Dice();  Console.WriteLine("Number of slides :");  dice.Slides = int.Parse(Console.ReadLine());  int luckyNumber = dice.roll();  Console.WriteLine($"Lucky Number is : {luckyNumber}"); |

|  |
| --- |
| public class Dice  {  private int slides;    public int Slides{  get { return slides; }  set  {  if (value <= 0) slides = 1;  else slides = value;  }  }  public int roll()  {  return new Random().Next(1, slides + 1);  }  } |

Exercise 2: Create classes with the following components  
CONSOLEA screenshot of a computer

Description automatically generated

Code:

|  |
| --- |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace OOP\_01\_02  {  internal class Person  {  private int age;  private string name;  public Person(string name,int age) {  this.name = name;  this.age = age;  }  public int Age  {  get { return age; }  set  {  if (value < 0 || value > 100) age = 0;  else age = value;  }  }  public string Name  {  get { return name; }  set { name = value; }  }  public void DisplayPerson()  {  Console.WriteLine($"name: {name}");  Console.WriteLine($"age: {age}");  }  public override string ToString()  {  return $"[ name:{name},age:{age}]";  }  public void Input()  {  Console.WriteLine("name:");  name = Console.ReadLine();  Console.WriteLine();  age = int.Parse( Console.ReadLine() );  }  }  } |

|  |
| --- |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using System.Xml.Linq;  namespace OOP\_01\_02  {  internal class Student:Person  {  private float gpa;  public Student(float gpa,string name, int age) : base(name,age)  {  Age = age;  Name = name;  this.gpa = gpa;  }  public float Gpa  {  get { return gpa; }  set {  if (value < 0 || value > 4) gpa = 0;  else gpa = value;  }  }  public void DisplayStudent()  {  Console.WriteLine($"gpa : {gpa}");  }    public void Input()  {  Console.WriteLine("GPA: ");  gpa = float.Parse( Console.ReadLine() );  }  }  } |
| using System;  namespace OOP\_01\_02  {  internal class Program  {  static void Main(string[] args)  {  Student student = new Student(3.2f, "Le Huu Hoan", 21);  student.DisplayPerson();  student.DisplayStudent();  student.ToString();  }  }  } |

Exercise 3:

Define a class Song, which holds the following information about songs: Type List, Name and Time.

On the first line you will receive the number of songs - N.

On the next N-lines you will be receiving data in the following format: &quot;{typeList}\_{name}\_{time}&quot;.

On the last line you will receive Type List / &quot;all&quot;. Print only the Names of the songs which are from that Type List / All songs.  
  
CONSOLE:

A computer screen shot of a black square with white text

Description automatically generated  
CODE:

|  |
| --- |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace OOP\_01\_03  {  internal class Song  {    public Song( string typelist , string name, string time)  {  Typelist = typelist;  Name = name;  Time = time;  }  public string Typelist { get; set; }  public string Name { get; set; }  public string Time { get; set ; }  }  } |

|  |
| --- |
| using System;  using System.Collections.Generic;  namespace OOP\_01\_03  {  internal class Program  {  static void Main(string[] args)  {  int num = int.Parse(Console.ReadLine());  List<Song> list = new List<Song>();  for ( int i = 0; i < num; i++ )  {  string[] data = Console.ReadLine().Split("\_");  list.Add(new Song(data[0], data[1], data[2]));  }  string type = Console.ReadLine();  if( type == "all")  {  foreach ( Song song in list )  {  Console.WriteLine(song.Name);  }  }  else {  foreach ( Song song in list)  {  if (song.Typelist==type)  {  Console.WriteLine(song.Name);  }  }  }  }  }  } |

Exercise 4: Make a Fraction class in C#

Create a class called Fraction to perform arithmetic operations with fractional numbers.

Write a program to test this class:

Use integer values to represent the attributes of the class – the numerator and the denominator.

Create public methods to perform the following:

 Input two fractions. The result is normalized.

 Add two fractions. The result is normalized.

 Subtract two fractions. The result is normalized.

 Multiply two fractions. The result is normalized.

 Divide two fractions. The result is normalized.

 Display a fraction as follows: (A/B). A is the numerator, B is the denominator

 Display a fraction on the screen as a decimal number.  
  
CONSOLE:

A screen shot of a computer

Description automatically generated

CODE:

|  |
| --- |
| using System;  namespace OOP\_01\_04  {  internal class Program  {  static void Main(string[] args)  {  Fraction fraction = new Fraction();  fraction.InputFractions();  Console.WriteLine("Add Fractions");  fraction.AddFractions();  fraction.DisplayFraction();  fraction.DisplayDecimal();  Console.WriteLine("Subtract Fractions");  fraction.SubtractFractions();  fraction.DisplayFraction();  fraction.DisplayDecimal();  Console.WriteLine("Divide Fractions");  fraction.DivideFractions();  fraction.DisplayFraction();  fraction.DisplayDecimal();  Console.WriteLine("Multiply Fractions");  fraction.MultiplyFractions();  fraction.DisplayFraction();  fraction.DisplayDecimal();  }  }  } |

|  |
| --- |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace OOP\_01\_04  {  internal class Fraction  {  private int n1;  private int n2;  private int d1;  private int d2;  private int numerator;  private int denominator;  public void InputFractions()  {  Console.WriteLine("Enter the first fraction:");  n1 = ReadInt("Numerator: ");  d1 = ReadInt("Denominator: ");  Console.WriteLine();  Console.WriteLine("Enter the second fraction:");  n2 = ReadInt("Numerator: ");  d2 = ReadInt("Denominator: ");  }  public void AddFractions()  {  numerator = n1 \* d2 + n2 \* d1;  denominator = d1\*d2;  }  public void SubtractFractions()  {  numerator = n1 \* d2 - n2 \* d1;  denominator =d1\*d2;  }  public void MultiplyFractions()  {  numerator = n1\*n2;  denominator = d1\*d2;  }  public void DivideFractions()  {  numerator = n1\*d2;  denominator = d1 \* n2;  }  public void DisplayFraction()  {  Console.WriteLine($"Fraction: ({numerator}/{denominator})");  }  public void DisplayDecimal()  {  float result = (float)numerator / denominator;  Console.WriteLine($"Decimal: {result}");  }  private int ReadInt(string message)  {  int value;  while (true)  {  Console.Write(message);  if (int.TryParse(Console.ReadLine(), out value))  return value;  Console.WriteLine("Invalid input. Please enter an integer.");  }  }  }  } |