Computer Programming Laboratory		
Laboratory 2: Data Types and Variable	**School of Information Technology**	
Name: Thu Rein Oo	**ID: 6731501110**	**Section:3**

Course Learning Outcomes (CLO)

1. **CLO1.1:** Students do their own work, do not present the work of others as their own, do not cheat on exams, and are responsible for completing their work.
2. **CLO2.3:** Students understand the principles of programming including emerging technology, and how to utilize them together.
3. **CLO6.1:** Students are able to complete their assigned work for each particular concept.

Lesson Learning Outcomes (LLOs)

1. **LLO1:** Explain the fundamental data types in C# (int, double, char, bool) and differentiate between value types and reference types in C#. [Aligns with **CLO2.3**]
2. **LLO2:** Declare and initialize variables of different data types in C#. [Aligns with **CLO2.3** and **CLO6.1**]

Lab Objectives

- To explain the data types and variables in C# [Aligns with **CLO2.3**]
- To write a program that declare and initialize variables [Aligns with **CLO2.3** and **CLO6.1**]
- To practice coding for inputting data to variables, processing data, and displaying variables [Aligns with **CLO2.3** and **CLO6.1]**

Remark:

- **CLO □ LLO □ Lab Objectives**
- Students are expected to complete the lab on their own, with no plagiarism or cheating. [Aligns with **CLO1.1**]

Instruction:

- Complete and answer all required questions (**Practice I, Practice II and Homework**)

Submission:

- Save this document file that contains all answers as "**Lab02_YourID.pdf**" and submit it to the MFU LMS system by the deadline.

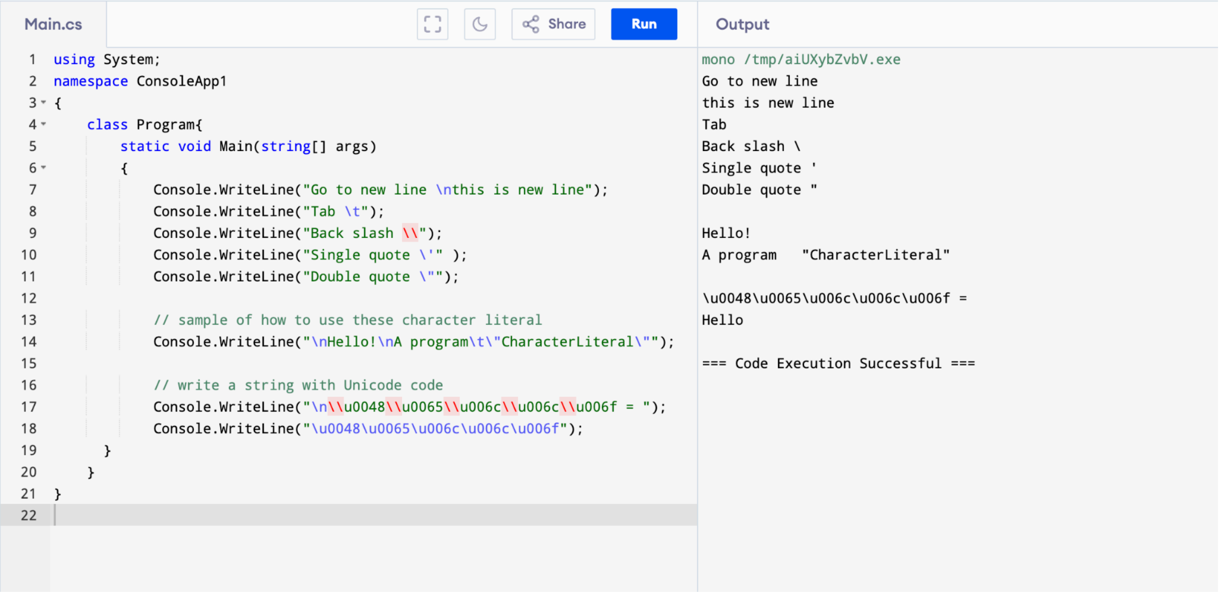
Practice I – Literal

This practice shall give a sample of code to demonstrate how to use literal for different data type.
Character Literal – The following code represents a sample of *character and string type literal.*

```
using System;
namespace ConsoleApp1
{
    class Program{
        static void Main(string[] args)
        {
                    Console.WriteLine("Go to new line \nthis is new line");
                    Console.WriteLine("Tab \t");
                    Console.WriteLine("Back slash \\");
                    Console.WriteLine("Single quote \'" );
                    Console.WriteLine("Double quote \"");

                    // sample of how to use these character literal
                    Console.WriteLine("\nHello!\nA program\t\"CharacterLiteral\"");

                    // write a string with Unicode code
                    Console.WriteLine("\n\\u0048\\u0065\\u006c\\u006c\\u006f = ");
                    Console.WriteLine("\u0048\u0065\u006c\u006c\u006f");
         }
    }
}
```

What is the result after running this program?

Answer:



```
using System;
namespace ConsoleApp1
{
    class Program{
        static void Main(string[] args)
        {
            Console.WriteLine("Go to new line \nthis is new line");
            Console.WriteLine("Tab \t");
            Console.WriteLine("Back slash \\");
            Console.WriteLine("Single quote \'" );
            Console.WriteLine("Double quote \"");

            // sample of how to use these character literal
            Console.WriteLine("\nHello!\nA program\t\"CharacterLiteral\"");

            // write a string with Unicode code
            Console.WriteLine("\n\\u0048\\u0065\\u006c\\u006c\\u006f = ");
            Console.WriteLine("\u0048\u0065\u006c\u006c\u006f");
        }
    }
}
```

```
mono /tmp/aiUXybZvbV.exe
Go to new line
this is new line
Tab
Back slash \
Single quote '
Double quote "

Hello!
A program   "CharacterLiteral"

\u0048\u0065\u006c\u006c\u006f =
Hello

=== Code Execution Successful ===
```

Practice II – Variables

This practice will provide a sample code for how to declare variables in a program.

Fahrenheit to Celsius:

$$C=(F-32)\cdot\frac{5}{9}$$

Celsius to Fahrenheit:

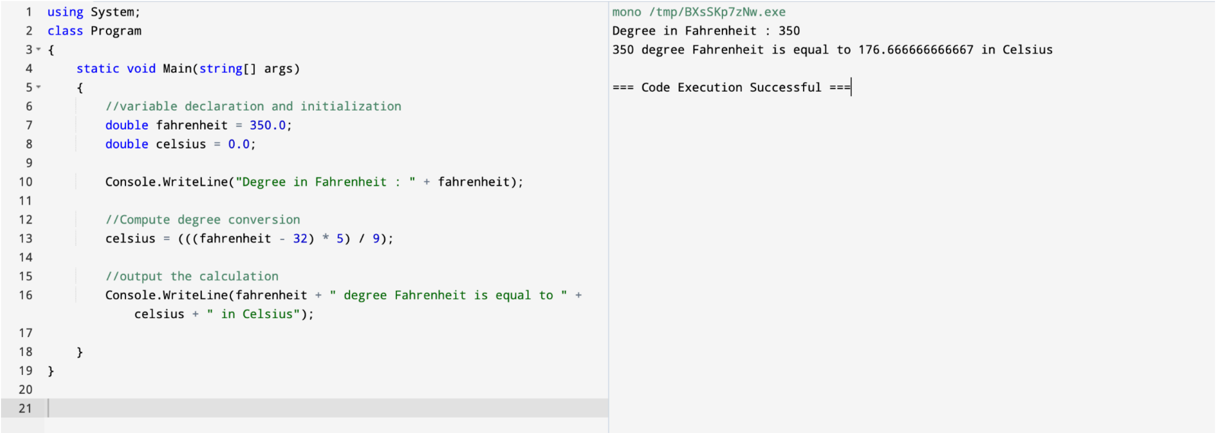
$$F=C\cdot\frac{9}{5}+32$$

Temperature Conversion: This program **converts 350-degree Fahrenheit** to a **Celsius degree.**

```
using System;
class Program
{
     static void Main(string[] args)
     {
          //variable declaration and initialization
          double fahrenheit = 350.0;
          double celsius = 0.0;

          Console.WriteLine("Degree in Fahrenheit : " + fahrenheit);

          //Compute degree conversion
          celsius = (((fahrenheit - 32) * 5) / 9);

          //output the calculation
          Console.WriteLine(fahrenheit + " degree Fahrenheit is equal to " +
celsius + " in Celsius");

     }
}
```

Question 1:

1.1) What is the result after running this program?



```
using System;
class Program
{
    static void Main(string[] args)
    {
        //variable declaration and initialization
        double fahrenheit = 350.0;
        double celsius = 0.0;

        Console.WriteLine("Degree in Fahrenheit : " + fahrenheit);

        //Compute degree conversion
        celsius = (((fahrenheit - 32) * 5) / 9);

        //output the calculation
        Console.WriteLine(fahrenheit + " degree Fahrenheit is equal to " +
            celsius + " in Celsius");

    }
}

```

```
mono /tmp/BXsSKp7zNw.exe
Degree in Fahrenheit : 350
350 degree Fahrenheit is equal to 176.666666666667 in Celsius

=== Code Execution Successful ===
```

1.2) How many value type variables are there in the program? What are their variable names?

Answer: 1 and that is “double”

1.3) Based on question 1.2, what are the statements that represent these variable declarations and initializations?

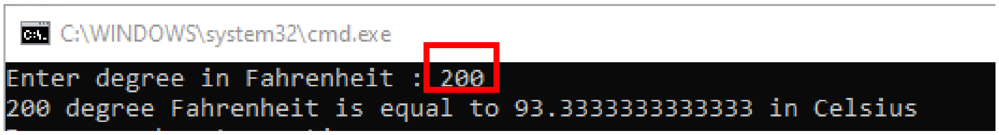
Answer:

```
double fahrenheit = 350.0;

double celsius = 0.0;
```

Question 2: User input part

Please modify the coding in question 1. "Temperature" (original program) to demonstrate how to get keyboard input instead of assigning value directly in code; for the value of **Fahrenheit**.

Result:



```
Enter degree in Fahrenheit : 200
200 degree Fahrenheit is equal to 93.3333333333333 in Celsius
```

Answer: source code

```
using System;
class Program
{
    static void Main(string[] args)
    {
        //variable declaration and initialization
        double fahrenheit = 350.0;
        double celsius = 0.0;
        Console.WriteLine("Please Input Your temp in fahrenheit : ");
        fahrenheit = Convert.ToDouble(Console.ReadLine());

        Console.WriteLine("Degree in Fahrenheit : " + fahrenheit);

        //Compute degree conversion
        celsius = (((fahrenheit - 32) * 5) / 9);

        //output the calculation
        Console.WriteLine(fahrenheit + " degree Fahrenheit is equal to " + celsius +
" in Celsius");

    }
}
```

Question 3:

Modify the coding in question 2 to convert from **Celsius** to **Fahrenheit**

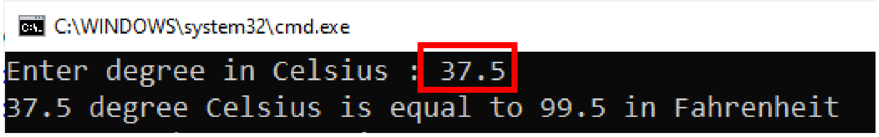
Fahrenheit to Celsius:

$$C = (F - 32) \cdot \frac{5}{9}$$

Celsius to Fahrenheit:

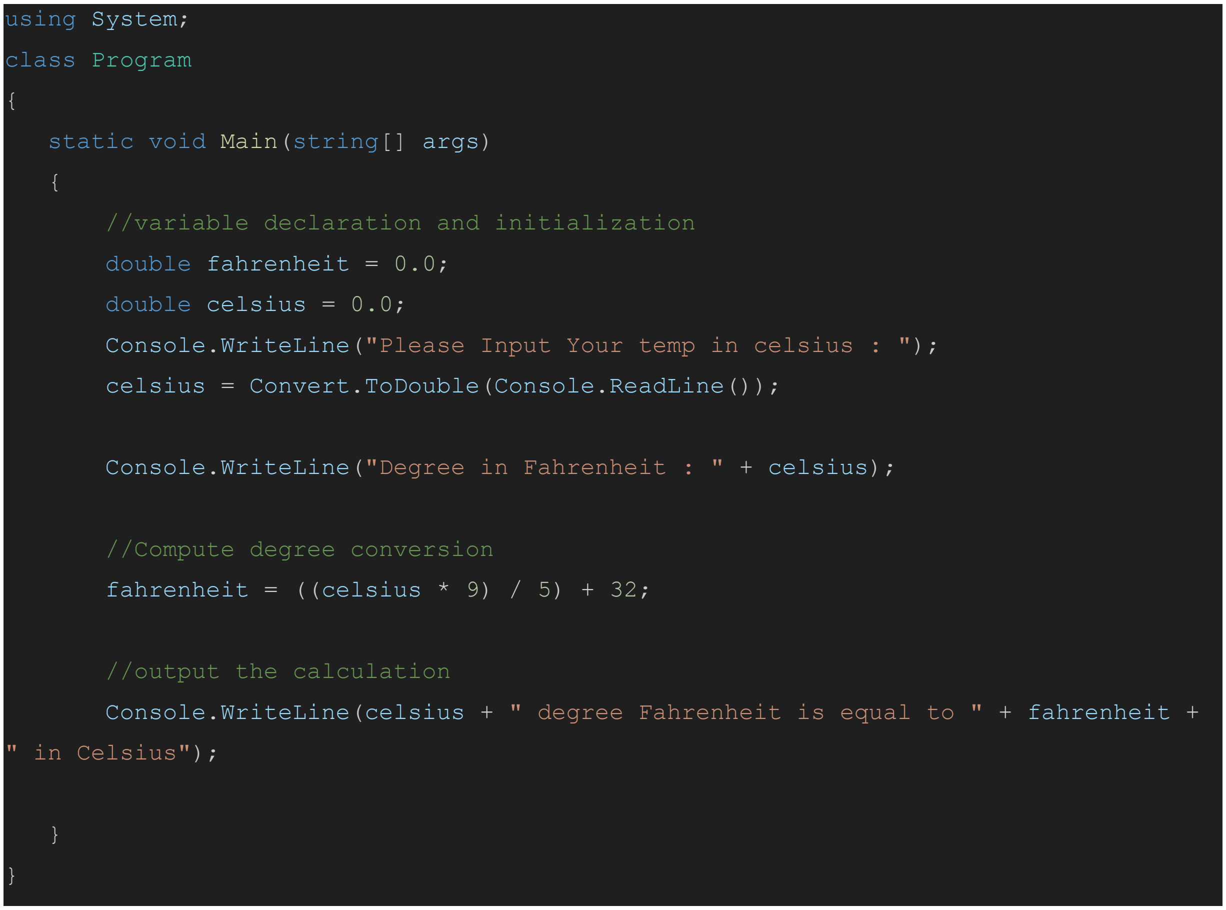
$$F = C \cdot \frac{9}{5} + 32$$

Result:



```
Enter degree in Celsius : 37.5
37.5 degree Celsius is equal to 99.5 in Fahrenheit
```

Answer: source code

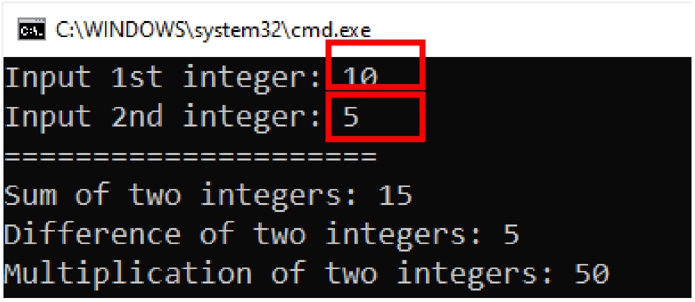


```
using System;
class Program
{
   static void Main(string[] args)
   {
       //variable declaration and initialization
       double fahrenheit = 0.0;
       double celsius = 0.0;
       Console.WriteLine("Please Input Your temp in celsius : ");
       celsius = Convert.ToDouble(Console.ReadLine());

       Console.WriteLine("Degree in Fahrenheit : " + celsius);

       //Compute degree conversion
       fahrenheit = ((celsius * 9) / 5) + 32;

       //output the calculation
       Console.WriteLine(celsius + " degree Fahrenheit is equal to " + fahrenheit +
" in Celsius");

   }
}
```

[Homework]

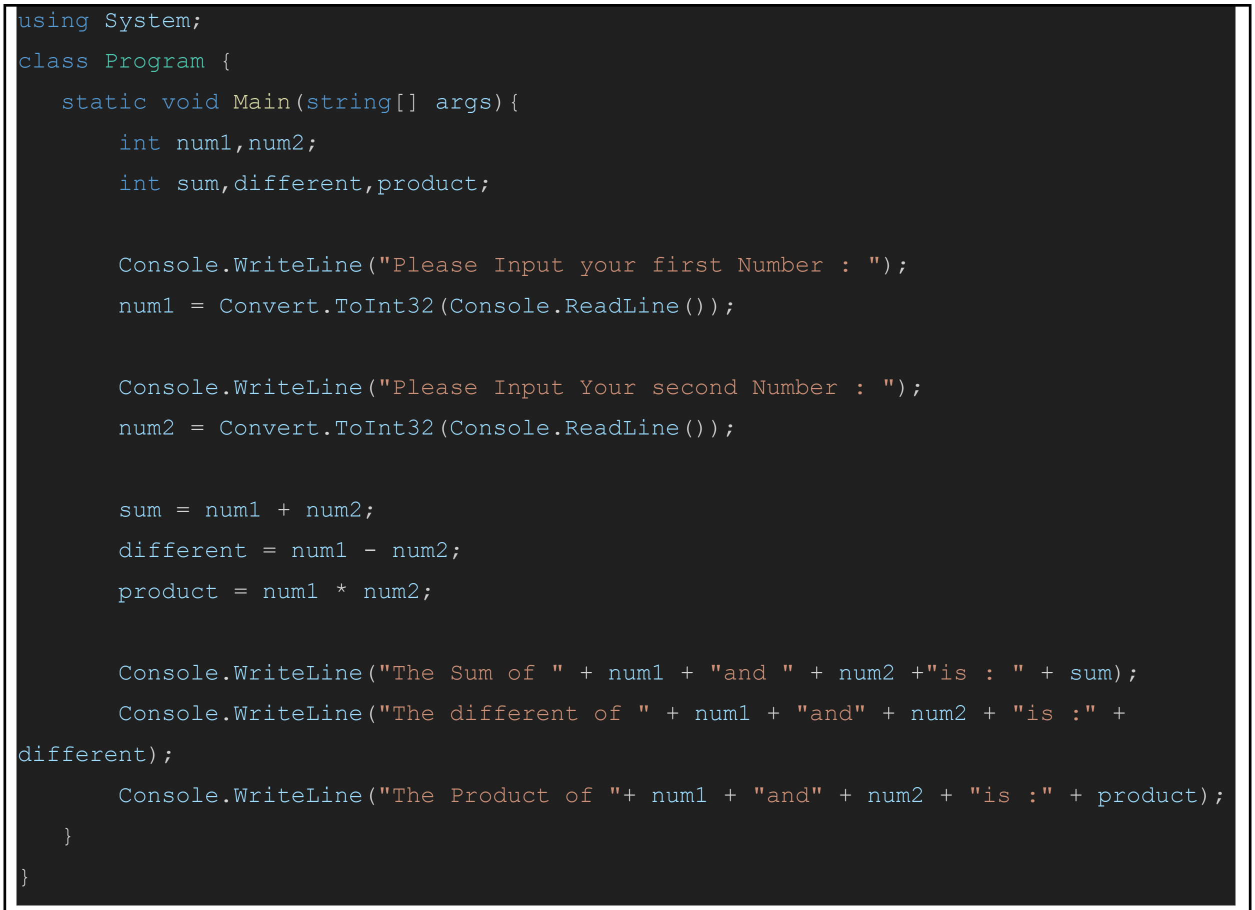
Question 1: Write a program that accepts two integer inputs from the user, computes and shows the **sum, difference,** and **multiplication.**

Result:



```
Input 1st integer: 10
Input 2nd integer: 5
=====================
Sum of two integers: 15
Difference of two integers: 5
Multiplication of two integers: 50
```

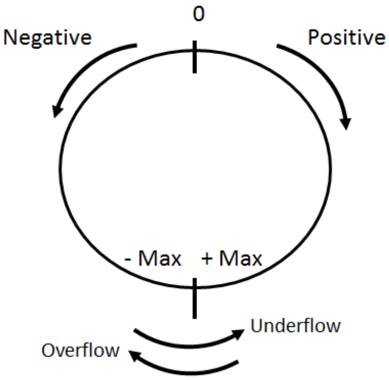
Answer: source code



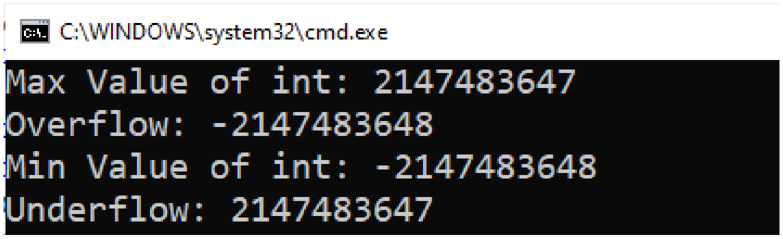
```
using System;
class Program {
   static void Main(string[] args){
       int num1,num2;
       int sum,different,product;

       Console.WriteLine("Please Input your first Number : ");
       num1 = Convert.ToInt32(Console.ReadLine());

       Console.WriteLine("Please Input Your second Number : ");
       num2 = Convert.ToInt32(Console.ReadLine());

       sum = num1 + num2;
       different = num1 - num2;
       product = num1 * num2;

       Console.WriteLine("The Sum of " + num1 + "and " + num2 +"is : " + sum);
       Console.WriteLine("The different of " + num1 + "and" + num2 + "is :" +
different);
       Console.WriteLine("The Product of "+ num1 + "and" + num2 + "is :" + product);
   }
}
```

[Optional Question – Challenge] - This part is not required for completion and submission.

Question #1 The following code demonstrates **overflow** and **underflow values** for integers.

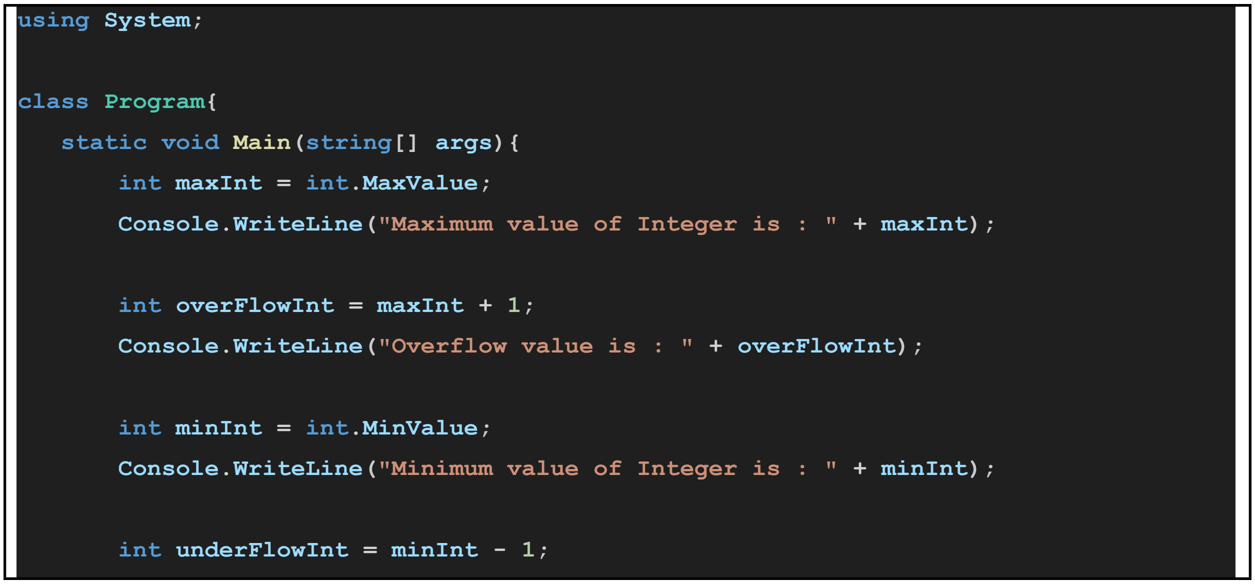


Result:



```
Max Value of int: 2147483647
Overflow: -2147483648
Min Value of int: -2147483648
Underflow: 2147483647
```

Answer: source code



```
using System;

class Program{
   static void Main(string[] args){
       int maxInt = int.MaxValue;
       Console.WriteLine("Maximum value of Integer is : " + maxInt);

       int overFlowInt = maxInt + 1;
       Console.WriteLine("Overflow value is : " + overFlowInt);

       int minInt = int.MinValue;
       Console.WriteLine("Minimum value of Integer is : " + minInt);

       int underFlowInt = minInt - 1;
```

```
        Console.WriteLine("UnderFlowInt value of Integer is :   " + underFlowInt);
    }


}
```