DAY 12 : Assignment
By
Vihar D.

Assignment 1

What is Exception Handling and why do we need exception handling ?

Answer :

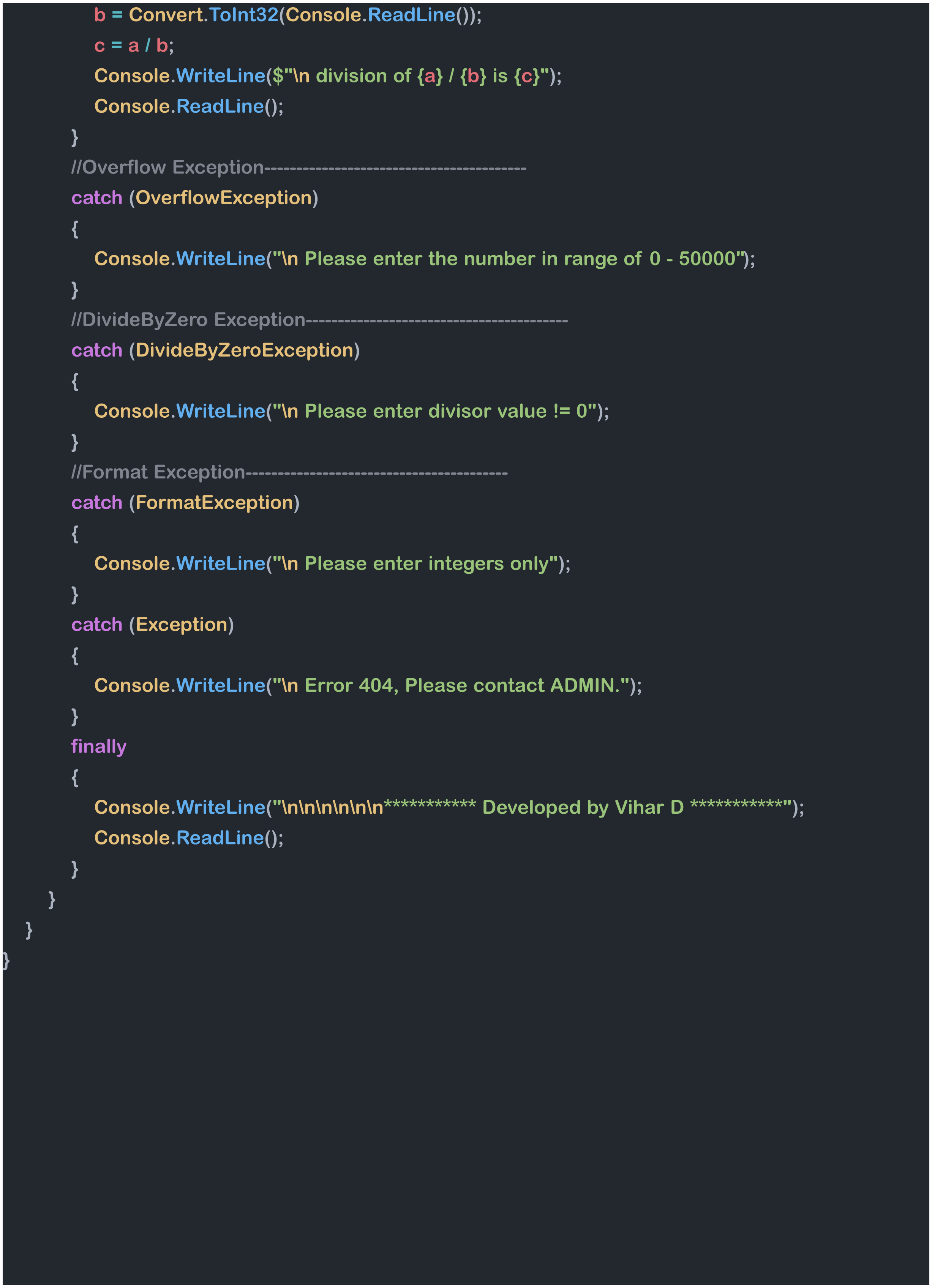
- **Exception Handling is done to ensure that the application will not crash.**
- **It will not display any technical details and to make sure we handle errors gracefully and display friendly messages.**
- **Exception is an event or object which is thrown at runtime.**
- **Exceptions are derived from the system namespace.**

Assignment 2

Write a C# code for division program, add exceptions along with super exception at the end.

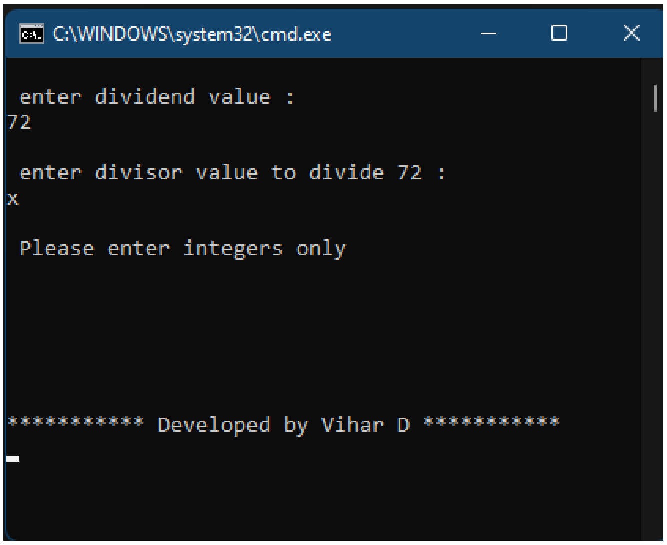
Answer :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace div_3except_super
{
    internal class Program
    {
        static void Main(string[] args)
        {
            try
            {
                int a, b, c;
                Console.WriteLine("\n enter dividend value : ");
                a = Convert.ToInt32(Console.ReadLine());
                Console.WriteLine($"\n enter divisor value to divide {a} : ");
```



```
                b = Convert.ToInt32(Console.ReadLine());
                c = a / b;
                Console.WriteLine($"\n division of {a} / {b} is {c}");
                Console.ReadLine();
            }
            //Overflow Exception-----------------------------------------
            catch (OverflowException)
            {
                Console.WriteLine("\n Please enter the number in range of 0 - 50000");
            }
            //DivideByZero Exception-----------------------------------------
            catch (DivideByZeroException)
            {
                Console.WriteLine("\n Please enter divisor value != 0");
            }
            //Format Exception-----------------------------------------
            catch (FormatException)
            {
                Console.WriteLine("\n Please enter integers only");
            }
            catch (Exception)
            {
                Console.WriteLine("\n Error 404, Please contact ADMIN.");
            }
            finally
            {
                Console.WriteLine("\n\n\n\n\n\n*********** Developed by Vihar D ***********");
                Console.ReadLine();
            }
        }
    }
}
```

Output :



```
 enter dividend value :
72

 enter divisor value to divide 72 :
x

 Please enter integers only




*********** Developed by Vihar D ***********
```

Assignment 3

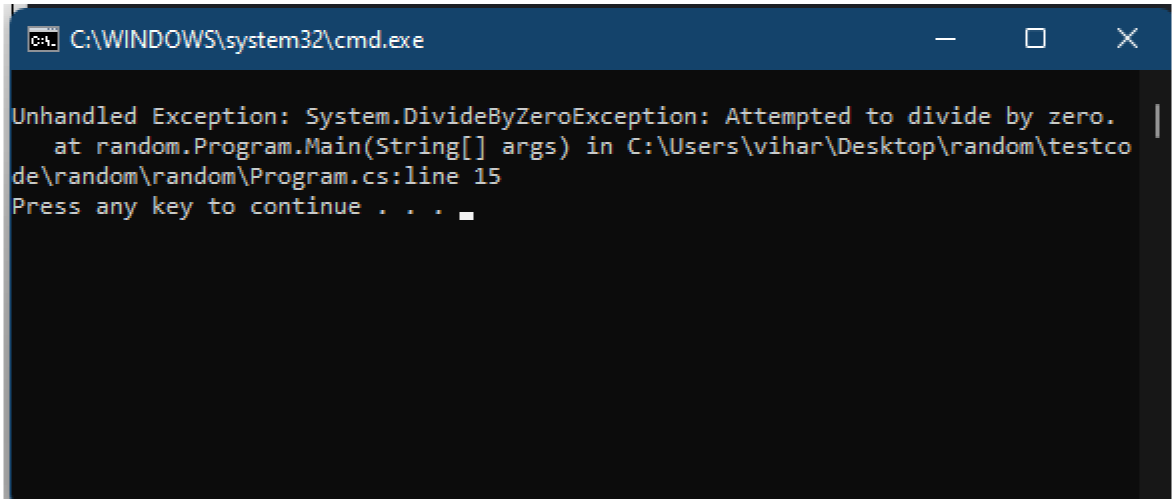
Research and write at least 6 exceptions that occur in C# with code snippets.

Answer :

i : DivideByZeroException (Code)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    internal class Program
    {
        //DivideByZeroException
        static void Main(string[] args)
        {
            int a = 5;
            int b = 0;
            int c = a / b;
            Console.WriteLine(c);
        }
    }
}
```

Output :

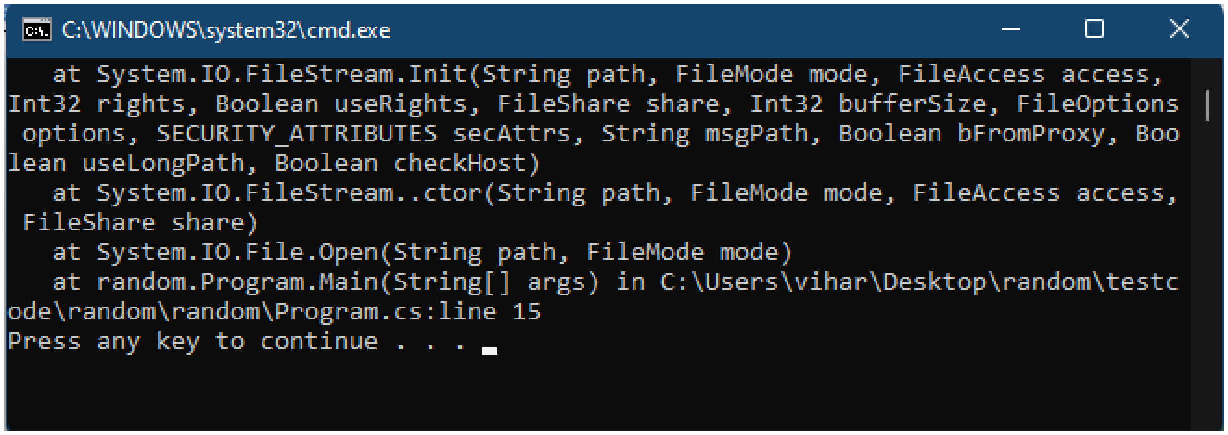


```
Unhandled Exception: System.DivideByZeroException: Attempted to divide by zero.
   at random.Program.Main(String[] args) in C:\Users\vihar\Desktop\random\testco
de\random\random\Program.cs:line 15
Press any key to continue . . .
```

ii : FileNotFoundException (Code)

```
using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    internal class Program
    {
        //FileNotfoundException
        static void Main(string[] args)
        {
            File.Open("d:\\random.txt", FileMode.Open);
        }
    }
}
```

Output :

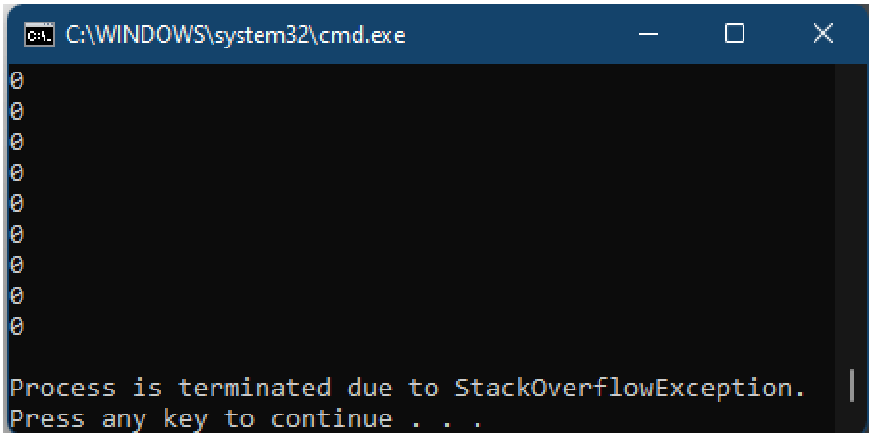


```
   at System.IO.FileStream.Init(String path, FileMode mode, FileAccess access,
Int32 rights, Boolean useRights, FileShare share, Int32 bufferSize, FileOptions
 options, SECURITY_ATTRIBUTES secAttrs, String msgPath, Boolean bFromProxy, Boo
lean useLongPath, Boolean checkHost)
   at System.IO.FileStream..ctor(String path, FileMode mode, FileAccess access,
 FileShare share)
   at System.IO.File.Open(String path, FileMode mode)
   at random.Program.Main(String[] args) in C:\Users\vihar\Desktop\random\testc
ode\random\random\Program.cs:line 15
Press any key to continue . . .
```

iii : StackOverFlowException (Code)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    internal class Program
    {
        static void Postinc(int x)
        {
            Console.WriteLine(x);
            Postinc(x++);
        }
        //StackOverFlowException
        static void Main(string[] args)
        {
            Postinc(0);
        }
    }
}
```

Output :

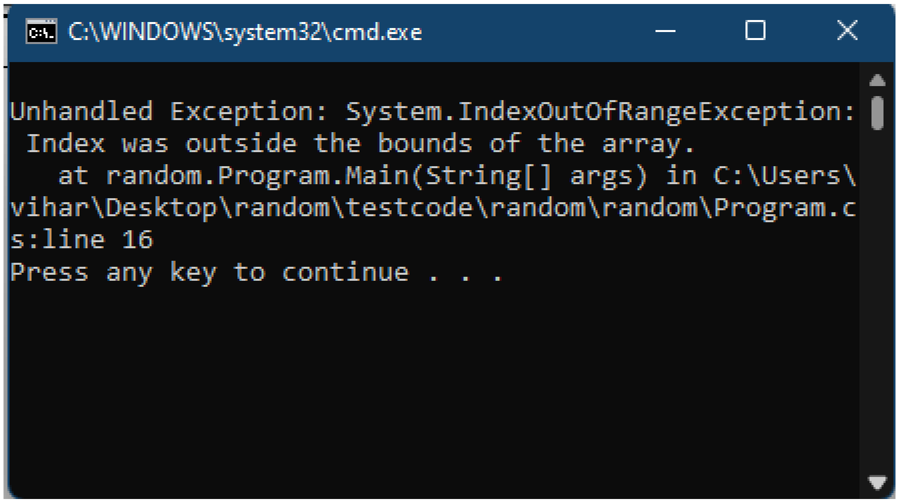


```
0
0
0
0
0
0
0
0
0

Process is terminated due to StackOverflowException.
Press any key to continue . . .
```

iv : IndexOutOfRangeException (Code)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    internal class Program
    {
        //IndexOutOfRangeException
        static void Main(string[] args)
        {
            int[] num = new int[10];
            num[0] = 5;
            num[10] = 10;
            num[15] = 15;
        }
    }
}
```

Output :

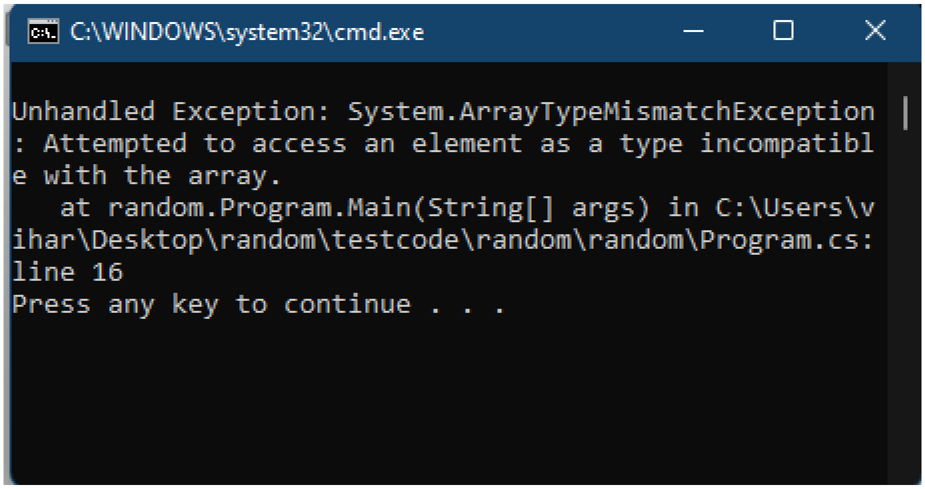


```
Unhandled Exception: System.IndexOutOfRangeException:
 Index was outside the bounds of the array.
   at random.Program.Main(String[] args) in C:\Users\
vihar\Desktop\random\testcode\random\random\Program.c
s:line 16
Press any key to continue . . .
```

v : ArrayTypeMismatchException (Code)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    internal class Program
    {
        //ArrayTypeMismatchException
        static void Main(string[] args)
        {
            string[] array1 = { "Hello", "World" };
            object[] array2 = array1;
            array2[0] = 100;
        }
    }
}
```

Output :

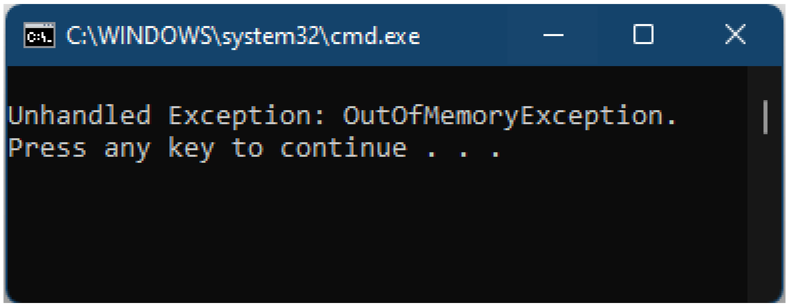


```
Unhandled Exception: System.ArrayTypeMismatchException
: Attempted to access an element as a type incompatibl
e with the array.
   at random.Program.Main(String[] args) in C:\Users\v
ihar\Desktop\random\testcode\random\random\Program.cs:
line 16
Press any key to continue . . .
```

vi : OutOfMemoryException (Code)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    internal class Program
    {
        //OutOfMemoryException
        static void Main(string[] args)
        {
            string str1 = new string('x', int.MaxValue);
        }
    }
}
```

Output :



```
Unhandled Exception: OutOfMemoryException.
Press any key to continue . . .
```

Assignment 4

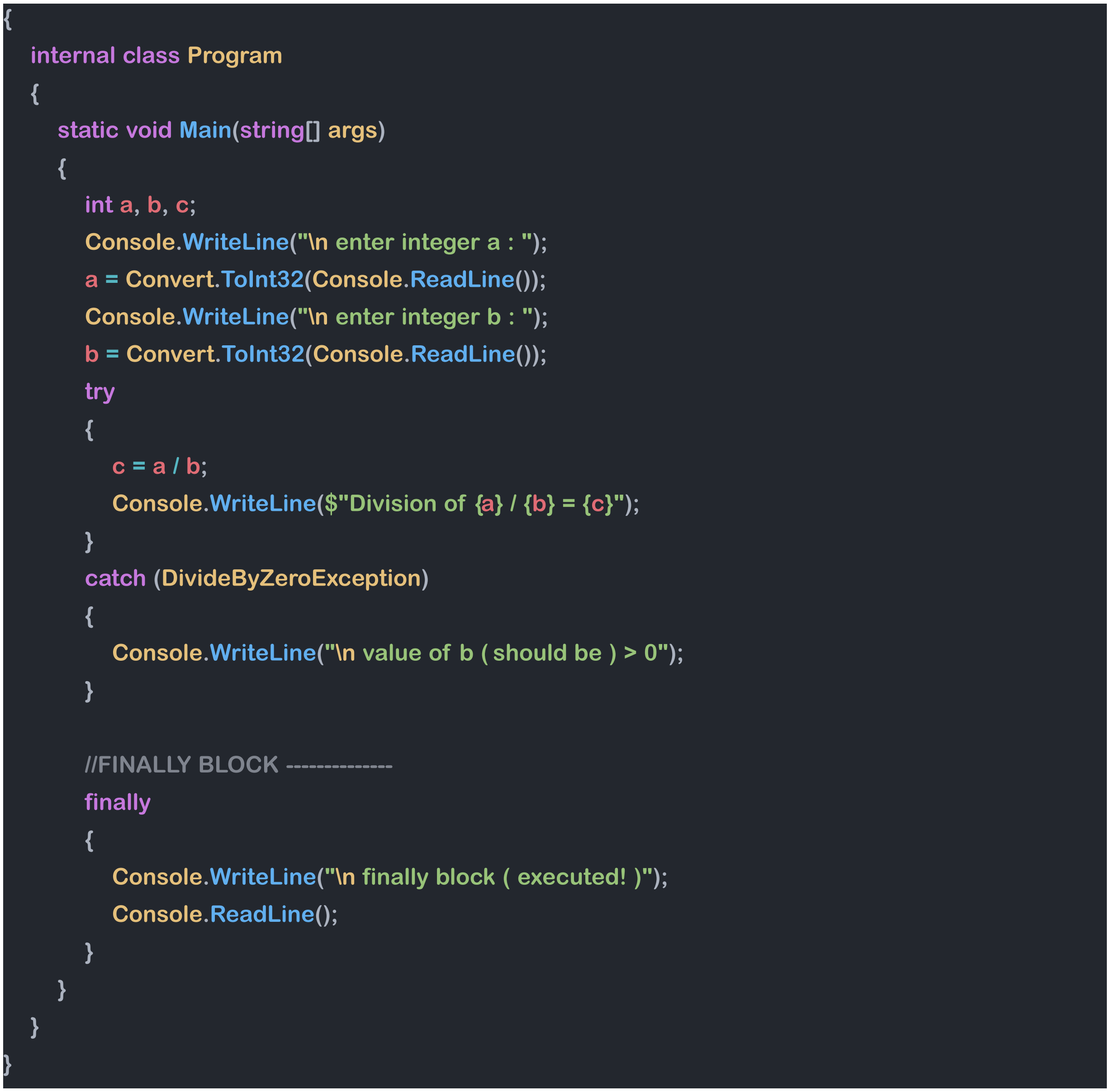
What is the use of finally block and illustrate with an example ?

Answer :

The finally block refers to a block of statements which will always be executed regardless of other exceptions that may or may not occur during an execution.

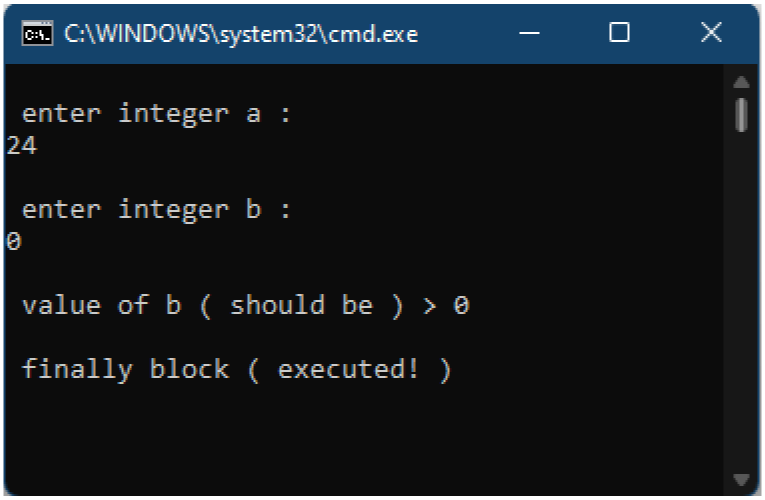
Finally Block (code) :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
```



```
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int a, b, c;
            Console.WriteLine("\n enter integer a : ");
            a = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("\n enter integer b : ");
            b = Convert.ToInt32(Console.ReadLine());
            try
            {
                c = a / b;
                Console.WriteLine($"Division of {a} / {b} = {c}");
            }
            catch (DivideByZeroException)
            {
                Console.WriteLine("\n value of b ( should be ) > 0");
            }

            //FINALLY BLOCK --------------
            finally
            {
                Console.WriteLine("\n finally block ( executed! )");
                Console.ReadLine();
            }
        }
    }
}
```

Output :



```
 enter integer a :
24

 enter integer b :
0

 value of b ( should be ) > 0

 finally block ( executed! )
```

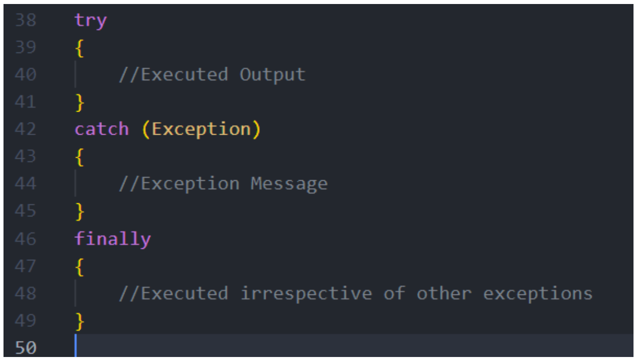
Assignment 5

What are the points about Exception Handling discussed in the session ?

Answer :

- **Exception Handling is done to handle errors gracefully.**
- **General exception block must be implemented at the end of all catch blocks.**
- **Finally block statements are implemented irrespective of other exceptions.**
- **One <u>try</u> block can have multiple <u>catch</u> blocks.**

<u>SYNTAX</u> :



```
try
{
    //Executed Output
}
catch (Exception)
{
    //Exception Message
}
finally
{
    //Executed irrespective of other exceptions
}
```

Assignment 6

What are <u>Compilation</u> & <u>Runtime</u> errors and write at least 3 differences ?

Answer :

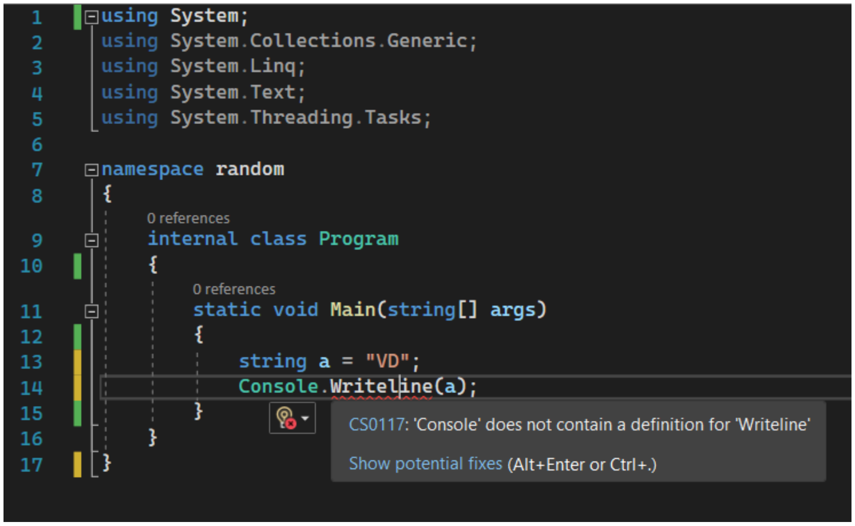
Compilation Error	Runtime Error
- This error more often refers to semantic or syntax errors.	- This error refers to what we encounter during code execution during runtime screen.
- Usually occurred due to developer mistakes.	- They usually are logical errors.
- Easy to resolve these errors.	- These errors are hard to identify.

Assignment 7

Write any 5 compilation errors with code snippets and screenshots

Answer :

1. **Improper naming conventions :**

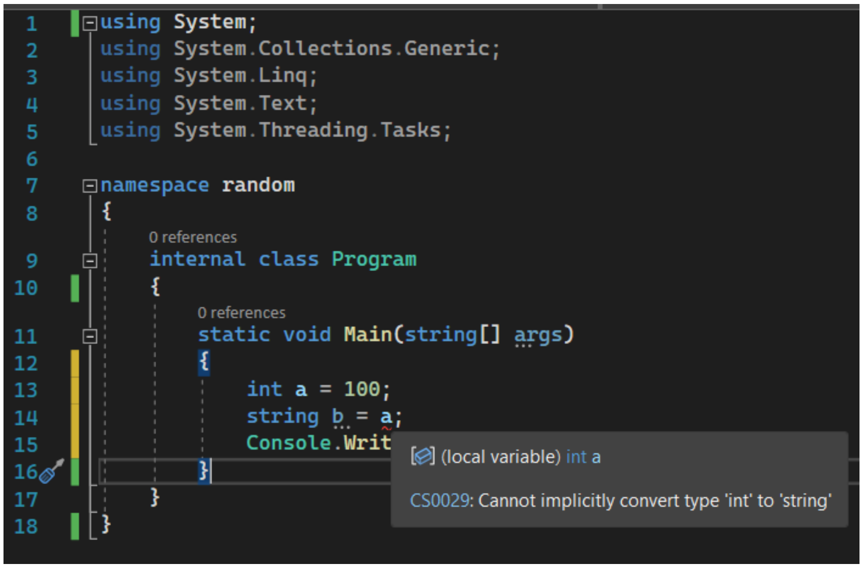


```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            string a = "VD";
            Console.Writeline(a);
        }
    }
}
```

CS0117: 'Console' does not contain a definition for 'Writeline'

Show potential fixes (Alt+Enter or Ctrl+.)

2. **Improper knowledge on type casting :**

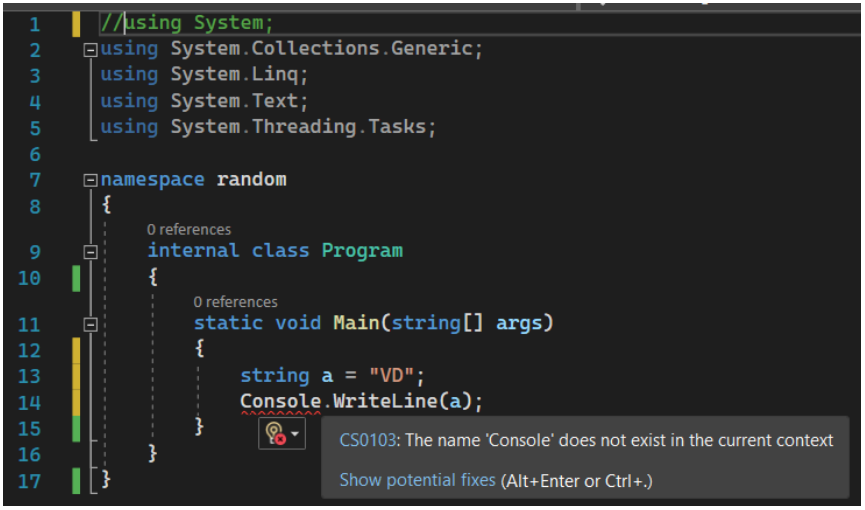


```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            int a = 100;
            string b = a;
            Console.Writ
        }
    }
}
```

(local variable) int a

CS0029: Cannot implicitly convert type 'int' to 'string'

3. **Improper importing of namespace** :

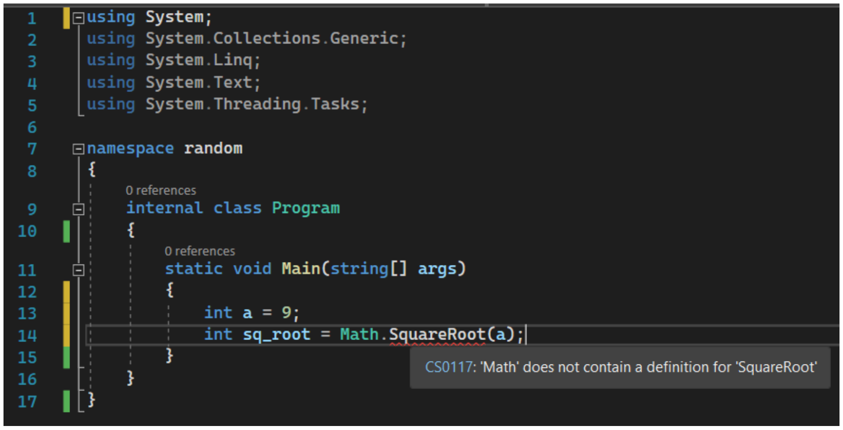


```
//using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            string a = "VD";
            Console.WriteLine(a);
        }
    }
}
```

CS0103: The name 'Console' does not exist in the current context

Show potential fixes (Alt+Enter or Ctrl+.)

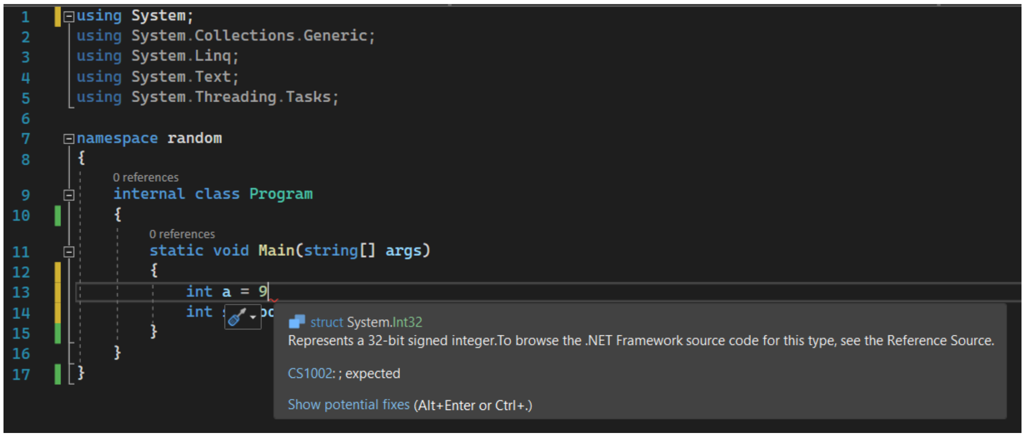
4. **Improper usage of methods and properties** :



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            int a = 9;
            int sq_root = Math.SquareRoot(a);
        }
    }
}
```

CS0117: 'Math' does not contain a definition for 'SquareRoot'

5. **Improper Semicolon placement** :



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            int a = 9
            int   bc
        }
    }
}
```

struct System.Int32
Represents a 32-bit signed integer.To browse the .NET Framework source code for this type, see the Reference Source.

CS1002: ; expected

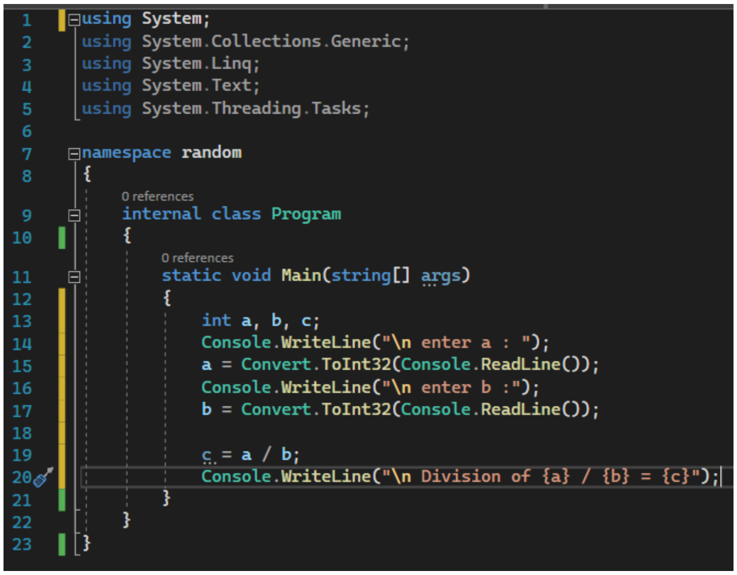
Show potential fixes (Alt+Enter or Ctrl+.)

Assignment 8

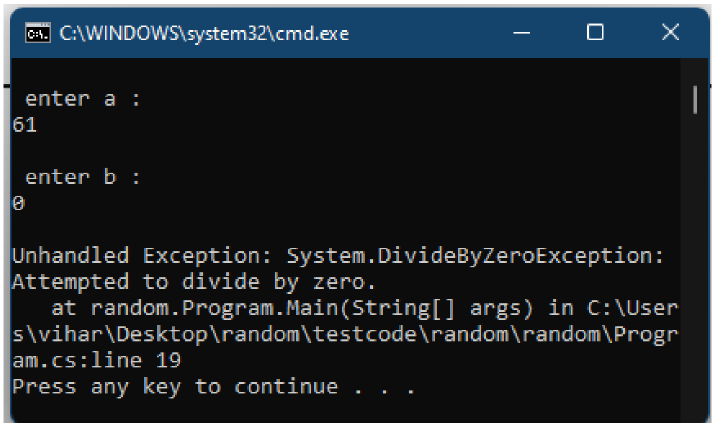
Write any 6 runtime errors with code snippets and screenshots.

Answer :

1. **DivideByZeroException :**



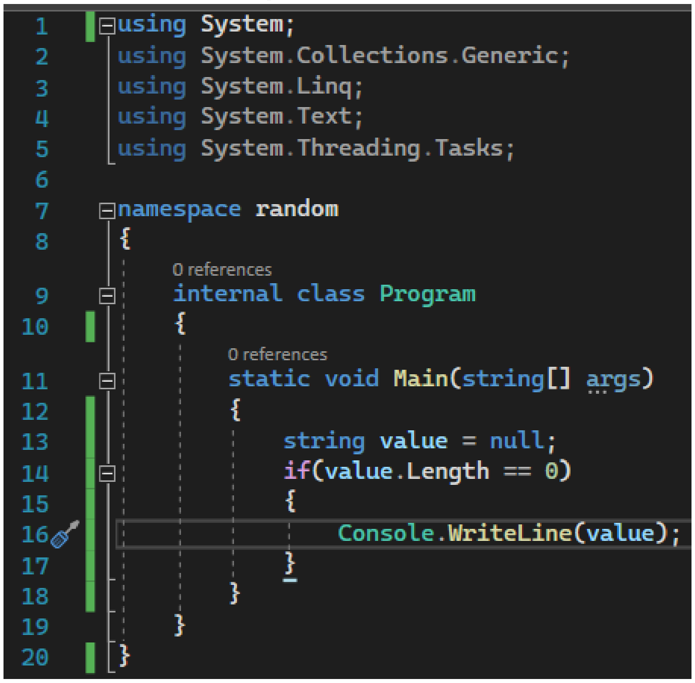
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            int a, b, c;
            Console.WriteLine("\n enter a : ");
            a = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("\n enter b :");
            b = Convert.ToInt32(Console.ReadLine());

            c = a / b;
            Console.WriteLine("\n Division of {a} / {b} = {c}");
        }
    }
}
```



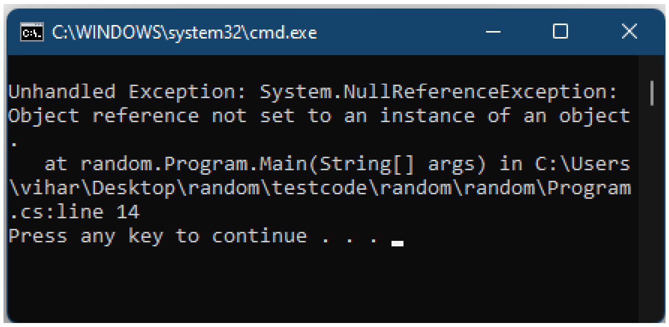
C:\WINDOWS\system32\cmd.exe

```

 enter a :
61

 enter b :
0

Unhandled Exception: System.DivideByZeroException:
Attempted to divide by zero.
   at random.Program.Main(String[] args) in C:\User
s\vihar\Desktop\random\testcode\random\random\Progr
am.cs:line 19
Press any key to continue . . .
```

2. NullReferenceException :

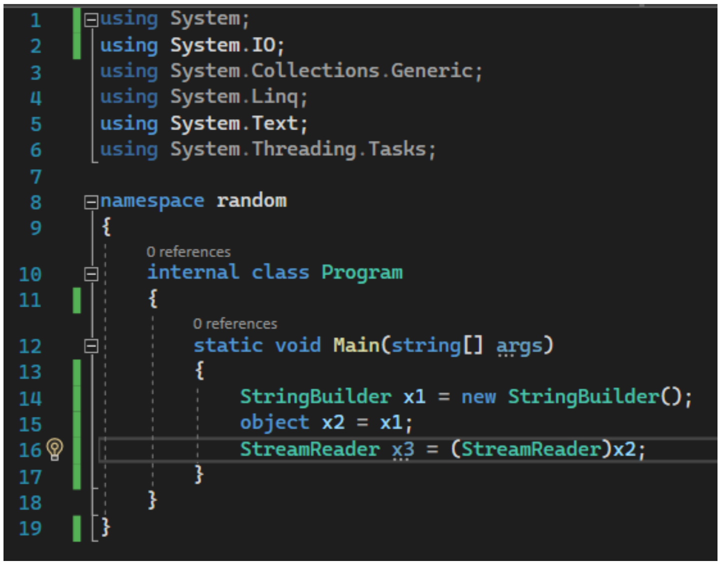


```
 1  using System;
 2  using System.Collections.Generic;
 3  using System.Linq;
 4  using System.Text;
 5  using System.Threading.Tasks;
 6
 7  namespace random
 8  {
            0 references
 9      internal class Program
10      {
                0 references
11          static void Main(string[] args)
12          {
13              string value = null;
14              if(value.Length == 0)
15              {
16                  Console.WriteLine(value);
17              }
18          }
19      }
20  }
```

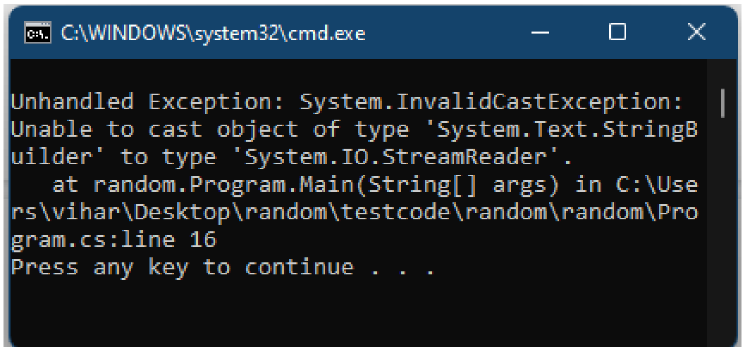


```
Unhandled Exception: System.NullReferenceException:
Object reference not set to an instance of an object
.
   at random.Program.Main(String[] args) in C:\Users
\vihar\Desktop\random\testcode\random\random\Program
.cs:line 14
Press any key to continue . . .
```

3. InvalidCastException :

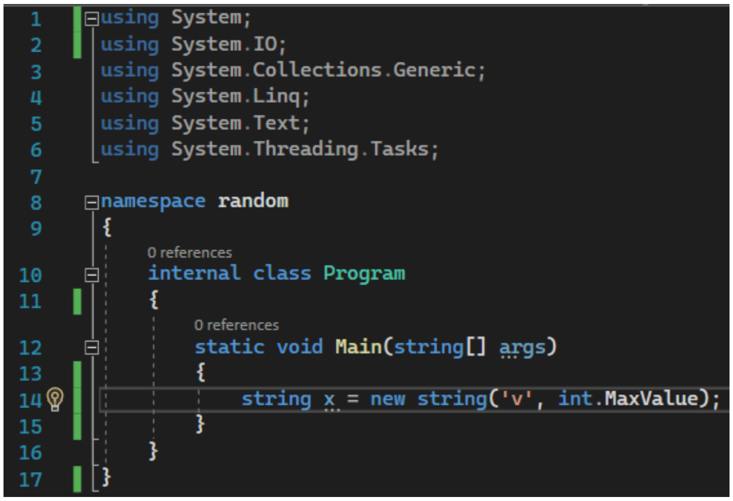


```
using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    internal class Program
    {
        static void Main(string[] args)
        {
            StringBuilder x1 = new StringBuilder();
            object x2 = x1;
            StreamReader x3 = (StreamReader)x2;
        }
    }
}
```

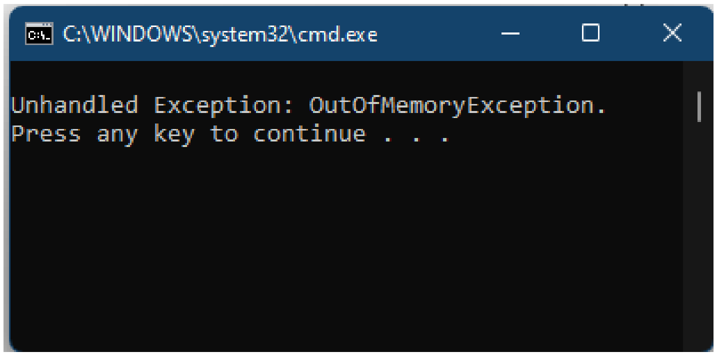


```
Unhandled Exception: System.InvalidCastException:
Unable to cast object of type 'System.Text.StringB
uilder' to type 'System.IO.StreamReader'.
   at random.Program.Main(String[] args) in C:\Use
rs\vihar\Desktop\random\testcode\random\random\Pro
gram.cs:line 16
Press any key to continue . . .
```

4. OutOfMemoryException :

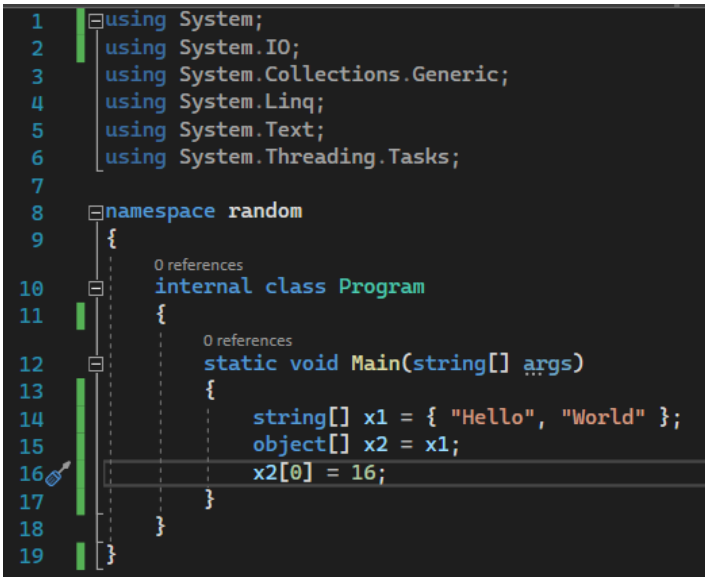


```
using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    internal class Program
    {
        static void Main(string[] args)
        {
            string x = new string('v', int.MaxValue);
        }
    }
}
```

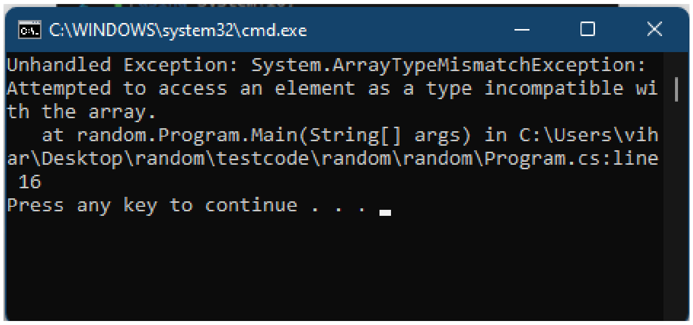


```
Unhandled Exception: OutOfMemoryException.
Press any key to continue . . .
```

5. ArrayTypeMismatchException :

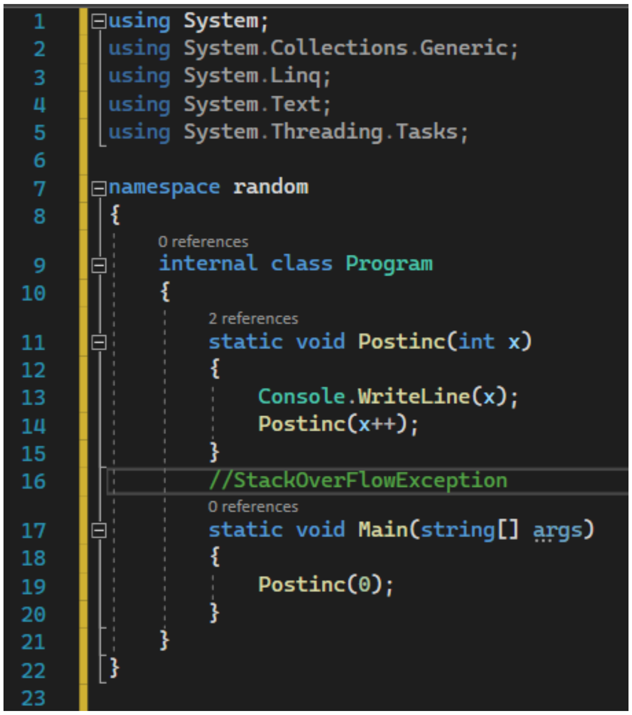


```csharp
using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    internal class Program
    {
        static void Main(string[] args)
        {
            string[] x1 = { "Hello", "World" };
            object[] x2 = x1;
            x2[0] = 16;
        }
    }
}
```

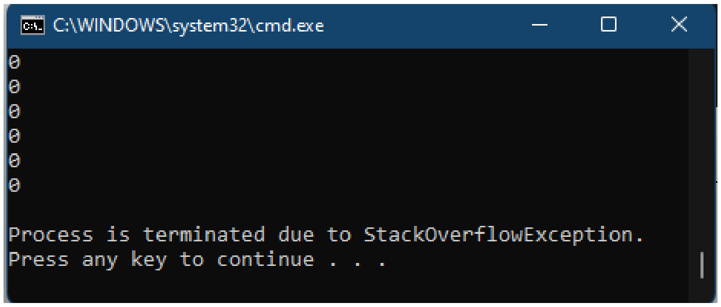


```
Unhandled Exception: System.ArrayTypeMismatchException:
Attempted to access an element as a type incompatible wi
th the array.
   at random.Program.Main(String[] args) in C:\Users\vih
ar\Desktop\random\testcode\random\random\Program.cs:line
 16
Press any key to continue . . .
```

6. StackOverflowException :



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace random
{
    0 references
    internal class Program
    {
        2 references
        static void Postinc(int x)
        {
            Console.WriteLine(x);
            Postinc(x++);
        }
        //StackOverFlowException
        0 references
        static void Main(string[] args)
        {
            Postinc(0);
        }
    }
}
```



```
0
0
0
0
0
0

Process is terminated due to StackOverflowException.
Press any key to continue . . .
```