DAY 12 ASSIGNMENT
BY
ANDE MANOHAR
8TH FEB 2022

Q1.What is exception handling and why we need exception handling.

- Exception is done ensure that our application will not crash.
- will not display any technical details to make sure and it handle errors gracefully and display friendly messages.

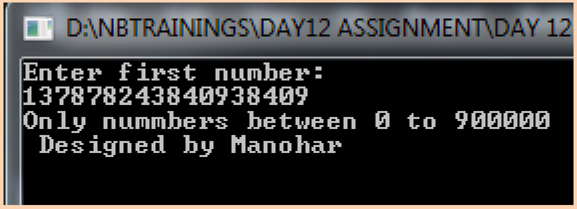
Q2Write a simple division program and handle three exceptions discussed in the class also add super exception at the last.

Code:

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

namespace DAY_12_PROJECT1
{
    internal class Program
    {
        static void Main(string[] args)
        {
            try

            {
                int a, b, c;
                Console.WriteLine("Enter first number:");
                a = Convert.ToInt32(Console.ReadLine());
                Console.WriteLine("Enter second number:");
                b = Convert.ToInt32(Console.ReadLine());
                c = a / b;
                Console.WriteLine("division of two numbers is {0}", c);
                Console.ReadLine();

            }
            catch (OverflowException)// This is for overflow exception

            {
                Console.WriteLine("Only nummbers between 0 to 900000");
            }
            catch (FormatException)
            {
                Console.WriteLine("Only numbers are allowed");

            }
            catch (DivideByZeroException)
            {
                Console.WriteLine(" Can not divide with zero");

            }
            catch (Exception)
            {
                Console.WriteLine("some error occured please contact admin");
            }
            finally
            {
                Console.WriteLine(" Designed by Manohar");
                Console.ReadLine();
            }
        }
    }
}
```

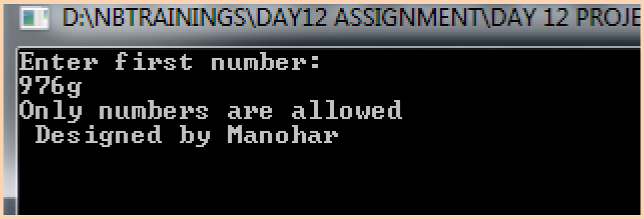
Output:

This is for overflow Exception



D:\NBTRAININGS\DAY12 ASSIGNMENT\DAY 12

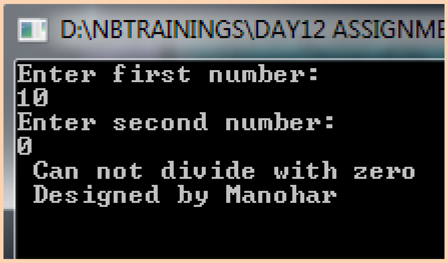
```
Enter first number:
137878243840938409
Only nummbers between 0 to 900000
 Designed by Manohar
```



D:\NBTRAININGS\DAY12 ASSIGNMENT\DAY 12 PROJE

```
Enter first number:
976g
Only numbers are allowed
 Designed by Manohar
```

This is for Format Exception



D:\NBTRAININGS\DAY12 ASSIGNME

```
Enter first number:
10
Enter second number:
0
 Can not divide with zero
 Designed by Manohar
```

Dividebyzero Exception

Q6.Write difference between compilation error and Run time error.

Compilation error	Run time error
1.Complier errors get when we want to Compile the code. 2.Errors that occur when we violet the rules of writing syntax is called Compiler errors. 3.All these errors are detected most frequent compile. 3. This are easy to detect and easy to fix . • Missing parenthesis({) • Printing the value of variable without declaring it.	1. Run time errors occur during program execution after successful compilation are called Run - time error. 2.These types of error are hard to find as the compiler doesn't point to the line at which the error occur. 3.This are hard to fix.

Q4.Wha is the use of "Finally " block illustrate with an example

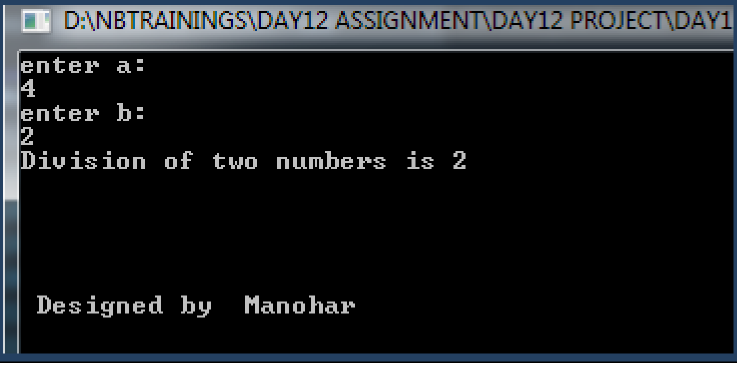
Statements inside the finally block will be execute irrespective of Exception occurs or not.

Uses:

- To display any common messages.
- To close any opened database or any files connections close it in finally block

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

namespace DAY12_PROJECT
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int a, b, c;
            try
            {
                Console.WriteLine("enter a:");
                a = Convert.ToInt32(Console.ReadLine());
                Console.WriteLine("enter b:");
                b = Convert.ToInt32(Console.ReadLine());
                c = a / b;
                Console.WriteLine("Division of two numbers is {0}", c);

            }
            catch (OverflowException)
            {
                Console.WriteLine("enter numbers between 0 o 600000");

            }
            catch (DivideByZeroException)
            {
                Console.WriteLine("Number cant be divided by zero");

            }
            catch (FormatException)
            {
                Console.WriteLine("Enter numbers only");

            }
            catch (Exception)
            {
                Console.WriteLine("Something went wrong, Manohar");

            }
            finally
            {
                Console.WriteLine("\n \n \n \n \n Designed by  Manohar");
                Console.ReadLine();



            }
        }
    }
}
```

Ouput:



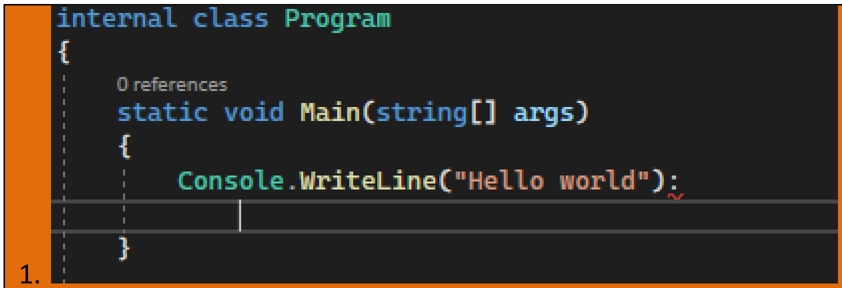
D:\NBTRAININGS\DAY12 ASSIGNMENT\DAY12 PROJECT\DAY1

```
enter a:
4
enter b:
2
Division of two numbers is 2




 Designed by  Manohar
```

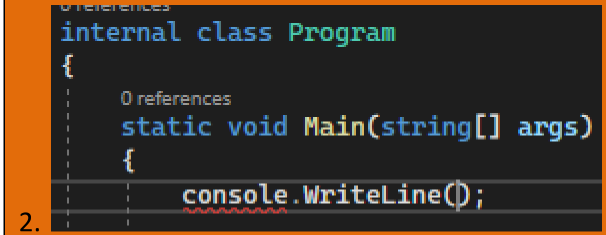
Q5.Write 5 points about Exception handling.

1.Exception handling is done to handle errors and ensure errors gracefully will no crash.
2.A single try block can have multiple catch blocks.
3.Always write a general Exception at the last.
4.Statements written inside finally block will executed irrespective of Exceptions occur or not.
5.The genral syntax for Exception is Try ,catch, finally.

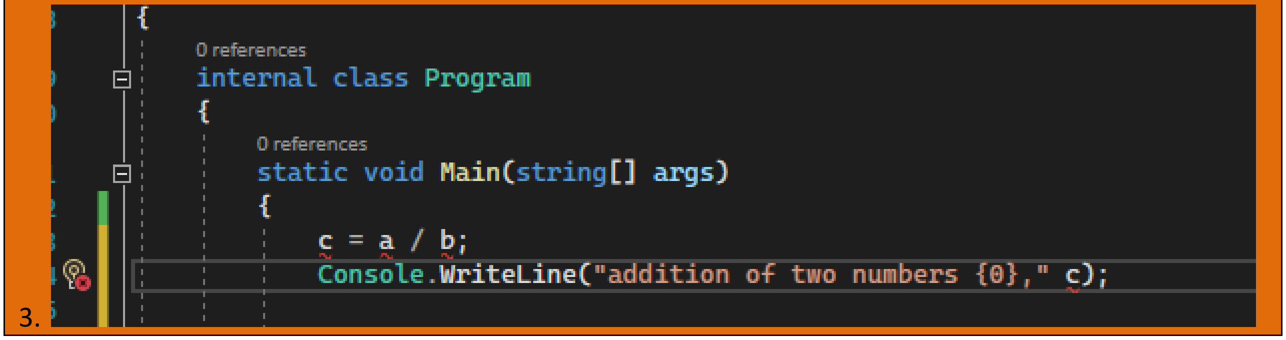
Q.7 Write 6 compilation errors in c#

1. 

```
internal class Program
{
    0 references
    static void Main(string[] args)
    {
        Console.WriteLine("Hello world"):

    }
```

2. 

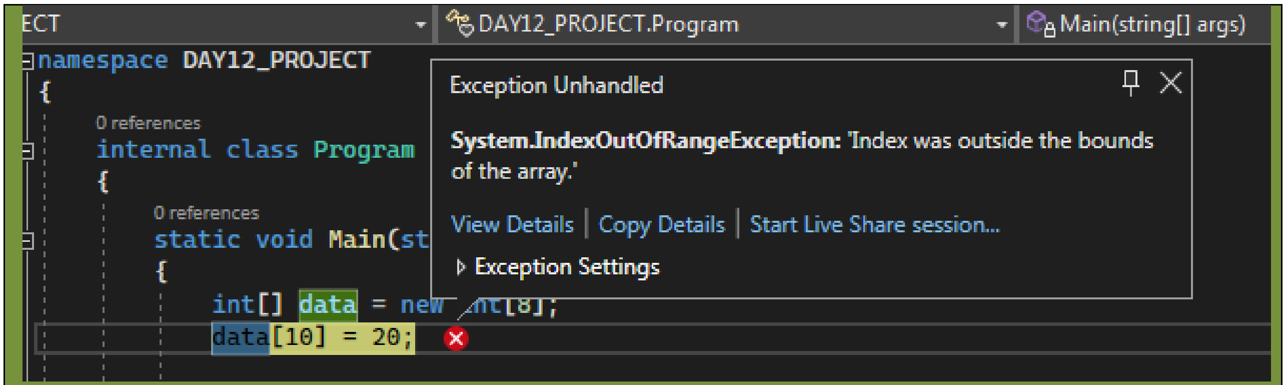
```
0 references
internal class Program
{
    0 references
    static void Main(string[] args)
    {
        console.WriteLine();
```

3. 

```
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            c = a / b;
            Console.WriteLine("addition of two numbers {0}," c);
```

Q8 Write any 6 run time errors

1.
```
namespace DAY12_PROJECT
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int[] data = new int[8];
            data[10] = 20;

        }
    }
}
```



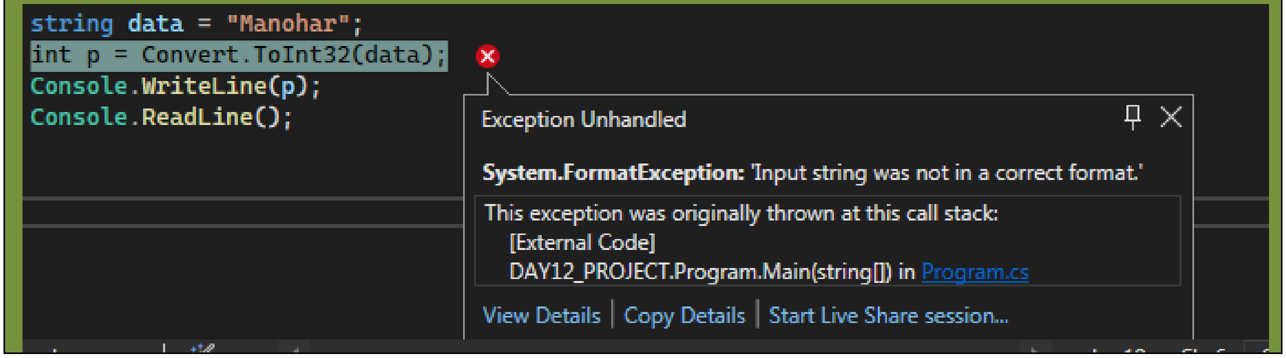
```
namespace DAY12_PROJECT
{
    0 references
    internal class Program
    {
        0 references
        static void Main(st
        {
            int[] data = new int[8];
            data[10] = 20;
```

Exception Unhandled

System.IndexOutOfRangeException: 'Index was outside the bounds of the array.'

RUNTIME ERROR 2.

```
internal class Program
    {
        static void Main(string[] args)
        {
            string data = "Manohar";
            int p = Convert.ToInt32(data);
            Console.WriteLine(p);
            Console.ReadLine();

        }
    }
}
```



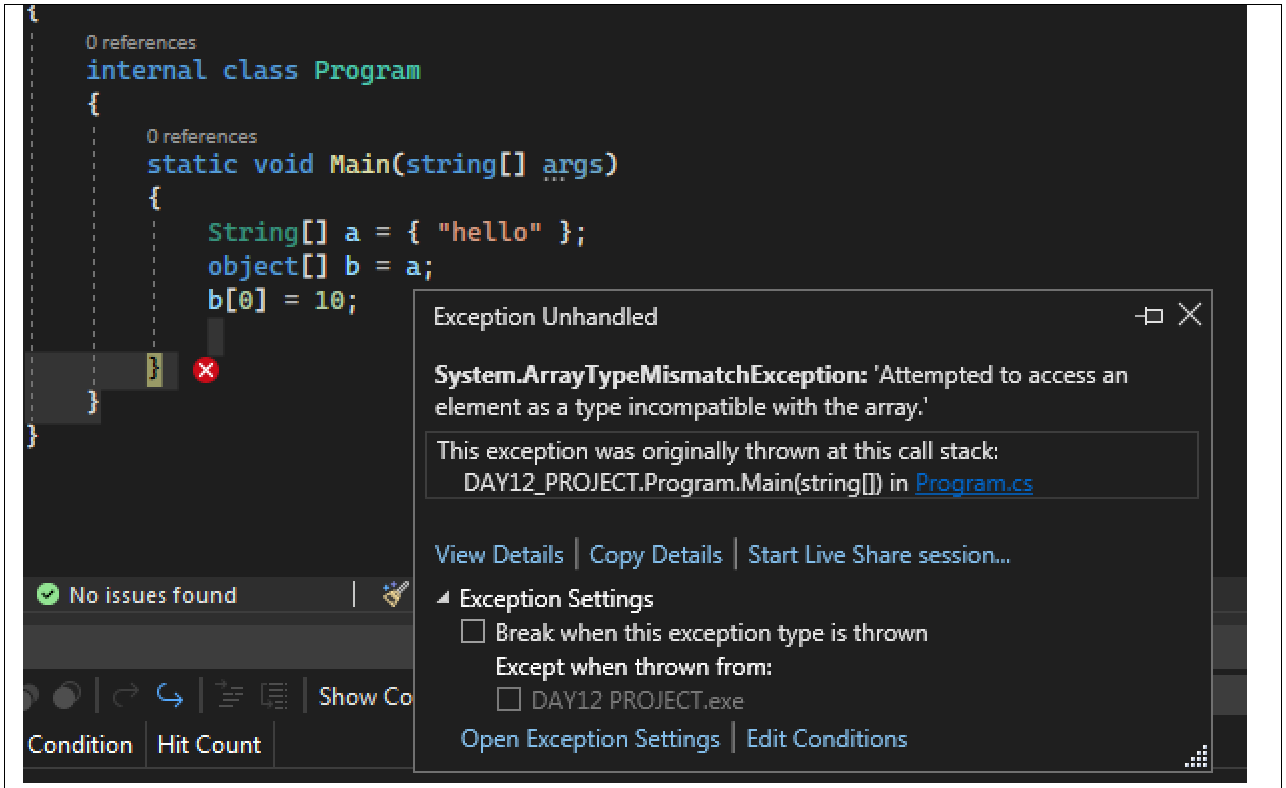
```
string data = "Manohar";
int p = Convert.ToInt32(data);
Console.WriteLine(p);
Console.ReadLine();
```

Exception Unhandled

System.FormatException: 'Input string was not in a correct format.'

This exception was originally thrown at this call stack:
    [External Code]
    DAY12_PROJECT.Program.Main(string[]) in Program.cs

3.



```
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            String[] a = { "hello" };
            object[] b = a;
            b[0] = 10;

        }
    }
}
```

Exception Unhandled

System.ArrayTypeMismatchException: 'Attempted to access an element as a type incompatible with the array.'

This exception was originally thrown at this call stack:
    DAY12_PROJECT.Program.Main(string[]) in Program.cs

View Details | Copy Details | Start Live Share session...

Exception Settings

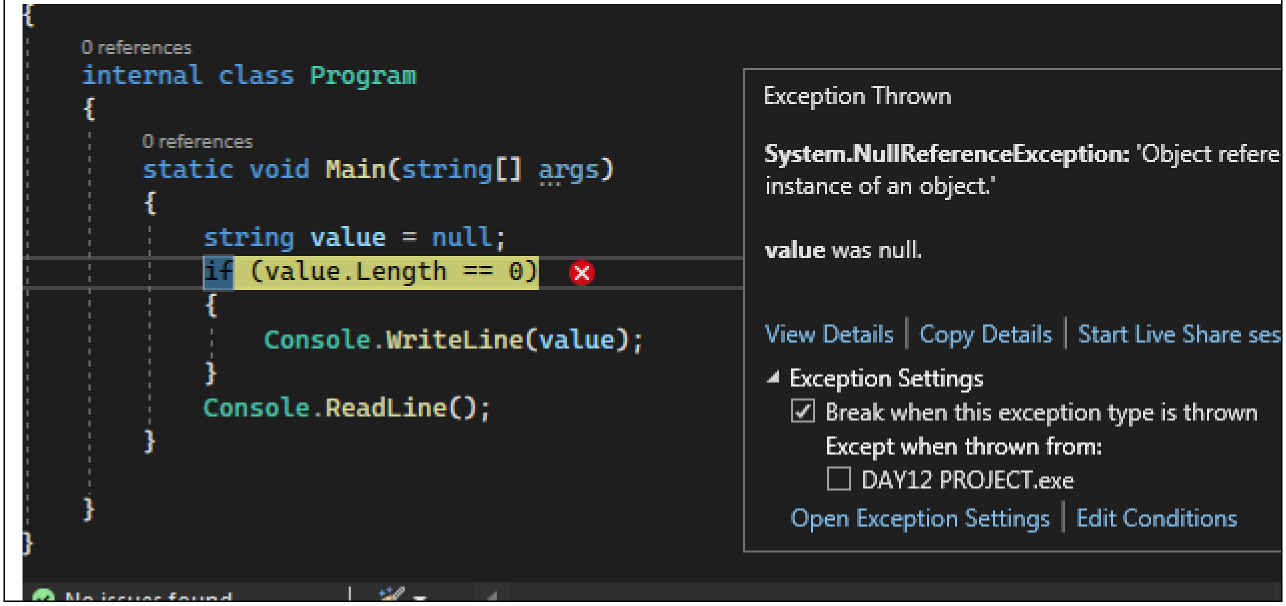
□ Break when this exception type is thrown

Except when thrown from:

□ DAY12 PROJECT.exe

Open Exception Settings | Edit Conditions

4.



```
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            string value = null;
            if (value.Length == 0)
            {
                Console.WriteLine(value);
            }
            Console.ReadLine();
        }

    }
}
```

Exception Thrown

System.NullReferenceException: 'Object refere
instance of an object.'

value was null.

View Details | Copy Details | Start Live Share ses

Exception Settings

☑ Break when this exception type is thrown

Except when thrown from:

□ DAY12 PROJECT.exe

Open Exception Settings | Edit Conditions