DAY12 Assignment By Ram Charan

1. What is Exception Handling and why we need exception handling.

- It is done to ensure that our application should not be crashed
- It will not display any technical issues and to make sure that it handle errors gracefully and display friendly messages.
- The process of responding the occurrence of exceptions is known as Exception Handling.

2.Write a simple division program and handle three exceptions and also add super exception at last.

```
Code:
```

```
Console.WriteLine("enter b:");
  b = Convert.ToInt32(Console.ReadLine());
  c = a / b;
  Console.WriteLine("Division of 2 numbers is {0}",c);
  Console.ReadLine();
catch (OverflowException) //It occurs when numbers are greater than range
  Console.WriteLine("enter numbers between -2147483648 to 2147483647");
  Console.ReadLine();
catch(DivideByZeroException)//If number is divided by 0
  Console.WriteLine("Number cant be divided by zero");
  Console.ReadLine();
catch(FormatException)//If we enter wrong input like alphabets
  Console.WriteLine("Enter numbers only");
  Console.ReadLine();
catch(Exception)//It is the super
  Console. WriteLine("Something went wrong, contact Rc");
  Console.ReadLine();
```

Output 1: Division Success

```
enter a:
8
enter b:
Division of 2 numbers is 2
```

Output2: Format Exception

C:\Users\91807\source\repos\Day12Project1\Day12Project1\bin\Debug\Day12Project1.exe

enter a:

4t

Enter numbers only

Output3:Overflow Exception

■ C:\Users\91807\source\repos\Day12Project1\Day12Project1\bin\Debug\Day12Project1.exe

enter a:

enter numbers between -2147483648 to 2147483647

Output4:Divide by Zero Exception

C:\Users\91807\source\repos\Day12Project1\Day12Project1\bin\Debug\Day12Project1.exe

enter a:

5

enter b:

0

Number cant be divided by zero

Output5: Super Exception C:\Users\91807\source\repos\Day12Project1\Day12Project1\bin\Debug\Day12Project1.exe enter a: pp Something went wrong, contact Rc

- 3.Research and write atleast 6 exceptions that occur in C# with sample code.
- a.)OutOfMemory Exception:

Code:

```
static void Main(string[] args)
{
    string name = new string('r',int.MaxValue);

    Console.ReadLine();
}

Exception Unhandled

System.OutOfMemoryException: 'Exception of type
```

'System.OutOfMemoryException' was thrown.'

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

Exception Settings



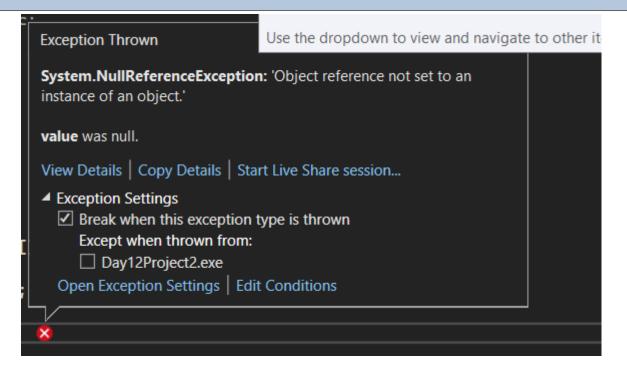
b.)NullReference Exception: Code:

```
static void Main(string[] args)
{
    string value = null;

    if(value.Length==0)
    {
        Console.WriteLine(value);
    }

    Console.ReadLine();
}
```

Output:

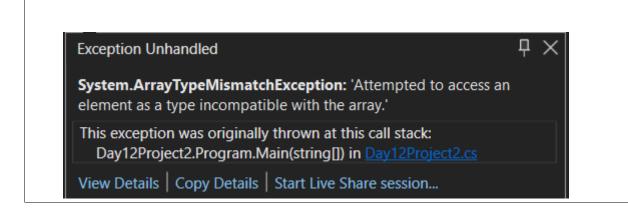


c.)InvalidCast Exception:

Code:

d.)ArrayTypeMismatch Exception:

Code:



4. What is use of "finally" block illustrate with an example.

• The statements in finally block are executed irrespective of exceptions.

Code:

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

```
namespace Day12Project1
  //Author:Rc
  /*Purpose: Division program and handle 3 exceptions and add super exception*/
  internal class Program
    static void Main(string[] args)
       //Variables declaration
       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 2 numbers is {0}", c);
       catch (OverflowException) //It occurs when numbers are greater than range
         Console.WriteLine("enter numbers between -2147483648 to 2147483647");
       catch (DivideByZeroException)//If number is divided by 0
         Console.WriteLine("Number cant be divided by zero");
       catch (FormatException)//If we enter wrong input like alphabets
         Console.WriteLine("Enter numbers only");
       catch (Exception)//It is the super exception
         Console.WriteLine("Something went wrong, contact Rc");
       finally
         Console.WriteLine("\n \n \n \n Designed by Rc");
         Console.ReadLine();
       }
```

```
Output1:

C:\Users\91807\source\repos\Day12Project3\Day12Project3\bin\Debug\Day12Project3.exe
enter a:
6
enter b:
3
Division of 2 numbers is 2

Designed by Rc
```

Output2:

```
C:\Users\91807\source\repos\Day12Project3\Day12Project3\bin\Debug\Day12Project3.exe

enter a:
3
enter b:
pp
Enter numbers only

Designed by Rc
```

5. Write about exception handling.

- Exception handling is done to gracefully handle errors.
- A single try block can have multiple catch blocks.
- Always write general exception at last.
- Statements inside final block will executed irrespective of errors.
- Syntax:

```
try
{
```

```
catch (Exception1)
{

catch(Exception2)
{

catch(Exception)
{

finally
{
}
```

6. Write differences between Compilation error and Runtime error.

Parameters	Compile time errors	Run time errors
Detection	 Compilers can easily detect compile time errors during development of code. 	 Compiler doesnot detect runtime errors during the code execution.
Reference	 These errors generally refers to errors that corresponds to syntax. 	 It refers to the error that occur during code execution during runtime
Fixation	We can easily fix these errors during development of code.	 A compiler cannot identify run-time error. But we can fix it after execution of code.

- 7. Write 6 compilation errors with small code snippets.
 - 1. Code: Semicolon error

```
Console.WriteLine("Division of 2 numbers is {0}", c)
```

2. Code: Use of unassigned variable

```
c = a / b;
Console.WriteLine("Division of 2 numbers is {0}", c);
```

3. Code: Mistake in code

```
Console.Writeline("enter a:");
```

4. Code: Forget to implement namespace

```
Console.WriteLine("enter a:");
a = Convert.ToInt32(Console.ReadLine());
Console.WriteLine("enter b:");
b = Convert.ToInt32(Console.ReadLine());
```

5. Code:



6. Code:

```
oreferences

static Main(string[] args)

{
Program.Program(string[] args)

int a, CS1520: Method must have a return type
try
{
Show potential fixes (Alt+Enter or Ctrl+.)
```

8. Write 6 runtime errors with small snippets.

1. IndexOutOfRange

```
Exception Unhandled

O references
internal class Program
{

O references
static void Main(s
{

int[] data = new int[3];

data[7] = 66;
```

2. FormatException

3. DivideByZero

```
Oreferences
internal class Program

{
    Oreferences
    static void Mai
    {
        int a, b, c
        Console.Wri
        a = Convert
        Console.Wri
        b = Convert. Int32(Console.ReadLine());

        c = a / b;
        Console.WriteLine(c);
```

4. Overflow O references internal class Program Oreferences static void Main(string[] args) int a, b, c; Console.WriteLine("enter a:"); a = Convert.ToInt32(Console.ReadLine()); Console.WriteLine("enter b:"); P × b = Convert.ToInt32(Console.ReadLine()); **Exception Unhandled** System.OverflowException: 'Value was either too large or too small Console.WriteLine(c); for an Int32." This exception was originally thrown at this call stack: Console.ReadLine(); [External Code] Day12Project2.Program.Main(string[]) in Day12Project2.cs **∛** ▼ C:\Users\91807\source\repos\Day12Project2\Day12Project2\bin\Debug\Day12Project2.exe enter a: 66666666666666666 5. ArrayTypeMisMatch 12Project2.cs 🕫 🗙 → %Day12Project2.Program Day12Project2 Exception Unhandled □namespace Day12Project2 **{**▶| System.ArrayTypeMismatchException: 'Attempted to access an element as a type incompatible with the array. internal class Program This exception was originally thrown at this call stack: Day12Project2.Program.Main(string[]) in Day12Project2.cs View Details | Copy Details | Start Live Share session... static void Main(string[■ Exception Settings ☐ Break when this exception type is thrown string[] a = { "Hi" object[] b = a; Except when thrown from: b[0] = 3;Open Exception Settings | Edit Conditions 19 🖗 Console.ReadLine();

6. NullReference error:

```
Oreferences
internal class Program

{
    Oreferences
    static void Main(string[] args)
    {
        string value = null;
        Console.WriteLine(v
        }
        Console.WriteLine(v
         }
        Console.ReadLine();
        View Details | Copy Details | Start Live Share session...

In the static void Main(string[] args)

{
        string value = null;
        System.NullReferenceException: 'Object reference not set to an instance of an object.'
        value was null.

No issues found

In 16
```

END OF DAY