**Part 43 - C# Tutorial - Exception Handling abuse**

**Exceptions are unforeseen errors that occur when a program is running**. For example, when an application is executing a query, the database connection is lost. Exception handling is generally used to handle these scenarios.   
  
  
But many a times I have seen programmers using exception handling to implement **programming logic which is bad, and this is called as exception handling abuse**.

**Program using exception handling to implement logical flow:**  
using System;  
public class ExceptionHandlingAbuse  
{  
    public static void Main()  
    {  
        try  
        {  
            Console.WriteLine("Please enter Numerator");  
            int Numerator = Convert.ToInt32(Console.ReadLine());  
  
  
            Console.WriteLine("Please enter Denominator");  
            //Convert.ToInt32() can throw FormatException, if the entered value  
            //cannot be converted to integer. So use int.TryParse() instead  
            int Denominator = Convert.ToInt32(Console.ReadLine());  
  
  
            int Result = Numerator / Denominator;  
  
  
            Console.WriteLine("Result = {0}", Result);  
        }  
        catch (FormatException)  
        {  
            Console.WriteLine("Only numbers are allowed!");  
        }  
        catch (OverflowException)  
        {  
            Console.WriteLine("Only numbers between {0} & {1} are allowed",  
                Int32.MinValue, Int32.MaxValue);  
  
  
        }  
        catch (DivideByZeroException)  
        {  
            Console.WriteLine("Denominator cannot be zero");  
        }  
        catch (Exception ex)  
        {  
            Console.WriteLine(ex.Message);  
        }  
    }  
}  
[**In Part 44, we will learn how to prevent exception handling abuse. please click here to watch it now.**](http://csharp-video-tutorials.blogspot.com/2012/06/part-44-c-tutorial-preventing-exception.html)