using System;

namespace ExceptionHandlingDemos

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Enter Numerator");

int num = Byte.Parse(Console.ReadLine());

Console.WriteLine("Enter Denominator");

int den = Byte.Parse(Console.ReadLine());

int res = num / den;

Console.WriteLine("Result is "+ res);

}

}

}

using System;

namespace ExceptionHandlingDemos

{

class Program

{

static void Main(string[] args)

{

int res = 0;

try {

Console.WriteLine("Enter Numerator");

int num = Byte.Parse(Console.ReadLine());

Console.WriteLine("Enter Denominator");

int den = Byte.Parse(Console.ReadLine());

res= num / den;

Console.WriteLine("Result is " + res);

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

}

Console.WriteLine($"Result is {res}");

}

}

}

using System;

namespace ExceptionHandlingDemos

{

class Program

{

static void Main(string[] args)

{

int res = 0;

try {

int[] nums = new int[10];

Console.WriteLine("Enter Numerator");

int num = Byte.Parse(Console.ReadLine());

Console.WriteLine("Enter Denominator");

int den = Byte.Parse(Console.ReadLine());

res= num / den;

Console.WriteLine("Result is " + res);

nums[20] = 100;

}

catch(DivideByZeroException ex)

{

Console.WriteLine("Cannot divide by 0");

}

catch (FormatException ex)

{

Console.WriteLine("Please enter number only");

}

catch(OverflowException ex)

{

Console.WriteLine("Please enter numbers within a range");

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

}

Console.WriteLine($"Result is {res}");

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ExceptionHandlingDemos

{

class Program2

{

static void Main()

{

try

{

Console.WriteLine("Enter Name");

string name = Console.ReadLine();

if (name.Length < 10)

throw new CustomException("Min 10 characters needed");

Console.WriteLine("Enter Age");

int age = Byte.Parse(Console.ReadLine());

if (age < 20 || age > 40)

throw new CustomException("Range for Age is 20 to 40");

}

catch(FormatException ex)

{

Console.WriteLine(ex.Message);

}

catch(OverflowException ex)

{

Console.WriteLine(ex.Message);

}

catch(CustomException ex)

{

Console.WriteLine(ex.Message);

}

finally

{

Console.WriteLine("I m in Finally block");

}

}

}

}

Extenstion Methods

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ExceptionHandlingDemos

{

class Practice

{

public void Func1()

{

Console.WriteLine("Inside Func1");

}

public void Func2()

{

Console.WriteLine("Inside Func2");

}

public void Func3()

{

Console.WriteLine("Inside Func3");

}

}

// This is how we add extension Methods

static class EnhancedPractice

{

public static void Func4(this Practice practice)

{

Console.WriteLine("Inside Func4");

}

public static void Func5(this Practice practice)

{

Console.WriteLine("Inside Func5");

}

}

static class PracticeDemo

{

public static string ReturnFirstCharacter(this String str)

{

return str.Substring(1, 1);

}

}

class ExtensionMethodDemo

{

static void Main()

{

Practice practice = new Practice();

practice.Func1();

practice.Func4();

Console.WriteLine("Deepak".ReturnFirstCharacter());

}

}

}