**Different Types of Exceptions – Examples**

1. DivideByZeroException

using System;

class Program

{

static void Main()

{

try

{

int x = 10;

int y = 0;

int result = x / y; }

catch (DivideByZeroException ex)

{

Console.WriteLine("Error: " + ex.Message);

}

}

}

1. NullReferenceException

using System;

class Program

{

static void Main()

{

try

{

string str = null;

Console.WriteLine(str.Length); }

catch (NullReferenceException ex)

{

Console.WriteLine("Error: " + ex.Message);

}

}

}

1. IndexOutOfRangeException

using System;

class Program

{

static void Main()

{

try

{

int[] arr = { 10, 20, 30, 40, 50 };

// Loop condition will go out of bounds when i equals arr.Length

for (int i = 0; i <= arr.Length; i++)

{

Console.WriteLine(arr[i]); }

}

catch (IndexOutOfRangeException ex)

{

Console.WriteLine("Error: " + ex.Message);

}

Console.WriteLine("End of the program");

}

}

1. FormatException

using System;

class Program

{

static void Main()

{

try

{

int x = 10;

int y = a;

int z = x / y;

}

catch (FormatException ex)

{

Console.WriteLine("Error: " + ex.Message);

}

}

}

1. InvalidOperationException

using System;

using System.Collections.Generic;

class Program

{

static void Main()

{

try

{

List<int> numbers = new List<int> { 1, 2, 3, 4, 5 };

foreach (var number in numbers)

{

if (number == 3)

{

numbers.Remove(number); // This will throw InvalidOperationException

}

}

}

catch (InvalidOperationException ex)

{

Console.WriteLine("Error: " + ex.Message);

}

Console.WriteLine("End of the program");

}

}

1. ArgumentNullException

using System;

class Program

{

static void Main()

{

try

{

int age = -5; // Invalid age

// Check if the age is out of valid range

if (age < 0 || age > 120)

{

throw new ArgumentOutOfRangeException(nameof(age), "Age must be between 0 and 120.");

}

Console.WriteLine("Age is valid: " + age);

}

catch (ArgumentOutOfRangeException ex)

{

Console.WriteLine("Error: " + ex.Message);

}

}

}

1. InvalidCastException

using System;

class Program

{

static void Main()

{

try

{

object obj = "Hello, world!"; // obj is a string

int number = (int)obj; // Attempting to cast string to int

}

catch (InvalidCastException ex)

{

Console.WriteLine("Error: " + ex.Message);

}

}

}

Try with Multiple Catch Block Examples

Example 1

using System;

class Program

{

static void Main()

{

try

{

int[] numbers = { 1, 2, 3 };

int index = 5;

// Attempt to access an invalid index

int value = numbers[index];

// Attempt to parse an invalid number format

int result = int.Parse("InvalidNumber");

}

catch (IndexOutOfRangeException ex)

{

Console.WriteLine("IndexOutOfRangeException caught: " + ex.Message);

}

catch (FormatException ex)

{

Console.WriteLine("FormatException caught: " + ex.Message);

}

catch (Exception ex)

{

// General catch block for any other exceptions

Console.WriteLine("General exception caught: " + ex.Message);

}

}

}

Example 2

using System;

class Program

{

static void Main()

{

try

{

Console.WriteLine("Enter the First number:");

int x = int.Parse(Console.ReadLine()); // Corrected `int.parse` to `int.Parse`

Console.WriteLine("Enter the Second number:");

int y = int.Parse(Console.ReadLine()); // Corrected `int.parse` to `int.Parse`

int z = x / y;

Console.WriteLine("The result is: " + z);

Console.WriteLine("End of the program");

}

catch (DivideByZeroException ex)

{

Console.WriteLine("DivideByZeroException caught: " + ex.Message);

}

catch (FormatException ex)

{

Console.WriteLine("FormatException caught: " + ex.Message);

}

catch (IndexOutOfRangeException ex) // Corrected `IndexoutofrangeException` to `IndexOutOfRangeException`

{

Console.WriteLine("IndexOutOfRangeException caught: " + ex.Message);

}

catch (Exception ex)

{

// General catch block for any other exceptions

Console.WriteLine("General exception caught: " + ex.Message);

}

}

}

Throw and Throw ex Examples

Example 1

using System;

class Program

{

static void Main()

{

try

{

int x = 10;

int y = 0;

int z = x / y; // This will cause a DivideByZeroException

}

catch (Exception ex)

{

Console.WriteLine("Caught: " + ex.Message);

// Rethrow the same exception to propagate it

throw;

}

}

}

Example 2

using System;

class Program

{

static void Main()

{

try

{

try

{

// This will cause a FormatException

int number = int.Parse("invalid");

}

catch (FormatException ex)

{

// Throw the caught exception again

throw ex;

}

}

catch (FormatException ex)

{

Console.WriteLine("Caught in outer catch: " + ex.Message);

}

}

}