

# Excepciones



# Temas a tratar:

6.1- Manejo de Excepciones.

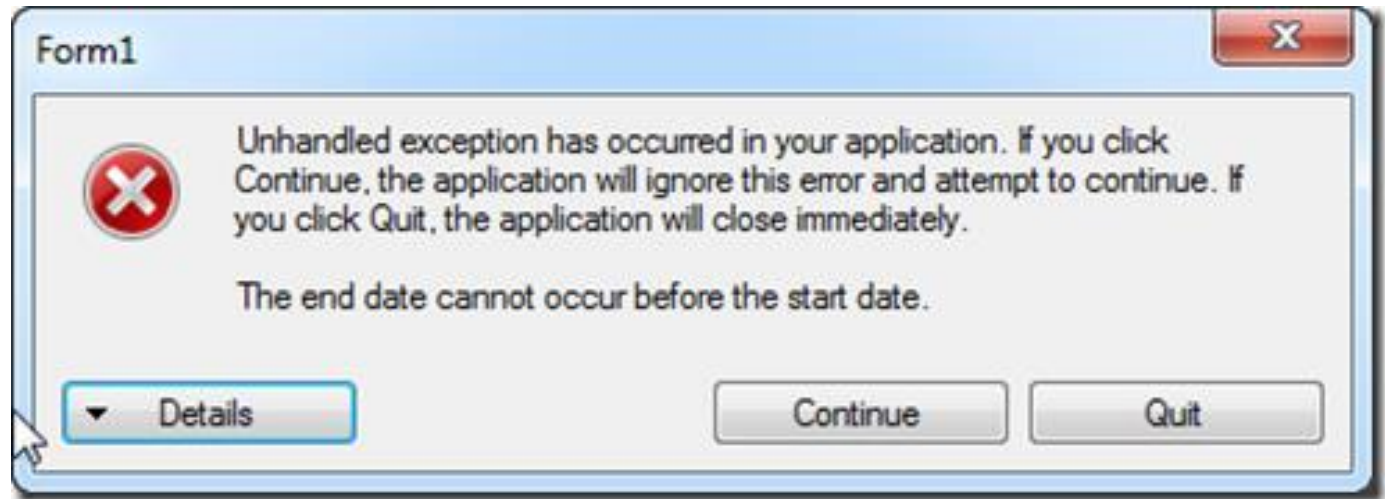
6.2- Tipo de Excepciones.

6.3- Lista de Excepciones.

6.4- Try, catch y finally.

## 6.1.- Manejo de excepciones

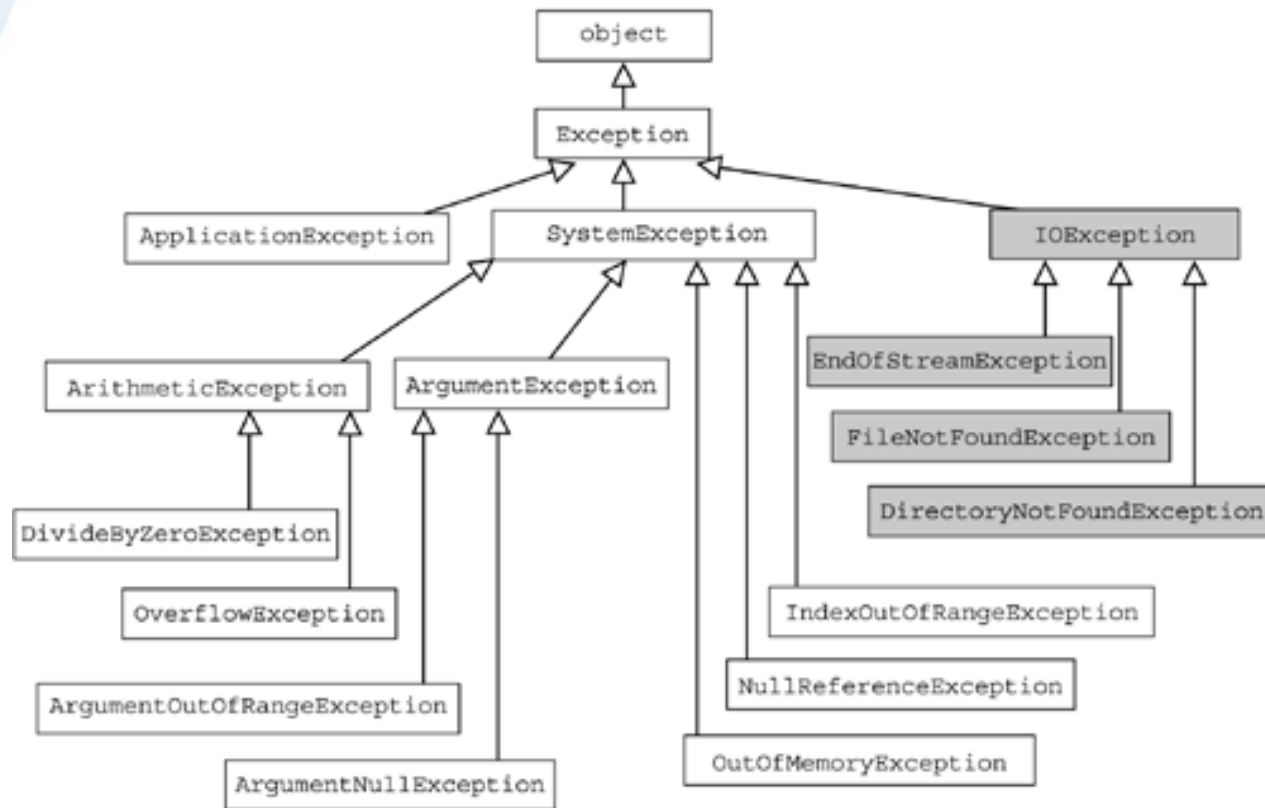
El **manejo de excepciones** es una técnica de programación que permite al programador controlar los **errores** ocasionados durante la ejecución de un programa informático.



Cuando ocurre cierto tipo de error, el sistema reacciona ejecutando un fragmento de código que resuelve la situación, por ejemplo retornando un mensaje de error o devolviendo un valor por defecto.

## 6.2.- Tipo de excepciones

Las clases derivadas de **Exception** representan estas excepciones. Esta clase identifica el tipo de excepción y contiene las propiedades que albergan detalles acerca de ésta.



## 6.3.- Lista de excepciones

Exception	When Occured
System.ArithmeticException	A base class for exceptions that occur during arithmetic operations, such as System.DivideByZeroException and System.OverflowException
System.ArrayTypeMismatchException	Thrown when a store into an array fails because the actual type of the stored element is incompatible with the actual type of the array.
System.DivideByZeroException	Thrown when an attempt to divide an integral value by zero occurs.
System.IndexOutOfRangeException	Thrown when an attempt to index an array via an index that is less than zero or outside the bounds of the array.
System.InvalidCastException	Thrown when an explicit conversion from a base type or interface to a derived type fails at run time.
System.NullReferenceException	Thrown when a null reference is used in a way that causes the reference object to be required.
System.OutOfMemoryException	Thrown when an attempt to allocate memory ( Via new ) fails.
System.OverflowException	Thrown when an arithmetic operation in a checked context overflows.
System.StackOverflowException	Thrown when the execution stack is exhausted by having too many pending method calls; typically indicative of very deep or unbounded recursion.
System.TypeInitializationException	Thrown when a static constructor throws an exception, and no catch clauses exists to catch it.



## 6.4.- Try, Catch, finally

Las estructuras try, catch y finally permiten la manipulación de excepciones logrando ejecutar un camino alternativo o mostrar algún mensaje en caso de ocurrir un error en tiempo de ejecución.

```
static void Main() {  
    int n;  
  
    try { // Do not initialize this variable here.  
        n = 123;  
    } catch {  
        // Error: Use of unassigned local variable 'n'.  
        Console.Write(n); }  
}
```