

Excepciones comunes en Java

EXCEPCIÓN	DESCRIPCIÓN Y USO
<i>IndexOutOfBoundsException</i>	Lanzada cuando se intenta acceder a un Array/String/Vector con un valor de índice inválido (sea éste negativo o superior a la longitud del Array/String/Vector)
<i>ClassCastException</i>	Lanzada cuando intentamos convertir una referencia a variable a un tipo que falla la prueba de cast
<i>NumberFormatException</i>	Esta excepción se produce cuando se trata de convertir un String a un número.
<i>ArithmeticException</i>	Se produce cuando una condición aritmética excepcional, se ha producido.
<i>InputMismatchException</i>	Se produce cuando intentamos introducir caracteres en una variable numérica
<i>NullPointerException</i>	Lanzada cuando intentamos acceder a un objeto con una variable de referencia cuyo valor actual es NULL.
<i>NumberFormatException</i>	Lanzada cuando un método que convierte un String a un número recibe un String que no puede ser convertido.
<i>AssertionError</i>	Lanzada cuando una sentencia Boolean retorna el valor falso después de ser evaluada.
<i>IOException</i>	Lanza la excepción cuando ocurre un fallo o es interrumpida la operación en curso. 2 comunes subtipos de excepción de IOException son EOFException y FileNotFoundException
<i>FileNotFoundException</i>	Lanza la excepción cuando al abrir un archivo no es encontrado.
<i>SQLException</i>	Lanza la excepción cuando ocurre un error en la Base de Datos
<i>InterruptedException</i>	Lanza la excepción cuando el Hilo es interrumpido
<i>NoSuchMethodException</i>	Lanza la excepción cuando se llama a un método y este no es encontrado.
<i>IllegalArgumentException</i>	Lanzada cuando un método recibe un argumento formateado de manera diferente a lo que el método esperaba.
<i>IllegalStateException</i>	Lanzada cuando el estado del entorno no coincide con la operación que se intenta ejecutar
<i>NullPointerException</i>	Lanzada cuando intentamos acceder a un objeto con una variable de referencia cuyo valor actual es NULL
<i>ExceptionInInitializerError</i>	Lanzada cuando intentamos inicializar una variable estática o un bloque de inicialización
<i>StackOverflowError</i>	Lanzada cuando un método es invocado demasiadas veces, por ejemplo, recursivamente
<i>NoClassDefFoundError</i>	Lanzada cuando no se puede ubicar la clase que se necesita, por un error en la línea de comandos, problemas de classpath o un archivo class perdido
<i>ClassNotFoundException</i>	Lanzada cuando intentamos ejecutar un proyecto y, por ejemplo, la clase que contiene la función "main" no ha sido añadida al mismo o no es encontrada.
<i>RuntimeException</i>	Son un conjunto de excepciones que pueden tener lugar durante el proceso de ejecución de un programa sobre la JVM. El usuario no tiene que prestar atención al hecho de capturarlas

<i>NegativeArraySizeException</i>	Lanzada cuando intentamos crear un array con longitud negativa
<i>InputMismatchException</i>	Lanzada por varios métodos de la librería Scanner de Java cuando están recorriendo una entrada en busca de un dato y este no existe en la entrada.
OutOfMemoryException	El intento de crear un objeto con el operador <i>new</i> ha fallado por falta de memoria. Y siempre tendría que haber memoria suficiente porque el <i>garbagecollector</i> se encarga de proporcionarla al ir liberando objetos que no se usan y devolviendo memoria al sistema.
UnsatisfiedLinkException	Se hizo el intento de acceder a un método nativo que no existe.
InternalException	Este error se reserva para eventos que no deberían ocurrir. Por definición, el usuario nunca debería ver este error y esta excepción no debería lanzarse.
IncompatibleClassChangeException	El intento de cambiar una clase afectada por referencias en otros objetos, específicamente cuando esos objetos todavía no han sido recompilados.

SINTAXIS DE LAS EXCEPCIONES

```
try{

    //Aquí está el código que puede lanzar excepciones

    ...

} catch (ClaseDeLaExcepcionobjetoException) {

    ...

    //Aquí está el código que se ejecutará si se lanza una excepción del

        //tipo ClaseDeLaExcepción

    ...

}
```

EJEMPLO SINTAXIS

```
public class EjemploExcepciones {

    public static void main(String[] args) {

        if (args.length != 1)

            System.out.println("Debe introducir un entero a la entrada");

        enteroLeido = stringToInt(args[0]);

        System.out.println("Ha introducido el entero " + enteroLeido);

    }

    private static int stringToInt(String entero) {

        int resultado = Integer.parseInt(entero);

        return resultado;

    }

}
```

