 



Conceptos fundamentales de Java 6-2: Manejo de errores Actividades prácticas

# Objetivos de la lección:

* Descripción de distintos tipos de errores que pueden ocurrir y cómo se manejan en Java
* Descripción de excepciones utilizadas en Java
* Determinación de las excepciones que se arrojan para una clase base
* Escribir un código para manejar una excepción arrojada mediante el método de una clase base

# Vocabulario:

Identifique el término correspondiente a cada una de las definiciones a continuación.

|  |  |
| --- | --- |
| Catch | Palabra clave en Java que indica que el bloque siguiente de código maneja una excepción especificada. |
| Try/catch | Una excepción que se maneja de manera opcional |
| throws | Una excepción que se DEBE manejar. |
| Sintaxis | Indica que hay un problema con la interpretación de su programa. |
| try | Evita que el intérprete ejecute el resto del código hasta que éste encuentre una palabra catch. |
| unicodeError | Error que indica un problema con el formato de codificación. |
| Error lógica | Error que se produce mientras se ejecuta el programa, también denominado excepción. |
| Errores en la secuencia de comandos, problemas con las condiciones, error en la estructura de datos | Error que se produce como resultado de una lógica de programación incorrecta. |
| Try/catch | Bloque de código que maneja excepciones al tratar casos donde se arroja una excepción. |
| Excepciones | Errores que se producen durante el tiempo de ejecución y que pueden ser corregidos o manejados por el código. |

# Inténtelo/resuélvalo:

1. Describa la diferencia entre error de sintaxis, error lógico y excepción.

R// error de sintaxis: ocurre cuando la información que se mete al ordenador no se reconoce o tiene un formato incorrecto

Una información mal escrita o sin el formato adecuado puede deberse a una equivocación del programador, el usuario final o puede ser incluso un problema del propio ordenador.

Error lógico: son error que se producen mientras estamos haciendo uso de un programa ya copilado

Error de excepción: son errores que no son críticos y por lo tanto pueden ser tratados y continuar con la ejecución de la aplicación

1. ¿Cuál es el problema del siguiente código? Debe imprimir " Hello World! " cuatro veces a la pantalla.

String str = “Hello World”;

for(int i = 0; i < 4; i++);

{

System.out.println(str); str+= ”!”;

}

R// El punto y como en for (;) y las comillas (“”)

1. Describa una excepción que haya experimentado antes en su programa. Explique de qué manera se podría manejar con un bloque de código try/catch.

R// Tratar de convertir entero a String que no contenga valores numéricos, el try/catch se podría manejar de esta manera colocar el try {las introducciones

Luego el catch ([ la excepción 1]) y luego las instrucciones; el catch ([excepciones}), introducciones

Catch ([excepciones n}) ([introducciones n])

1. Escriba un segmento de código que tenga:
   1. Un error de sintaxis.
   2. Un error lógico.
   3. Una excepción.
2. ¿Cuál es la diferencia entre una excepción marcada y una excepción sin marcar?

Las excepciones comprobadas son aquellas excepciones que el compilador verifica para la ejecución sin problemas del programa. El compilador verifica si el programa puede manejar la excepción verificada o no. Si el código no puede manejar la excepción, se produce un error de compilación. Todas las subclases de la clase "Excepción", excepto la clase RuntimeException, son las excepciones comprobadas.

Las excepciones no marcadas son aquellas excepciones que no son verificadas por el compilador. El compilador compila el programa exitosamente incluso si las excepciones no son manejadas por el código. La clase "Error" y sus clases secundarias, la clase "Tiempo de ejecución" y sus subclases son las excepciones no verificadas.