Trabajo práctico Nro. 4

|  |  |  |
| --- | --- | --- |
| Logo Departamento 2.jpg | **Asignatura: Programación II** | |
|  | |
| **Cursado:** Segundo semestre | **Horas semanales**: |
|  | **Horas semestrales:**  *Cantidad estimada de horas*  *semestrales/anuales.* |
| **Carrera**: *Tecnicatura Universitaria en*  *Programación* | **Nivel (Año):**  1°  2°  3° |
| **Ciclo Lectivo: 2024** |

**Integrantes de la Cátedra:**

* **DOCENTES:**

|  |  |  |
| --- | --- | --- |
| **Nombre del Profesor** | **Periodo** | **Cantidad horas materia** |
| **Cinthia Rigoni** |  | 6 horas |

* 1. **¿Qué mostrará por pantalla el siguiente código?** public class EjemploExcepciones {

public static int devuelveNumero(int num) { try {

if (num % 2 == 0) {

throw new Exception(“Lanzando excepción”);

}

return 1;

} catch (Exception ex) { return 2;

} finally { return 3;

}

}

public static void main(String[] args) { System.out.println(devuelveNumero(1));

}

}

Respuesta:

El código mostrará por pantalla el número 3.

* 1. **¿Cuál es el nombre en Java de la clase que define las excepciones, y de la que debe heredar cualquier clase que queramos usar para representar una excepción?**

En Java, la clase que define las excepciones es `java.lang.Exception`.

Cualquier clase que queramos usar para representar una excepción debe heredar de esta clase o de alguna de sus subclases, como `RuntimeException` o `IOException`, dependiendo del tipo de excepción que queramos manejar.

* 1. **¿Cuál es el nombre en Java de la clase que representa las excepciones que se producen al invocar un método de un objeto cuyo valor es null?**

En Java, la clase que representa las excepciones que se producen al invocar un método de un objeto cuyo valor es `null` es `NullPointerException`.

* 1. **¿Cuál es e l nombre en Java de la clase que representa las excepciones que se producen al obtener un comportamiento anómalo en la entrada/salida de información?**

La clase en Java que representa las excepciones que se producen al obtener un comportamiento anómalo en la entrada/salida de información es `IOException`.

* 1. **¿Qué nos aporta el método printStackTrace() sobre una excepción en Java?**

El método `printStackTrace()` en Java nos proporciona información detallada sobre una excepción, incluyendo el tipo de excepción, el mensaje asociado si lo hay, y la traza de la pila (stack trace) que muestra la secuencia de llamadas de métodos que condujeron a la excepción.

Esta traza de la pila es muy útil para depurar problemas, ya que muestra exactamente dónde ocurrió la excepción en el código.

A codificar!

1. **Escribe un programa que juegue con el usuario a adivinar un número. El ordenador debe generar un número entre 1 y 100, y el usuario tiene que intentar adivinarlo. Para ello, cada vez que el usuario introduce un valor, el ordenador debe decirle al usuario si el número que tiene que adivinar es mayor o menor que el que ha introducido el usuario. Cuando consiga adivinarlo, debe indicárselo e imprimir en pantalla el número de veces que el usuario ha intentado adivinar el número. Si el usuario introduce algo que no es un número, debe indicarlo en pantalla, y contarlo como un intento.**

Codigo resuelto en Java.

1. **Teniendo el siguiente código java:**

public class Excepciones2 {

static Scanner sc = new Scanner(System.in); public static void main(String[] args) {

double n; int posicion;

String cadena ;

double[] valores = {9.83, 4.5, -3.06, 0.06, 2.52, -11.3, 7.60, 3.00, -30.4, 105.2};

System.out.println("Contenido del array antes de modificar:"); for (int i = 0; i < valores.length; i++) {

System.out.printf("%.2f ", valores[i]);

}

System.out.print("\n\nIntroduce la posición del array a modificar: "); cadena = sc.nextLine();

posicion = Integer.parseInt(cadena);

System.out.print("\nIntroduce el nuevo valor de la posición " + posicion + ": "); n = sc.nextDouble();

valores[posicion] = n;

System.out.println("\nPosición a modificar " + posicion); System.out.println("Nuevo valor: " + n); System.out.println("Contenido del array modificado:"); for (int i = 0; i < valores.length; i++) {

System.out.printf("%.2f ", valores[i]);

}

}

}

**Capture y maneje todas las excepciones que puedan producirse.**

Codigo resuelto en Java.

1. **Escribe un programa java que permita calcular el área de un triángulo, la lectura de base y altura será por teclado, considere que el usuario podría ingresar de forma errada los datos como letras, espacios en blanco, simboles especiales.**

Codigo resuelto en Java.