

ÍNDICE

Bloque I - Captura y propagación de excepciones

1

Bloque I - Captura y propagación de excepciones

Actividad3.- Escribe un método que se encargue de pedir números al usuario hasta que se indique un número entre 1 y 5 (ambos incluidos). Cada número que incluya debe concatenarse en una variable de tipo String. Al finalizar la ejecución debe visualizar un mensaje similar a: "Números introducidos: 10, 20, 12, 56, 30".

Nota: Se debe validar que el usuario no inserte letras.

Actividad4.- Escribe un programa que lea por teclado una secuencia de números hasta que el usuario inserte letras. Al finalizar el programa, se debe visualizar el número más alto de todos los que se hayan introducido.

Puede ocurrir que un usuario no llegue a insertar ningún número. En este caso, se indicará con el mensaje correspondiente.

Nota: Se debe emplear un método para pedir los números.

Actividad5.- Con el fin de comprobar el funcionamiento de algunas excepciones, haz lo siguiente:

- Crea un método que reciba 2 parámetros: 1 array de enteros y un entero que represente el número de elementos a introducir
`public static void readNumbers(int[] numbers, int size).`
- Este método irá leyendo números desde el teclado y guardándolos en un array. ¿Qué situaciones se pueden producir y qué hacer en ellas?
 - Que se inserte una letra: informar al usuario.
 - Que se sobrepase la capacidad del *array*: Informar al usuario, finalizar y mostrar el contenido del *array*.
 - Que el array no haya sido creado: informar al usuario y finalizar.
- Desde el *main()* Crea un array de 5 números enteros y llama al método. Haz diferentes pruebas para forzar las dos primeras excepciones.

Actividad6.- Haz un programa con los siguientes requerimientos:

Queremos crear 2 alumnos. De cada alumno guardaremos el nombre, la edad y la altura.

Desde el `main()`, solicita los datos al usuario y comprueba que a la hora de introducir la edad y la altura no se inserten letras. Si es así, repite el proceso hasta que se insertan números. Utiliza métodos para solicitar los datos al usuario.

Crea 2 alumnos desde el `main()`, Visualiza la información de cada uno de ellos y luego indica qué alumno es mayor (investiga sobre la interfaz `Comparable` y utilízala convenientemente)

Actividad Opcional.- Diseña un método llamado `dividirEntreArray(int[] array, int num)` que reciba como parámetros un número entero y un array de enteros. El método mostrará por pantalla el resultado de la división entre el número recibido y cada uno de los elementos del *array*.

A continuación, escribe un programa que llame al método con el número 2 y un array con los elementos -2, -1, 0, 1 y 2.

- Reescribe el método para capturar la excepción derivada del intento de división entre 0, de forma que no se pare la ejecución del programa y se continúe con la división del resto de elementos del *array*.
- Crea un segundo método, a partir del anterior, con las sentencias de código necesarias para que la excepción no se llegue a producir.