

PRO – UD06.3

PROGRAMAS CON EXCEPCIONES

- 1) Escribe un programa que pida un double y un float. Si al pedir un numero el usuario inserta letras, se pedirá de nuevo ese dato. Una vez insertados, se visualizara el numero mas alto de los dos. Si son iguales, se visualizara cualquiera de los dos.

El control del error IOException se hará solamente dentro del método main. Cuando se produzca este error, se visualizara un mensaje y se cerrara el programa.

Se usara un método que pida un dato de tipo float. Dicho método controlara el error que se podría producir si se insertaran letras, visualizando en este caso un mensaje de error. El main sera el que se encargue de llamar de nuevo a este método si se insertaran letras.

De la misma forma se hará a la hora de pedir un dato de tipo double.

- 2) Escribe un programa que se encargue de pedir números hasta que se indique un numero comprendido entre 1 y 5 (ambos inclusive), visualizando al final del programa un mensaje como este: "Numero encontrado". Hay que validar que el usuario no inserte letras.
- 3) Escribe un programa que pida diez números con decimales. Si insertan letras, se pedirá de nuevo dicho numero. El programa debe visualizar el numero mas alto de entre todos los insertados. Cada vez que se produzca el error IOException se visualizara el mensaje correspondiente. Además, al final del programa se visualizara un mensaje que indica cuantas veces se ha producido dicho error.
- 4) Escribe un programa que lea una secuencia de números hasta que el usuario inserte letras. Al final del programa, se debe visualizar el numero mas alto entre todos los leídos

Puede suceder que el usuario no haya llegado a insertar ningún numero. En este caso, hay que visualizar el mensaje que corresponda.

Hay que usar un método que pida un numero y que controle el posible error que pudiera producirse si el usuario insertara letras. En este caso, el método visualizara un mensaje informando de lo ocurrido.

- 5) Escribe un programa que calcule el área y el perímetro de una superficie rectangular. Para ello, se pedirá el valor de los lados del rectángulo

Valida que los datos que inserta el usuario sean mayores o iguales que cero. Si no es así, se pedirá de nuevo el lado indicado incorrectamente. También se validara que el usuario inserte números y no letras.

Hay que usar un método que se encargue de pedir el valor de uno de los lados.

- 6) Modifica el ejercicio anterior de tal forma que no calcule el área y el perímetro de un solo rectángulo, sino de varios; exactamente, hasta que la base del rectángulo sea cero. Por supuesto, hay que validar que el usuario no inserte letras y que el valor de los lados sea positivo. Se admite que la altura tenga valor cero.

- 7) Escribe un programa que pida una serie de números hasta que la suma acumulada ascienda a 1000. Hay que visualizar cada uno de los números que se inserten.

Al final del programa, se visualizara un mensaje que indique si alguna vez se insertaron letras en vez de números

Hay que usar un método que pida un numero y visualice un mensaje de error si se insertaran letras.

- 8) Escribe un programa que pida números hasta que se inserten 10 números pares. Posteriormente, el programa debe visualizar la suma de todos los números que se han insertado, incluidos los impares.

Hay que validar que los datos que se inserten con el teclado sean números

- 9) Escribe un programa que lea tres números y visualice el mayor de ellos. Valida que el usuario inserte números Si en la inserción de alguno de los números se insertaran letras, se insertarían de nuevo los tres números

- 10) Escribe un programa que pida los datos de dos empleados. De cada empleado nos interesa: su nombre, edad y sueldo (con decimales). Hay que validar la entrada de datos, es decir, si insertan letras donde corresponden números, se volverá a pedir dicho dato. Posteriormente, hay que visualizar los datos del empleado mas joven. Si los dos empleados tienen los mismos años, se visualizara la información de los dos.