

**Folla 4.1.- Clase Number. A clase JOptionPane de Swing**

1. Empregando a clase **Integer** fai un programa que pida unha cadea de texto en binario, e mostres o seu correspondente número enteiro.
2. Consulta a API da clase **Integer**, para ver a sintaxe dos métodos *toBinaryString()*, *toOctalString()*, e *toHexString()*, e fai un programa cun menú que permita:
  - a) introducir un enteiro
  - b) mostrar o enteiro en binario
  - c) mostrar o enteiro en octal
  - d) mostrar o enteiro en hexadecimal
3. Empregando as variables de clase **MAX\_VALUE** e **MIN\_VALUE**, fai un programa que mostre por pantalla os valores máximos e mínimos para **Byte**, **Short**, **Integer**, **Float**, **Double** e **Long**.
4. Emprega a clase **JOptionPane** para ir pedindo unha letra por teclado (se son varias, o programa collerá a primeira). Despois emprega os métodos da clase **Character** para formar unha cadea de texto que indique se o carácter é ou non unha letra, se está en minúsculas ou maiúsculas, e se é ou non un número.
5. Empregando os métodos da clase anterior, pide co **JOptionPane** un usuario e contrasinal, simulando o acceso a un contido. Este contrasinal só será válido se ten unha letra maiúscula, un díxito e un carácter que non é nin díxito nin letra, pedíndoo de novo ata que sexa válido co **JOptionPane**. O programa rematará cando o contrasinal sexa válido, indicando ao usuario que pode acceder
6. Fai un programa que faga repetidamente conversións de euros a dólares ou viceversa, empregando **JOptionPane**. O valor numérico do resultado deberá ter sempre un ancho de 7 en total, con 2 decimais. O programa deberá preguntar que tipo de cambio quere facer, de cada vez, ou se quere saír do programa: mostra un menú con varias liñas empregando “\n” en linux (ou “\r\n” en Windows). Tamén podes empregar “%n” se empregas o método da clase **String.format()**
7. Fai un programa que simule do xogo do aforcado. Teremos 8 gardadas de 8 letras nun array de **String**. Cando o programa elixa aleatoriamente a palabra, aparecerá unha ventá mostrando 8 liñas “\_ \_ \_ \_ \_ \_ \_ \_” e o usuario terá que ir introducindo letras. Cando teclee unha letra da palabra elixida, a letra aparecerá no que se ve (por exemplo se a palabra é elefante, e introduce a 'e' verase así “e \_ \_ \_ \_ e”). Terá 10 intentos para atopar a palabra. Mostra ademais da palabra incompleta, os intentos que lle quedan, e as letras que foi elixindo previamente.
8. Define unha clase **Corredor**, con nome, DNI, e marca en 100m. As marcas de 100m gárdanse en segundos, décimas e centésimas de segundo. Fai de forma automática os getters, setters, e o método **toString()**. Modifica a frase ao teu xeito e comproba que se creas 2 xogadores, e os mostras con **System.out.println(corredor1)** o que sae por pantalla é o que está definido no método **toString()**.
9. Modifica o exercicio anterior para pedir e mostrar os datos dos 2 xogadores con **JOptionPane**. Comproba tamén que a letra do DNI é correcta (podes ver:  
<https://www.interior.gob.es/opencms/ca/servicios-al-ciudadano/tramites-y-gestiones/dni/calculo-del-digito-de-control-del-nif-nie/>