

PART 1 – POO

Implementar una classe **Joc** amb les següents característiques:

- Atributs: Té com a atribut públic un enter que indica el nombre de vides que li queden al jugador en la partida actual.
- Mètodes:
 - o Té com a mètode el **constructor** que accepta un paràmetre de tipus sencer que indica el nombre de vides inicials amb les quals parteix el jugador.
 - o Té un mètode **mostraVidesRestants** que visualitza per pantalla el nombre de vides que li queden al jugador en la partida actual.
 - o A més esta classe té també el **main** que ha de realitzar el següent: Crea una instància de la classe **Joc** indicant que el nº de vides és 5, crida al mètode **mostraVidesRestants** de l'objecte creat, resta una vida al valor de l'atribut amb les vides i torna a cridar a **mostraVidesRestants**; crea una altra instància de la classe **Joc** indicant que el nombre de vides és també de 5, crida al mètode **mostraVidesRestants** de la nova instància i després al de la instància anterior.

PART 2 – HERÈNCIA I POLIMORFISME

En aquesta part s'implementarà un joc en el qual l'usuari haja d'endevinar un número que coneix el programa.

Classe Joc

- No volem crear instàncies d'aquesta classe.
- Ha d'ocultar-se a qualsevol altra classe l'atribut amb les vides.
- Crear un nou atribut, també ocult per a la resta de classes, que guardi el nombre de vides que inicialment se li van passar al constructor de l'objecte. Este atribut s'utilitzarà per a poder reiniciar el joc.
- Crear un altre atribut també ocult i de tipus sencer que guardi el rècord. Este atribut serà un atribut de classe, per la qual cosa serà comuna a tots els jocs que s'implementen. Inicialment tindrà el valor 0.
- Afegeix un mètode **reiniciarPartida** que no prengui paràmetres i que assigni a l'atribut de vides actuals el número de vides que s'havien indicat en cridar al constructor de l'objecte.
- Afegeix un mètode **llevarVida** que disminueixi en 1 el nombre de vides del jugador i retorne un boolean indicant si al jugador li queden més vides o no. En cas que al jugador no li queden més vides, este mètode ha de mostrar un missatge "Joc Acabat" per pantalla.
- Afegeix un mètode **jugar** que no prengui paràmetres i que hauran d'implementar les classes derivades.
- Comenta el mètode main d'aquesta classe, ja que l'implementarem en altra classe per provar el codi.

Classe JocEndevinaNumero

- És un tipus de joc. Té un constructor que pren dos paràmetres de tipus sencer. El primer és el nombre de vides i el segon paràmetre és un número a endevinar entre 0 i 10.
- Implementa el mètode **jugar** de la classe base: Crida al mètode **reiniciarPartida** que ha heretat. Mostra un missatge a l'usuari demanant que endevine un número entre el 0 i el 10. Llig un enter del teclat* i el compara amb el valor predefinit pel programador:
 - o Si és igual, mostra un missatge "Has encertat!!" i, després, ix del mètode;
 - o si és diferent, crida al mètode **llevarVida** heretat: si el mètode **llevarVida** retorna true, significa que encara li queden més vides al jugador pel que es mostra un missatge indicant si el número a endevinar és major o menor i se li demana que ho intente de nou; si el mètode **llevarVida** retorna false, significa que ja no li queden més vides al jugador, amb el que ix del mètode **jugar**.

*Pots utilitzar la classe definida al final del document per llegir per teclat.

Classe Aplicacio

- Conté un **main** que, després de crear una instància de la nova classe JocEndevinaNumero que s'ha creat, crida al mètode **jugar**.

PART 3 – INTERFÍCIES

Implementa una interfície **IJugable** que implementaran els jocs desenvolupats fins ara i els nous que es desenvoluparan. Esta interfície ens permetrà especificar una sèrie d'operacions comunes que han d'implementar tots els jocs i que ens permetran manejar-los de manera genèrica posteriorment.

Classe Joc

- Comenta el seu mètode abstracte **jugar**, però la classe es continuarà mantenint com a abstracta ja que NO ens interessa que es creen instàncies d'ella directament.

Interfície IJugable

- Disposarà d'un mètode **jugar** que complirà el mateix objectiu que el que s'ha llevat a la classe Joc. S'incorporarà un mètode **mostrarNom** que no reba cap paràmetre i que oblige les classes que implementen la interfície a mostrar un missatge per pantalla amb el nom del joc.
- S'incorporarà un mètode **mostrarInfo** que no reba cap paràmetre i que oblige les classes que implementen la interfície a mostrar un missatge per pantalla amb una descripció de com jugar al joc.

Classe JocEndevinaNumero

- Ha d'implementar la interfície **IJugable**. El mètode **mostrarNom** visualitzarà per pantalla el text "Endevina un número".

- El mètode **mostrarInfo**, visualitzarà per pantalla una descripció de com es juga al joc, informant del nombre d'intents que se li donen al jugador.
- Afegeix un mètode **validarNumero** que torna un boolean i pren un enter (el número introduït per l'usuari). En aquesta classe, sempre es retornarà true; es cridarà des de el mètode **jugar**, però si torna false, no llevarà vida.

Classe JocEndevinaParell

- És un tipus de JocEndevinaNumero.
- Redefineix el mètode **mostrarNom** perquè visualitzi per pantalla el text Endevina un nombre parell. Redefineix el mètode **mostrarInfo** i el mètode **validarNumero** (si es imparell també mostrarà missatge d'error per pantalla).

Classe JocEndevinaImparell

- És un tipus de JocEndevinaNumero.
- Redefineix el mètode **mostraNom** perquè visualitzi per pantalla el text Endevina un nombre imparell. Redefinix el mètode **mostrarInfo** i el mètode **validarNumero** (si es parell també mostrarà un missatge d'error per pantalla).

Classe Aplicacio

- En el mètode **main** es crearà un objecte de cadascun dels jocs esmentats. Com a n° de vides de cada joc, es posarà 3 i com a n° a adivinar un valor qualsevol entre el 0 i el 10. A continuació crida els mètodes **mostraNom**, **mostrarInfo** i **jugar** de cadascun dels tres objectes creats.
- Afegeix el mètode **triarJoc**, públic i estàtic que no pren paràmetres i retorna un objecte del tipus **IJugable**.
- Crea un arrayList de tipus **IJugable** i afegeix els tres jocs anteriors.

A partir d'este moment, només es treballarà amb este array per a referir-se a qualsevol dels jocs.

- Mostra un menú per pantalla amb el nom dels tres jocs i demana a l'usuari que trie un joc introduint un número entre 0 i 2. Si el número introduït no és vàlid, continuarà demanant a l'usuari un número vàlid. Retorna l'element del array corresponent al número introduït per l'usuari.
- Si el número introduït si es vàlid, crida al mètode **mostrarNom** d'este joc. A continuació crida al mètode **mostrarInfo** del joc. Crida al mètode **jugar** del mateix per a començar una nova partida. Finalment, després de concloure la partida, pregunta a l'usuari si desitja jugar de nou i en cas afirmatiu torna a repetir els passos anteriors.

AMPLIACIÓ - Control de rècord

- Afegir un mètode **actualitzarRecord** en la classe **Joc** que compare el valor actual de rècord amb el nombre de vides restants.
 - Si el nombre de vides restants és = al rècord, mostrarà un missatge indicant que s'ha aconseguit igualar el rècord. Si el nombre és menor, no farà res.

- Si el nombre de vides restant és major que el rècord, actualitzarà el rècord i mostrarà un missatge dient que este s'ha batut i quin és el seu nou valor.
- En el mètode jugar de la classe `JocEndevinaNumero`, si encerta el número es cridarà al mètode **actualitzarRecord**.

CLASSE Teclat

```
import java.io.*;

public class Teclat {

    public static char lligCaracter() {
        char ch;
        try {
            ch = lligCadena().charAt(0);
        } catch( Exception e ) {
            ch = 0;
        }
        return ch;
    }

    public static String lligCadena() {
        BufferedReader br =
            new BufferedReader(new InputStreamReader(System.in));
        String str;
        try {
            str = br.readLine();
        } catch( Exception e ) {
            str = "";
        }
        return str;
    }

    public static int lligEnter() {
        int num;
        try {
            num = Integer.parseInt( lligCadena().trim() );
        } catch( Exception e ) {
            num = 0;
        }
        return num;
    }

} //Per provar el funcionament, recorda: Teclat.lligCadena(); etc.
```