P2 – Iniciació POO

Lliurament

Cal que facis un TAG i li posis el nom següent "UFX_PY_NCognom" on X és el número de UF que toqui (en aquest cas, 4) i Y és el número de pràctica, en aquest cas, 2 i NCognom són la inicial del teu nom i el teu cognom sencer. Es revisarà que els TAGS entregats estiguin generats en data correcta segons tasca del moodle, en cas contrari, es considerarà NO ENTREGAT.
Entrega també el document en PDF responent les preguntes.

Enunciat

1. **Responeu les preguntes següents, relatives al procés de descomposició en objectes d'un partit de futbol de primera divisió**

   - Quants objectes jugador hi ha?
     Como mínimo 22, 11 por los jugadores de cada equipo. Aunque este número puede variar dependiendo de cuantos estén en banquillo.

   - Pot variar el nombre d'objectes jugador?
     Si, todo dependerá de la cantidad de jugadores que se hayan presentado

   - Té sentit que hi hagi objectes arbitre:?
     Si, para representar a todos los árbitros que se encuentran en un partido, y las diferentes funciones que realizan

   - Té sentit que hi hagi les classes Defensa o Porter?
     Si, para diferenciar a las diferentes posiciones que se adoptan en un partido. Esto tiene especial importancia con los porteros, ya que se ven afectados de una manera diferente respecto al resto de jugadores.

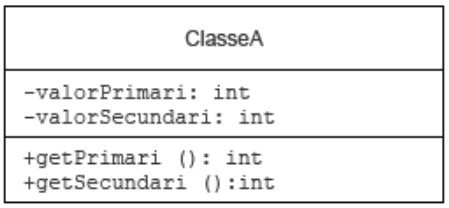
   - Té sentit que el número de la samarreta sigui un atribut de jugador?
     Si, ya que se trata de un identificador.

   - I si dos jugadors es poden canviar la samarreta entre ells?
     Habría que implementar una funcionalidad para cambiar el atributo.

   - Té sentit que un objecte arbitre tingui una operació rebreTargeta?
     Si, para representar una acción que se puede dar a cabo en un partido

2. Classe A:

L'objectiu d'aquesta activitat és veure com instanciar classes fetes per vosaltres. Creeu una classe anomenada ClasseA d'acord a la següent especificació. Els dos mètodes set… serveixen per llegir els valors emmagatzemats a cada atribut.



Afegiu tres constructors:

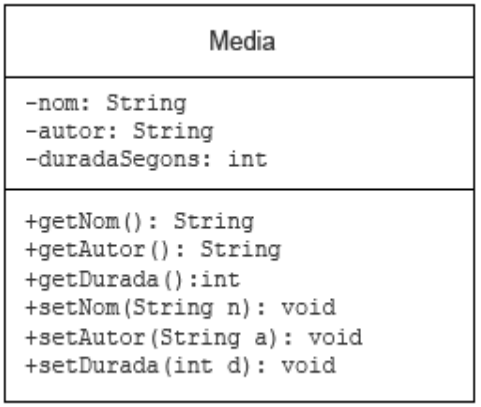
- public ClasseA(), que assigna els valors 5 i 10 als dos atributs, respectivament.
- public ClasseA(int vp), que assigna "vp" a "valorPrimari" i el valor 10 a "valorSecundari".
- public ClasseA(int vp, int vs), que assigna "vp" i "vs" als dos atributs, respectivament.

Proveu que heu fet la classe correctament afegint el següent mètode main al codi i executant la classe.

```
public static final void main (String[] args) {
ClasseA a = new ClasseA();
ClasseA b = new ClasseA(20);
ClasseA c = new ClasseA(20, 40);
System.out.println("L'objecte [a] conté: " + a.getPrimari() + ", " + a.getSecundari());
System.out.println("L'objecte [b] conté: " + b.getPrimari() + ", " + b.getSecundari());
System.out.println("L'objecte [c] conté: " + c.getPrimari() + ", " + c.getSecundari());
}
```

3. Objecte MEDIA

Donada la següent especificació, genereu el codi font d'aquesta classe. Els mètodes set… i get… serveixen per accedir i modificar, respectivament, els valors emmagatzemats als atributs. Podeu generar els constructors que vulgueu.



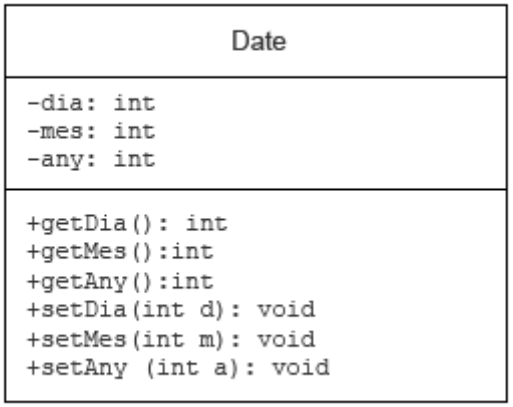
Un cop fet, comproveu que funciona amb un programa de prova (un mètode main) en una classe nova anomenada ProvaMedia, instanciant la classe i accedint i assignant valors als seus atributs.

- Proveu a imprimir al final directament l'objecte. I adjunteu la captura del que us apareix.
- Definiu el mètode toString() mostrant la informació de forma agradable (podeu consultar els exemples disponibles al moodle) i torneu a fer la prova anterior.

4. Objecte DATES

L'objectiu d'aquesta activitat és practicar la codificació d'una classe senzilla i entendre la utilitat dels mètodes set….
Donada la següent especificació, genereu el codi font d'aquesta classe. Els mètodes set… i get… serveixen per accedir i per modificar, respectivament, els valors emmagatzemats en els atributs. Podeu generar els constructors que vulgueu, però fes que tingui una data per defecte 1/1/2023.

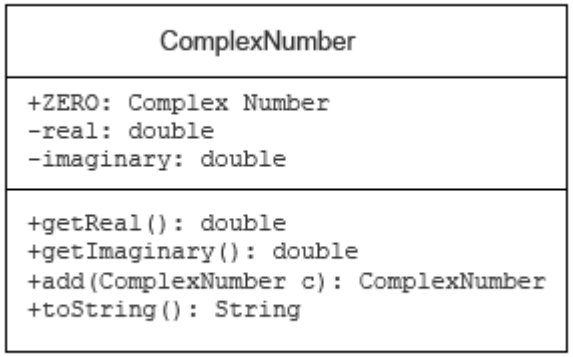
Ara bé, per a aquest exercici, cal que els mètodes set… controlin que el valor que es vol assignar és correcte, i si no és el cas, no fer res (deixar el valor que hi havia inicialment). Mai no hi pot haver una data amb un dia-mes-any que no sigui coherent. S'han de controlar casos exagerats, com ara intentar assignar el valor de mes 25, o el dia 50, o com ara intentar desar el dia 31 quan el mes actual és el febrer. Recordeu també els anys de traspàs.



Un cop fet, comproveu que funciona amb un programa de prova (un mètode main), instanciant la classe i accedint i assignant valors als seus atributs.

5. Objecte ComplexNumber

L'objectiu d'aquesta activitat és practicar la codificació d'una classe que, al mateix temps, usa classes al seu codi.
Donada la següent especificació, genereu el codi font d'aquesta classe. Podeu afegir-hi els constructors que vulgueu. La constant ZERO es refereix a una instància de ComplexNumber amb els valors "real" i "imaginary" ambdós a 0.



La descripció dels mètodes és:

- **get…**, serveixen per accedir als valors dels dos atributs.
- **add(…)**, retorna un nou objecte ComplexNumber. El valor de cadascun dels seus dos atributs nous és la suma del mateix atribut al paràmetre "c" i al propi objecte que executa add (o sigui, "suma"). Per exemple, el valor "real" del nou objecte és la suma de l'atribut "real" de "c" i de l'objecte que executa add.
- **toString()**, retorna una cadena de text on es mostren els valors dels atributs de l'objecte (per exemple, separats per una coma).

Un cop fet, comproveu que funciona amb un programa de prova (un mètode main), instanciant la classe diverses vegades i invocant els diferents mètodes. Ajudeu-vos del mètode toString per mostrar les dades per pantalla.

6. Construir una classe Punt.

L'objectiu d'aquesta classe és emmagatzemar les coordenades x i y d'un punt en un pla.

Les condicions de construcció són les següents:

- Crear un paquet anomenat "com.negoci".
- La classe ha de tenir un constructor que inicialitzi un punt. Els paràmetres del constructor proporcionen les coordenades inicials.
- Un mètode modificador que desplaça el punt al llarg dels eixos x i y. Aquest mètode rebrà com a paràmetre quina coordenada s'ha d'augmentar i el valor a augmentar. S'han d'utilitzar només operadors d'assignació.
- Dos mètodes observadors que recuperen els valors de les coordenades del punt.
- Un mètode que retorni una cadena de text (String) amb les coordenades del Punt (toString).
- Un mètode que calculi la distància entre dos punts (calculada utilitzant el teorema de Pitàgores).
- Un mètode que calculi i retorni el punt mig entre altres dos.
- Validar si els valors introduïts són diferents de null.
- La classe punt, els seus mètodes i atributs han de ser cridats des d'una altra classe anomenada "principal" que es troba al paquet "com.interfaces".
- Construir un menú mitjançant el qual es realitzin les trucades a cadascuna de les opcions de càlcul i/o assignació anteriorment introduïdes.