

Programació Orientada a Objectes.

Declaració de Classes:

Exercici 1:

Crea una classe anomenada Parell que compleixi la següent especificació. Els dos mètodes `getPrimari()` i `getSecundari()` serveixen per llegir els valors emmagatzemats als atributs privats.

Parell
<code>-valorPrimari: enter = 5</code> <code>-valorSecundari: enter = 10</code>
<code>+getPrimari(): enter</code> <code>+getSecundari(): enter</code> <code>+Parell()</code> <code>+Parell(valorPrimari:enter)</code> <code>+Parell(valorPrimari:enter, valorSecundari:enter)</code>

Nota: Podem observar que en aquest cas no tenim cap mètode set, és a dir, els atributs `valorPrimari` i `valorSecundari` són atributs de **només lectura (readonly)**

Els tres constructors inicialitzaran els atributs corresponents i, en cas que no s'hi indiqui un dels atributs, els carregarà amb el seu valor per defecte.

Exercici 2: (Examen Ordinari 2014/15)

Donada la següent especificació (Cançó), 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.

Canso
-nom: cadena -autor: cadena -durada: enter
+getNom(): cadena +getAutor(): cadena +getDurada(): enter +setNom(nom:cadena) +setAutor(autor:cadena) +setDurada(durada:enter)

Exercici 3

Donada la següent especificació, genera el codi font d'aquesta classe. Els mètodes set... i get... serveixen per accedir i modificar, respectivament, els valors emmagatzemats en els atributs. Podeu generar els constructors que vulgueu, però en cas del constructor per defecte, la data a consignar hauria de ser l'1 de gener de 1753.

Data
-DIA: enter = 1 -MES: enter = 1 -ANYO: enter = 1753 -dia: enter -mes: enter -anyo: enter
+getDia(): enter +getMes(): enter +getAnyo(): enter +setDia(dia:enter) +setMes(mes:enter) +setAnyo(anyo:enter)

Ara bé, per 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. Cal 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 de febrer. Recordeu també els anys de traspàs.

- **NOTA:** Cal que aquesta classe estigui convenientment documentada.

Exercici 4 (Examen Ordinari 2014/15)

Dona la següent especificació, genera el codi font d'aquesta classe.

NumeroComplex
+ZERO: NumeroComplex -real: double -imaginari: double
+getReal(): double +getImaginari(): double +suma(c:NumeroComplex): NumeroComplex +toString(): cadena

Afegeix-hi dos constructors:

- El constructor per defecte crearà un nombre complex amb els valors «real» i «imaginari» ambdós a 0.
- Un altre constructor amb dos paràmetres per inicialitzar la part real i la part imaginària.
- La constant ZERO es refereix a una instància a NumeroComplex amb els valors «real» i «imaginari» ambdós a 0.
- get..., serveix per accedir als valors dels dos atributs.
- suma(...), retorna un nou objecte NumeroComplex. 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. **ATENCIÓ, estem demanant que retornis la suma de l'objecte actual i el que es passa per paràmetre, no que modifiquis l'objecte actual sumant-li el que passes per paràmetre.**
- ToString(), retorna una cadena de text on es mostren els valors dels atributs de l'objecte (per exemple, separats per una coma)
- Afegeix un mètode de classe de nom **suma** que, donats quatre reals X1, Y1, X2, Y2 retorni un objecte NumeroComplex resultat de sumar

el nombre complex NumeroComplex (X1,Y1) amb el nombre complex NumeroComplex (X2,Y2).

- **NOTA:** Cal que aquesta classe estigui convenientment documentada.