

GRAU D'ENGINYERIA INFORMÀTICA
Curs acadèmic 2018-19

PROGRAMACIÓ II

Problema 5:

Exercicis interfícies

Diagrama de flux

Diagrama de seqüència



EXERCICIS INTERFÍCIES

Exercicis interfícies

- Començarem amb un exercici d'interfícies per a reforçar aquest aspecte de la pràctica i de la teoria.

Exercicis interfícies: les guitarres

- Donada la següent interfície (GuitarInterface.java):

```
public interface GuitarInteface
{
    public void tocar();
    public void afinar();
}
```

- Defineix una classe Guitarra que la implementi
- Aquesta classe imprimirà el missatge “tocant la Guitarra” dins del mètode tocar, i “afinant la guitarra” dins del mètode afinar

Exercicis interfícies: les guitarres

```
public class Guitarra implements GuitarInterface{  
    public Guitarra(){}  
    public void tocar(){  
        System.out.println("Tocant la guitarra");  
    }  
    public void afinar(){  
        System.out.println("Afinant la guitarra");  
    }  
}
```

Exercicis interfícies: les guitarres

- Contesta a la següents preguntes

1. Si implementem la interfície GuitarInterface, estem obligats a implementar els mètodes tocar i afinar?

2. Es correcta la següent instrucció? Raona la resposta

```
GuitarInterface unaGuitarra = new GuitarInterface();
```

Exercicis interfícies: les guitarres

- Contesta a la següents preguntes
 1. Si implementem la interfície GuitarInterface, estem obligats a implementar els mètodes tocar i afinar? **Sí**
 2. Es correcta la següent instrucció? Raona la resposta. **No és correcta; no podem fer un new d'una interfície**

```
GuitarInterface unaGuitarra = new GuitarInterface();
```

Exercicis interfícies: les guitarres de Laura, Sergio i Jeroni

- Continuem amb l'exercici de les guitarres, però el compliquem una mica més
- Implementa les següents classes:
 - LauraGuitar
 - SergioGuitar
 - JeroniGuitar
- Aquestes classes implementen la interfície GuitarInterface, i imprimeixen missatges personalitzats als mètodes tocar i afinar (per ex. "Tocant (afinant) la guitarra de Sergio")

Exercicis interfícies: les guitarres de Laura, Sergio i Jeroni

```
public class LauraGuitar implements GuitarInterface{
    public Guitarra(){}
    public void tocar(){
        System.out.println("Tocant la guitarra de la Laura");
    }
    public void afinar(){
        System.out.println("Afinant la guitarra de la Laura");
    }
}
```

```
public class Sergio (Jeroni) Guitar implements GuitarInterface{
    public Guitarra(){}
    public void tocar(){
        System.out.println("Tocant la guitarra de Sergio
(Jeroni)");
    }
    public void afinar(){
        System.out.println("Afinant la guitarra de Sergio
(Jeroni)");
    }
}
```

Exercicis interfícies: les guitarres de Laura, Sergio i Jeroni

- Ara implementa un menú similar al del Lliurament 3 per tal que l'usuari pugui escollir si vol crear una guitarra de Laura, de Sergio o de Jeroni
- **Per declareu la variable guitarra, utilitzeu una única variable – atribut de classe - i no pas tres (una per cada tipus de Guitarra), seguint l'esquema següent.**

Exercicis interfícies: les guitarres de Laura, Sergio i Jeroni

```
public class ExerciciGuitarres{  
    //Quin tipus? guitar;  
    main(...)  
    //show menu  
    switch (opcio)  
        case 1:  
            // guitar = new LauraGuitar();  
        case 2:  
            // guitar = new JeroniGuitar();  
        case 3:  
            // guitar = new SergioGuitar();  
        break:  
}
```

Com declareu l'atribut? De quin tipus hauria de ser i perquè?

Exercicis interfícies: les guitarres de Laura, Sergio i Jeroni

```
public class ExerciciGuitarres{  
    GuitarInterface guitar;  
    main(...)  
    //show menu  
    switch (opcio)  
        case 1:  
            // guitar = new LauraGuitar();  
        case 2:  
            // guitar = new JeroniGuitar();  
        case 3:  
            // guitar = new SergioGuitar();  
        break:  
}
```

Com declareu l'atribut?

Exercicis interfícies: les guitarres de Laura, Sergio i Jeroni

```
public class ExerciciGuitarres{  
    GuitarInterface guitar;  
    main(...)  
    //show menu  
    switch (opcio)  
        case 1:  
            // guitar = new LauraGuitar();  
        case 2:  
            // guitar = new JeroniGuitar();  
        case 3:  
            // guitar = new SergioGuitar();  
        break:  
}
```

De quin tipus hauria de ser i perquè? **GuitarInterface**, perquè les classes Laura, Jeroni i Sergio són també d'aquest tipus.

Exercicis interfícies

- Un cop acabat el primer exercici, ara farem un altre. En aquest cas, utilitzarem una interfície força interessant de JAVA, i la utilitzarem en un exercici sobre instruments musicals

Exercicis interfícies: comparador de preus d'instruments musicals; Comparable

- JAVA defineix una interfície, **Comparable**, que ens permet establir l'ordre de dos objectes (més gran, més petit, o igual)

```
public interface Comparable<E>{  
    public int compareTo(E o);  
}
```

- El mètode compareTo retorna:
 - > 0 , si l'objecte és menys petit que l'especificat
 - 0 , si són iguals
 - $<$, si l'objecte (this) és més gran que l'especificat

Exercicis interfícies: comparador de preus d'instruments musicals; Comparable

```
public interface Comparable<E>{  
    public int compareTo(E o);  
}
```

- Comparable<E>
 - Significa que és de tipus genèric
 - Substituïm E pel nom d'una classe (recordeu que les classes defineixen tipus)

Exercicis interfícies: comparador de preus d'instruments musicals

- Utilitzant la interfície Comparable, implementeu un programa que permeti comparar el preu de dos instruments musicals qualsevols i que imprimeixi per pantalla el nom de l'instrument amb preu més petit
- El programa té les següents classes:
 - InstrumentMusical
 - Guitarra
 - Piano
 - Main

Exercicis interfícies: comparador de preus d'instruments musicals

1. Implementa la classe **InstrumentMusical** amb un atribut privat preu. Implementa el mètode get i set.

Exercicis interfícies: comparador de preus d'instruments musicals

```
public class InstrumentMusical {  
  
    private int preu;  
  
    public InstrumentMusical(int p){  
  
        preu = p;  
  
    }  
  
    public int getPreu(){  
  
        return preu;  
  
    }  
  
}
```

Exercicis interfícies: comparador de preus d'instruments musicals

1. Implementa la classe **Guitarra**. Guitarra és un InstrumentMusical i és Comparable (és a dir, implementa la interfície)
2. Implementa la classe **Piano**. Piano és un InstrumentMusical i és Comparable (és a dir, implementa la interfície)

Exercicis interfícies: comparador de preus d'instruments musicals

```
public class Guitarra extends InstrumentMusical implements
Comparable<InstrumentMusical>{
    public Guitarra(int p){
        super (p);
    }
    @Override
    public int compareTo(InstrumentMusical t) {
        int res = 0;
        if (t.getPreu() > this.getPreu())
            res = -1;
        else if (t.getPreu() < this.getPreu())
            res = 1;
        else
            res = 0;
        return res;
    }
}
```

Exercicis interfícies: comparador de preus d'instruments musicals

```
public class Piano extends InstrumentMusical implements  
Comparable<InstrumentMusical>{  
    public Piano (int p){  
        super (p);  
    }  
    @Override  
    public int compareTo(InstrumentMusical t) {  
        int res = 0;  
        if (t.getPreu() > this.getPreu())  
            res = -1;  
        else if (t.getPreu() < this.getPreu())  
            res = 1;  
        else  
            res = 0;  
        return res;  
    }  
}
```

Exercicis interfícies: comparador de preus d'instruments musicals

3. Al mètode main, crea dos objectes de tipus guitarra i piano i implementa un mètode (l'anomenarem comparador) que et permeti comparar els seus preus

• De quin tipus han de ser els arguments del mètode comparador?

Exercicis interfícies: comparador de preus d'instruments musicals

```
public static void main(String[] args) {  
    // TODO code application logic here  
    Guitarra guitarra = new Guitarra(35);  
    Piano piano = new Piano(135);  
    int res = comparador (piano, guitarra);  
    //prints here  
}  
public static int comparador (Comparable c1, Comparable c2){  
    return (c1.compareTo(c2));  
}
```


DIAGRAMA DE FLUX

Diagrama de flux: exercici

- Pregunta de la memòria del L3: feu un diagrama del flux de l'execució del programa durant la reproducció de la biblioteca de fitxers.

DIAGRAMA DE SEQÜÊNCIA

Diagrama de seqüència: exercici

- Feu el diagrama de seqüència corresponent a l'exercici anterior
- El diagrama de seqüència el treballareu més endavant al Grau. Ara us donem algunes pinzellades

Diagrama de seqüència

- El **diagrama de seqüència** és un diagrama d'interacció en Unified Modeling Language (UML) que mostra la interacció dels objectes d'un sistema disposats en una seqüència de temps.
- La seva funció principal és mostrar com els objectes d'un sistema s'intercanvien missatges amb la intenció de complir un objectiu concret.

Diagrama de seqüència: exemple

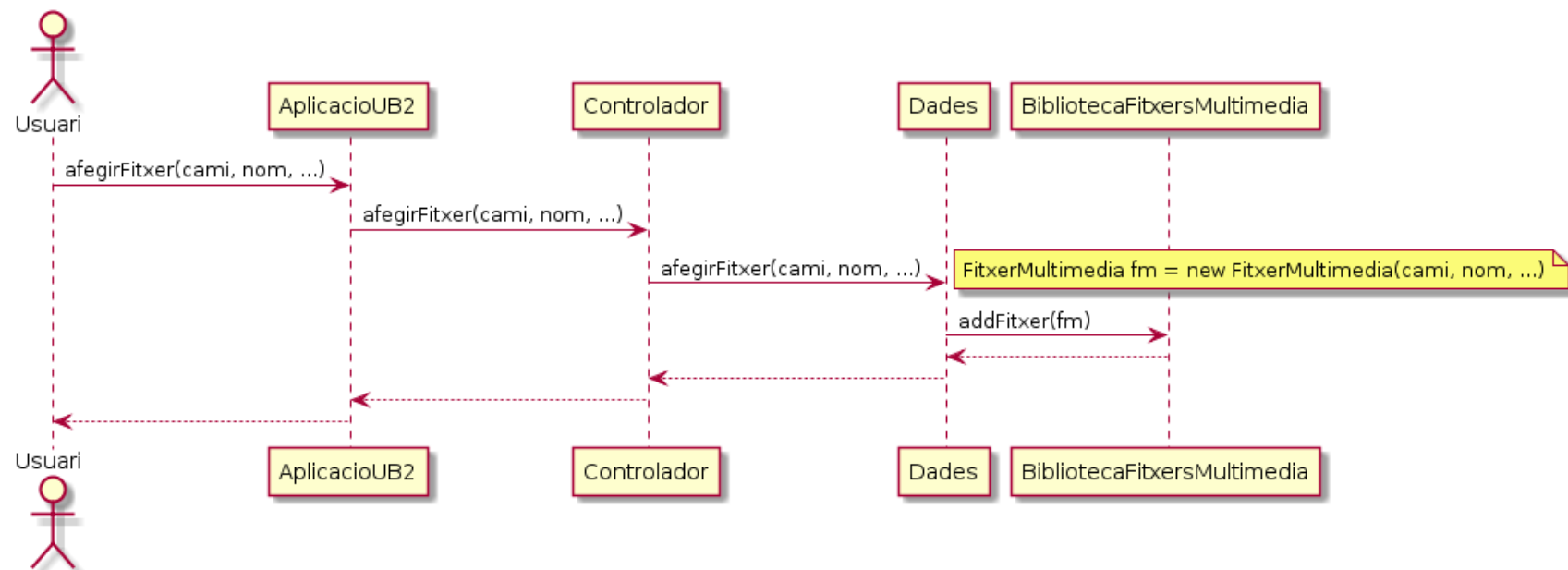


Diagrama de seqüència: algunes eines

- <https://sequencediagram.org/>
- <https://www.websequencediagrams.com/>
- <https://www.draw.io/>