Interface

Declarar i crear interfaces

Exemples

Interface

- **Les interfaces són una forma d'especificar què ha de fer una classe sense especificar el com.**
- **Les interfaces tenen una semblança amb les classes abstractes, ja que, en elles, tampoc té sentit definir objectes instancia d'una interfaz.**
- **Igual que en les classes abstractes, la classe associada es compromet a implementar tots els mètodes en elles definits, PERÒ en aquest cas la relació no és d'herència en plenitud, donat que no hi ha atributs en la definició d'una interfaz.**

Característiques d'una Interfaz

- Pot estar formada por constants i mètodes.
- Tots els mètodes que conté són ***public***.
- A més a més, els mètodes tindran algun d'aquest modificadors:

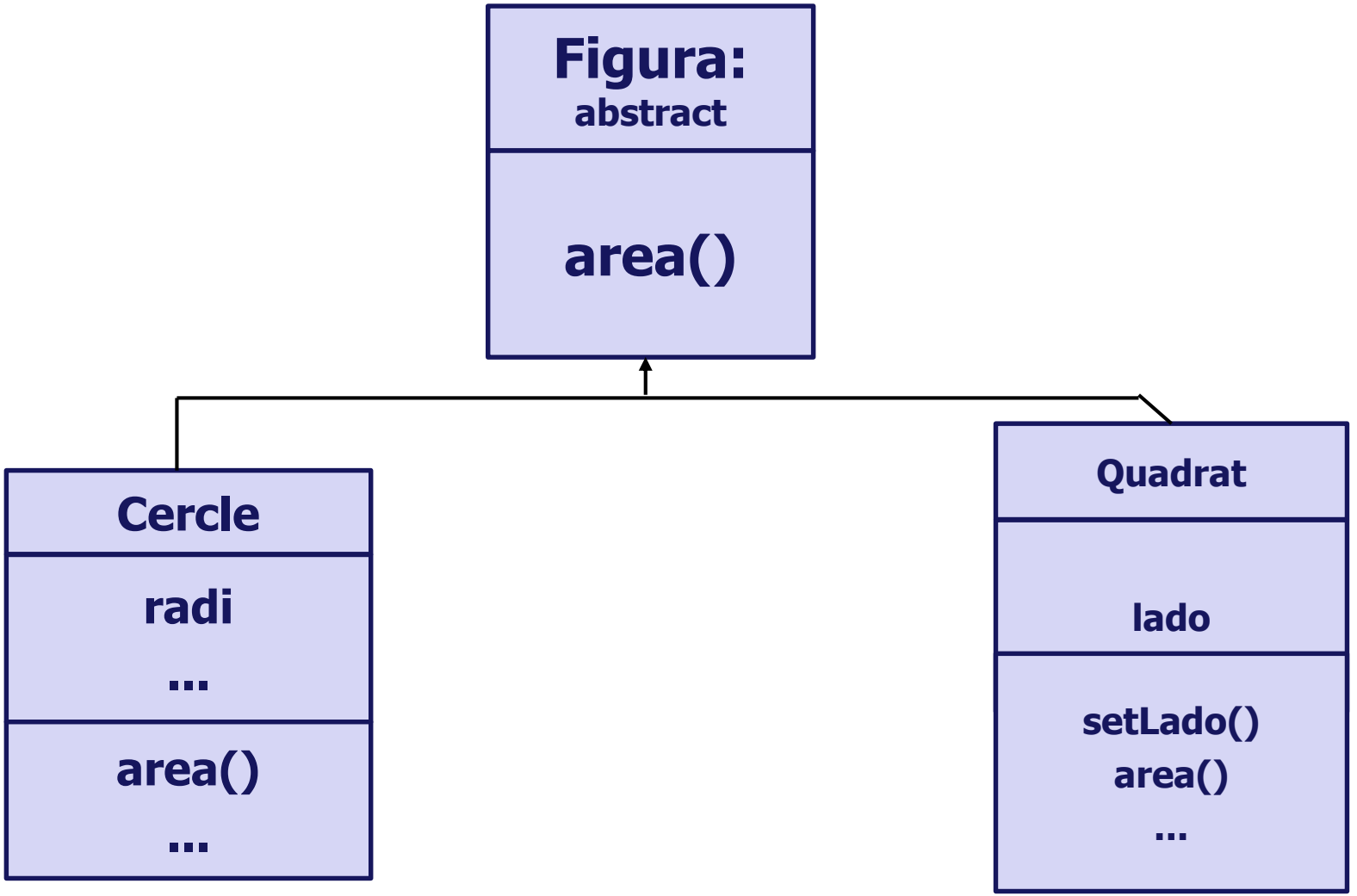
  abstract , ***default*** o ***static***

- Si no s'especifica res, sempre seran, per defecte, ***public*** i ***abstract***.
- Encara que no s'indiqui, tots els atributs són de tipus ***public static final***, es a dir, són constants.

Interface

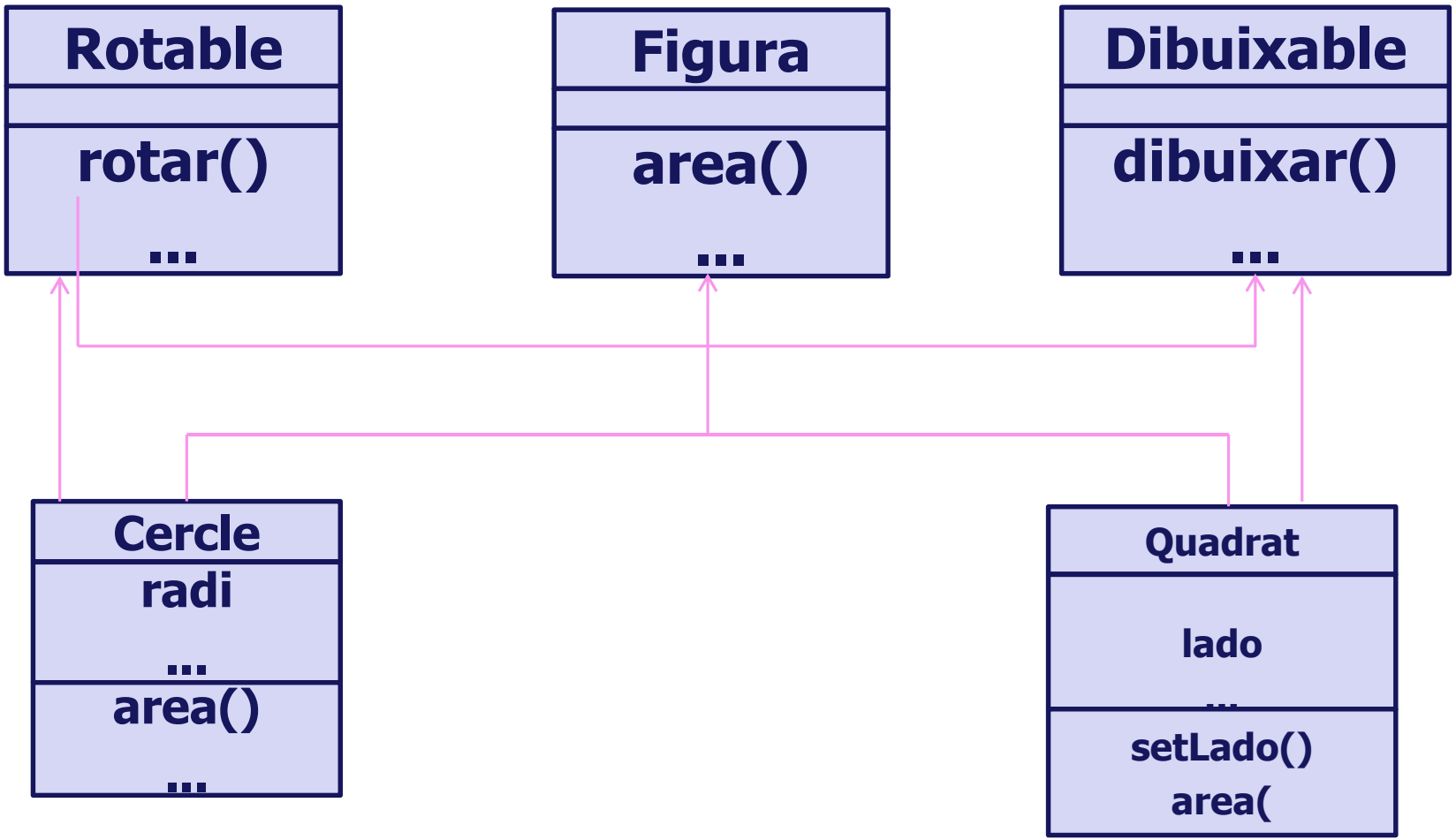
- Només declaren comportament.
- S'utilitza la paraula clau ***interface***.
- Permet simular alguns aspectes de la herència múltiple.
- Qualsevol classe que implementi una interfaz, ha de definir tots els mètodes d'aquesta interfaz. Es a dir, ha de proporcionar la implementació d'aquests mètodes.
- Si la classe no implementa tots els mètodes de la interfaz, ha de ser declarada com abstracta.

Interface: Exemple



Exemple Interface

- Exemple:



Exemple Interface

- Una classe pot implementar diferents interfases simultàniament, però només pot heretar d'una classe. (herència simple d'implementació, múltiple d'interfaces).

```
package UF4;

    public interface Rotable {
        public void rotar(double graus);

    }
```

```
public abstract class Figura {

    public abstract double area();


}
```

```
package UF4;

public interface Dibujable {
    public void dibuixa();

}
```

```
package UF4;

public class Cercle extends Figura implements Dibujable{
    private float radi;

    public float getRadi() {
        return radi;
    }

    public void setRadi(float radi) {
        this.radi = radi;
    }

    public double area(){
         return Math.PI*radi*radi;
    }
    public void dibuixa(){
        System.out.println("CERCLE");
    }

}
```

```
package UF4;

public class Quadrat extends Figura implements Dibujable, Rotable{

        private float lado;

        public float getLado() {
            return lado;
        }
        public void setLado(float lado) {
            this.lado = lado;
        }
        public double area(){
            return lado*lado;
        }
        public void dibuixa(){
            System.out.println("quadrat");
        }

        public void rotar(double graus){
            System.out.println("Rotem: "+graus);
        }

    }
```

Diferencies/Semblances entre classe abstractes i interfaces

CLASSES ABSTRACTES	INTERFACES
• Poden tenir atributs de qualsevol tipus.	• Tots els seus atributs són automàticament constants, es a dir, ***public static*** i ***final***.
• Poden tenir mètodes normals.	• No poden tenir mètodes normals
• Al menys ha de contenir un mètode ***abstract***.	• El mètodes seran ***public***, i a més a més, o seran ***abstract***, o ***default*** o ***static***.
• NO es poden instanciar	• NO es poden instanciar
• Per que una classe *hereti* de una classe abstracta s'utilitza la paraula ***extends***	• Per que una classe *implementi* una *interface*, s'utilitza la paraula ***implements***
• Una classe només pot heretar d'una classe.	• Una classe pot implementar totes les interfases que necessiti.