

# Características avanzadas de programación orientada a objetos



# Herencia

Una clase es especialización de otra

Palabra clave extends

Sólo se puede heredar de una clase (directamente)

Tener en cuenta constructores de superclase y subclase

`super()`...

# Encapsulamiento

“Proteger los datos”

- **Ámbito de acceso**
  - public: todos pueden acceder
  - protected: subclases y las del package
  - default (no poner nada): acceso desde el package, subclases solo si en el package
  - private: solo desde la propia clase

# Métodos y clases abstractas

Las clases abstractas no pueden instanciarse directamente, solo subclases

Los métodos abstractos no tienen cuerpo, hay que desarrollarlos en subclases

Si hay métodos abstractos, la clase es, obligatoriamente, abstracta.

# Interfaz

Solo declaran métodos, no los definen

Son abstractos y públicos (aunque no de indique)

Palabra reservada extends

Admite implementación múltiple

extends interfaz1, interfaz2, interfaz3

Hay que desarrollar los métodos en las clases

# Métodos y clases final

Una clase final no se puede heredar

Un método final no se puede sobrescribir

# Acceso a métodos de superclase

Si se han sobrescrito, se puede acceder a los métodos de superclase desde la subclase utilizando

```
super.nombreMetodoSobrescrito()
```

Solo se puede acceder a la clase “madre”, no a superclases más lejanas