Interfaces

UNIVERSIDAD Popular del cesar

Interfaces (Que es ?)

Las **interfaces** definen y estandarizan las formas en que pueden interactuar las cosas entre sí, como las personas y los sistemas.

La **interfaz** especifica las operaciones que pueden realizar los objetos, pero no ***cómo*** lo hacen.

Ej. Los controles en un radio sirven como una **interfaz** entre los usuarios del radio y sus componentes internos.

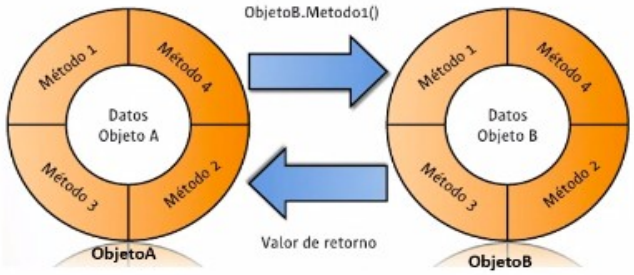
Radios con diferentes interfaces

UNIVERSIDAD Popular del cesar

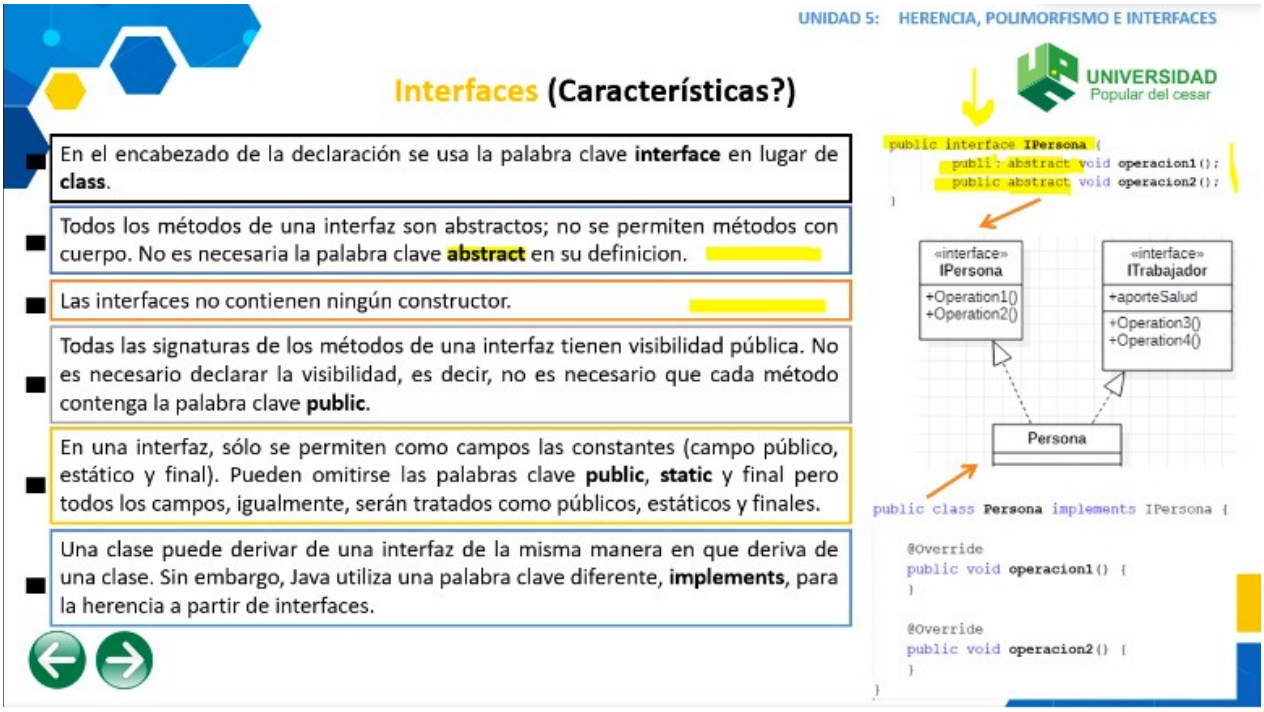
Interfaces (Que es ?)

Los objetos de software también se comunican a través de interfaces.

Una **interfaz** describe un conjunto de métodos que pueden llamarse sobre un objeto



Una **interfaz** en Java es la especificación de un tipo (bajo la forma de un nombre de tipo y un conjunto de métodos) que no define ninguna implementación para los métodos.

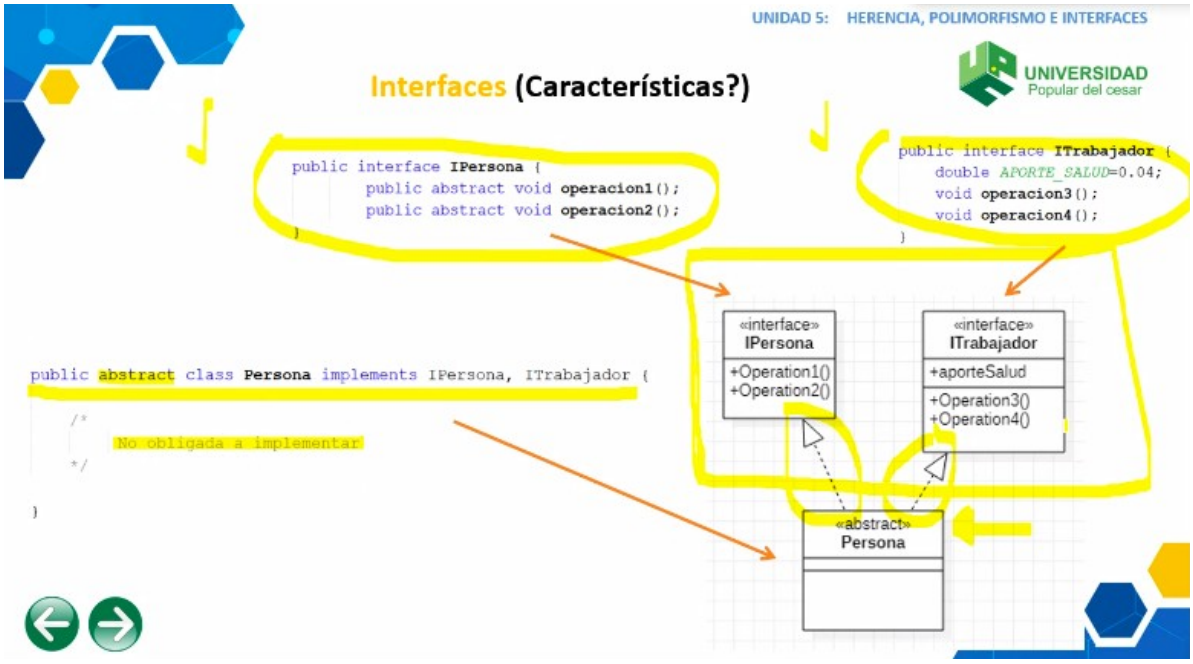


con las interfaces podemos aplicar herencia multiplica:

Coloocamos en el Implement todas las herencias que requerimos:

Implemente Personai, Trabajari por ejemplo.

Ejm con clase abstractas...

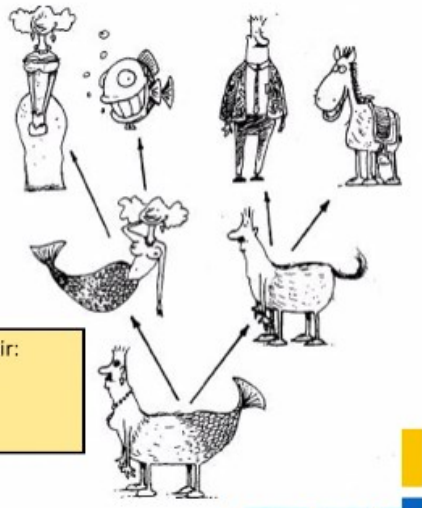


UNIVERSIDAD Popular del cesar

Interfaces (Herencia Múltiple con Interfaces)

La Herencia Múltiple presentará problemas cuando:

- Se herede varias veces de una misma clase base.
- Se herede métodos implementados de forma diferente que se llamen igual



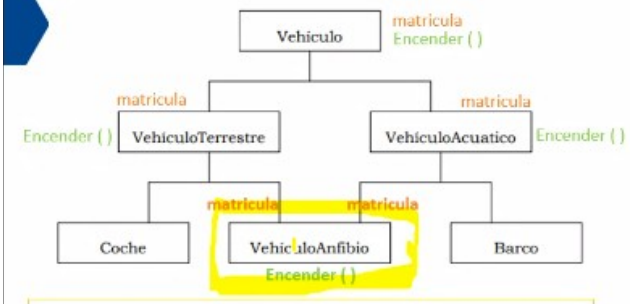
La flexibilidad de algunos lenguajes gracias a la H.M, deben pagarla en complejidad o ineficiencia.

El conflicto de la herencia múltiple ocurre porque una clase derivada puede recibir:

- **Varias implementaciones para un mismo mensaje.**
- **Varias copias de un mismo atributo.**

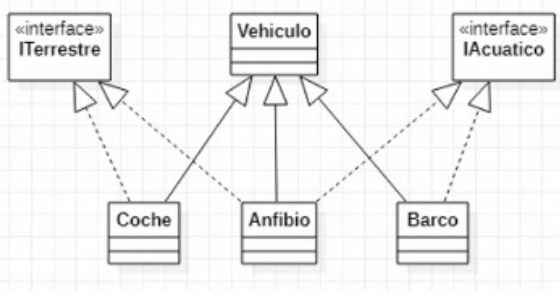
UNIVERSIDAD Popular del cesar

Interfaces (Herencia Múltiple con Interfaces)



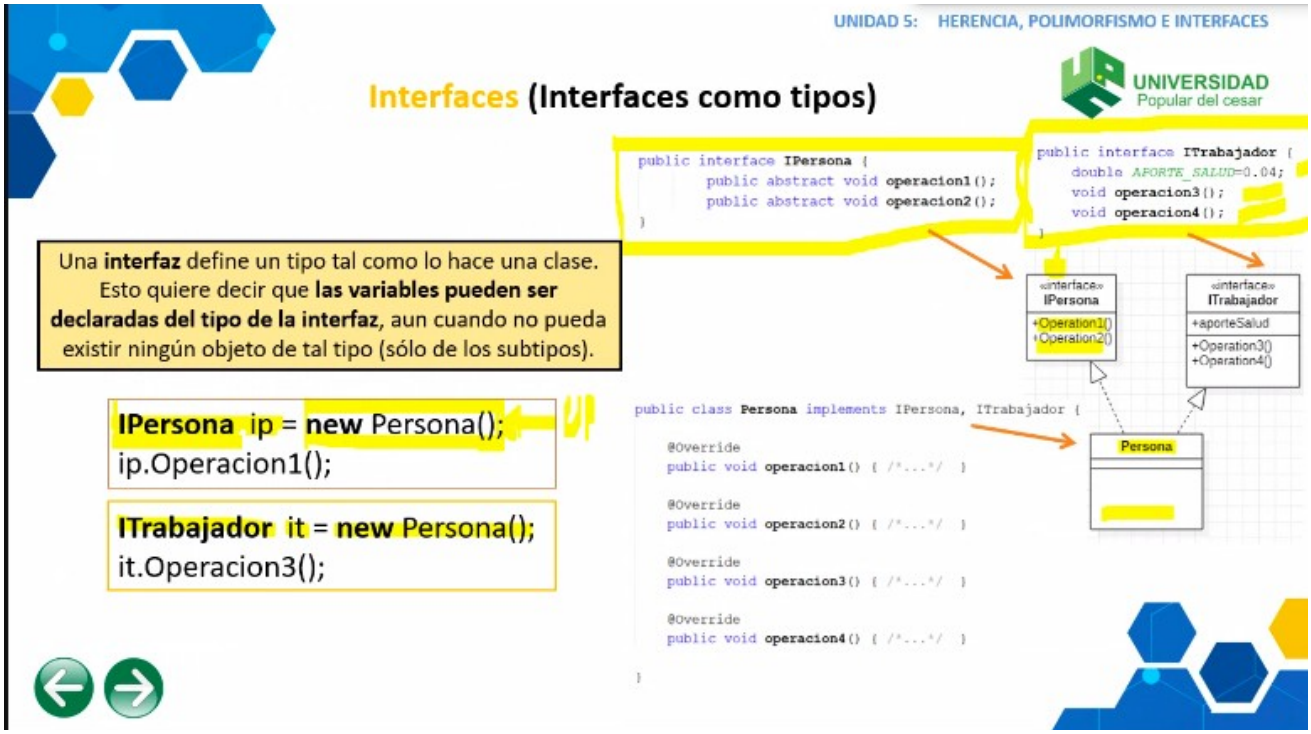
¿Qué pasaría si la clase **Vehículo** tuviera un atributo llamado matrícula? ¿Cuántas matrículas tendría la clase **VehículoAnfibio**?

¿qué pasaría si **Vehículo** tuviera un método que fuera implementado de forma diferente por **VehículoTerrestre** y por **VehículoAcuatico**.

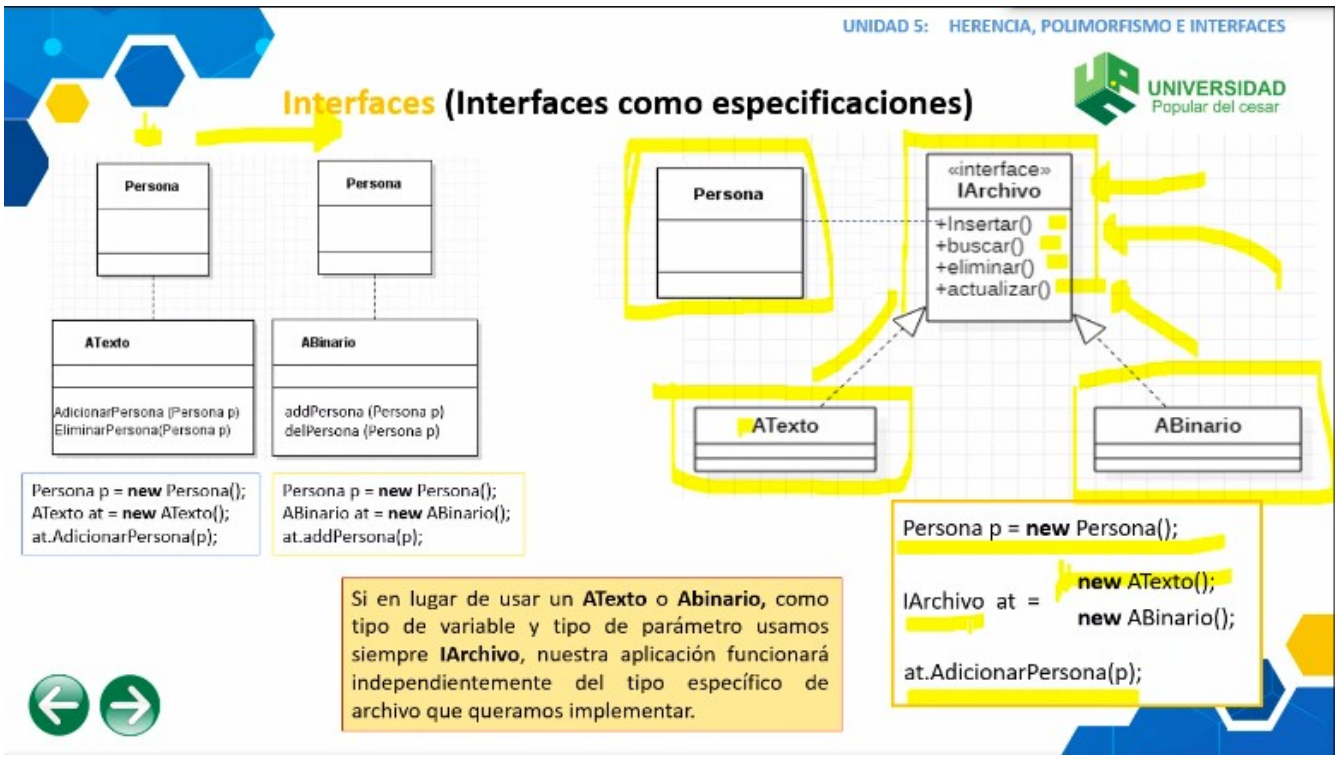


Un **interface** se pueden indicar como "padres" de una jerarquía múltiple.

Es decir, con los interfaces se obtiene lo bueno de la herencia múltiple, eliminando lo malo. No se hereda codificación sólo definición de métodos.



Cuando se crean upcasting o downcasting, es decir, se hace una generalización, solamente se puede acceder a los metodos que estén en el tipo de objeto que se creo. Por ejemplo: ip solo puede acceder a los metodos de su tipo.



UNIVERSIDAD Popular del cesar

Interfaces (Polimorfismo mediante interfaces)

Las **interfaces** permite implementar de forma polimórfica un conjunto de métodos comunes para clases no relacionadas .

UNIVERSIDAD Popular del cesar

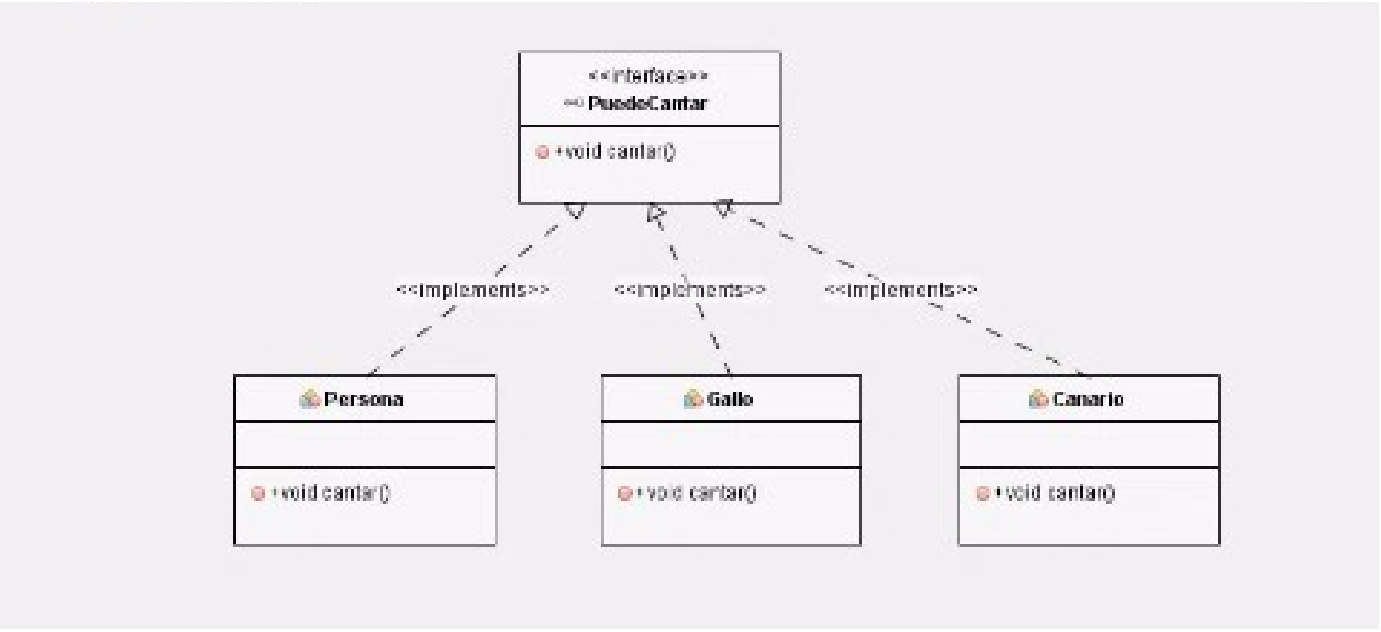
Clases abstractas Vs Interfaces

	Clases	Interfaces
Tipo	Tipo que al extenderlo mediante herencia se obtienen sus mensajes y sus implementaciones (métodos y atributos).	Tipo que al extenderlo mediante herencia se obtiene solamente mensajes.
Ventajas	Menor codificación al crear nuevos subtipos ya que los mensajes vienen con sus implementaciones.	Se pueden heredar varios interfaces sin conflictos.
Inconvenientes	Sólo se puede derivar de una de ellas.	Hay que codificar el método de cada mensaje en cada subtipo.

Interfaces útiles de java: Serializable, Cloneable, Enumeration, Comparable, List, Comparator

Cuando se debe utilizar **interfaces** o cuando se debe utilizar **Clases abstractas ??**

Ejercicio1)



Ejercicio2)