

INTERFACES



INTERFACES

- Una interfaz consiste en una colección de firmas de métodos y declaraciones de constantes agrupadas bajo un nombre.
- En una interfaz se indica qué se hace, pero no cómo se hace. Definen un protocolo de comportamiento que puede ser implementado por cualquier clase.
- Una clase puede extender a varias interfaces.

```
package nombrePaquete;
```

```
{imports}
```

```
[public interface NombreInterfaz [extends ListaDeSuperInterfaces] {  
[Constantes]  
[Signaturas de métodos]  
}
```



INTERFACES



- Ejemplos:

```
public interface Observer {  
    /**  
     * This method is called whenever the observed object is changed. An  
     * application calls an Observable object's  
     * notifyObservers method to have all the object's  
     * observers notified of the change.  
     *  
     * @param o the observable object.  
     * @param arg an argument passed to the notifyObservers  
     * method.  
     */  
    void update(Observable o, Object arg);  
}
```

INTERFACES

```
interface InstrumentoMusical{  
void sonar();  
int LA = 440;  
}
```

```
public class Cello implements InstrumentoMusical {
```

```
    @Override  
    public void sonar() {  
        System.out.println("Tocamos en " + LA);
```

```
    }
```

```
}
```



INTERFACES - CONCLUSIONES

- Una interfaz es un protocolo de comportamiento. Puede ser implementada por cualquier clase, perteneciente a cualquier jerarquía.
- Una interfaz evita establecer relaciones forzadas entre clases para compartir una superclase abstracta.
- Las interfaces no deben crecer. Si se cambia el comportamiento de una interfaz, todas las clases que la implementan fallarán.
- Una clase puede implementar más de una interfaz.
- Java provee herencia múltiple de interfaces.
- El mayor beneficio obtenido es potenciar polimorfismo en un sistema tipado.



INTERFACES

```
interface ColPrimarios{  
    int ROJO=1, VERDE=2, AZUL=4;  
}
```

```
interface TodosLosColores  
    extends Collmpresion, ColArcolris {  
    int FUCSIA=17, BORDO=ROJO+90;  
}
```

```
interface ColArcolris extends  
    ColPrimarios {  
    int AMARILLO=3, NARANJA=5,  
    INDIGO=6, VIOLETA=7;  
    public void unMetodo():  
}
```

```
interface Collmpresion extends  
    ColPrimarios {  
    int AMARILLO=8, CYAN=16,  
    MAGENTA=32;  
    public void unMetodo();  
}
```

```
public class MisColores implements  
    Collmpresion, ColArcolris {
```

```
    public MisColores() {  
        int unColor = AMARILLO;  
    }  
    public void unMetodo(){  
        //algo  
    }  
}
```

OK

No se
especifica cual
constante se
utiliza

