

## EJERCICIO EXTRA - INTERFACES

1. Partiendo de las interfaces siguientes:

```
public interface TelevisorPorControlRemoto {  
    void encender();  
    void apagar();  
    void setCanal(int numCanal);  
    void subirCanal();  
    void bajarCanal();  
    void subirVolumen();  
    void bajarVolumen();  
}  
  
public interface MandoUniversalTV {  
    void vincular(TelevisorPorControlRemoto tele);  
    void desvincular();  
    void encenderTV();  
    void apagarTV();  
    void setCanalTV (int numCanal);  
    void subirCanalTV ();  
    void bajarCanalTV ();  
    void subirVolumenTV ();  
    void bajarVolumenTV ();  
}
```

Debes crear una clase **Televisor4K** que implementa la interfaz *TelevisorPorControlRemoto* y que tenga las propiedades necesarias para poder simular los comportamientos especificados en la interfaz. Además cada método imprimirá un mensaje en la consola del tipo: “Televisor4K sube el volumen”, “Televisor4K se apaga”...

Crea también otra clase **TelevisorAntiguo** que implementa la interfaz *TelevisorPorControlRemoto* y que tenga las propiedades necesarias para poder simular los comportamientos especificados en la interfaz. Además cada método imprimirá un mensaje en la consola del tipo: “TelevisorAntiguo sube el volumen”, “TelevisorAntiguo se apaga”...

También debes crear una clase **MandoUniversalChino** que implementa la interfaz *MandoUniversalTV* y que guardará una propiedad de tipo referencia a interfaz *TelevisorPorControlRemoto*. En esta referencia guardaremos el objeto que se reciba en el método *vincular*. Con el método *desvincular* se “olvidará” la TV que se estaba controlando remotamente.

El método *encenderTV* comprobará si hay una TV vinculada y le “derivará” la tarea a realizar

llamando al método *encender* de la TV. Si no hubiera ninguna TV vinculada se debe mostrar el mensaje “ERROR: no hay ninguna TV vinculada”.

El resto de los métodos seguirán la misma “filosofía” que acabamos de describir para el método *encenderTV*.

Por último, crea una clase *PruebaMandoTV* que cree un objeto *Televisor4K*, otro *TelevisorAntiguo* y otro *MandoUniversalChino*. A continuación, vincularemos el mando con el televisor antiguo y probaremos algunos métodos, después lo desvincularemos y lo vincularemos con la televisión 4K y volveremos a probar algunos métodos.

2. Para entender bien este ejercicio debes saber que para comunicar dos aplicaciones de dos ordenadores en Internet se utilizan dos valores:

- Dirección IP: identifica a un ordenador en Internet
- Puerto: identifica a una aplicación concreta dentro de un ordenador.

De este modo la pareja IP:Puerto nos identifica una aplicación de un ordenador cualquiera conectado a Internet.

Teclea la siguiente interfaz:

```
public interface InterfazRed {  
    void setDireccionLocal(String ip);  
    String getDireccionLocal ();  
    void setDireccionRemota (String ip);  
    String getDireccionRemota ();  
    void setPuertoLocal (int puerto);  
    int getPuertoLocal();  
    void setPuertoRemoto(int puerto);  
    int getPuertoRemoto ();  
    void abrirConexionRemota();  
    void cerrarConexionRemota();  
    void enviar (InterfazFichero fich);  
    InterfazFichero recibir();  
}
```

donde *InterfazFichero* se corresponde con la realizada en el ejercicio 14 del boletín.

Además, crea una clase que implemente *InterfazRed* sabiendo que:

- Para poder abrir una conexión remota antes deben estar establecidos los parámetros, tanto locales como remotos, de la dirección IP y el puerto por el que se va a realizar la transmisión.
- El método *enviar* debe comprobar que la conexión está abierta, emitiendo un mensaje de error en caso contrario. A continuación, se enviará el fichero y se

imprimirá en la pantalla el siguiente mensaje: “Se han enviado xx bytes del fichero *nombre* desde *direccionLocal:puertoLocal* a *direccionRemota:puertoRemoto*”

- El método *recibir* debe comprobar que la conexión está abierta, emitiendo un mensaje de error en caso contrario. A continuación, se recibirá el fichero y se imprimirá en la pantalla el siguiente mensaje: “Se han recibido xx bytes del fichero *nombre* desde *direccionRemota:puertoRemoto* a *direccionLocal:puertoLocal*”.

Por último, crea una clase *PruebaInterfazRed* que tenga un método *main* que cree los objetos necesarios para probar todos los métodos de la clase que has creado.