

Vamos a modificar el ejercicio de DABRadioCD, lo tenéis resuelto en los ejercicios de Interfaces del punto anterior, para que cumpla alguno de los principios SOLID que vimos los dos últimos días. Sobre todo el de segregación de interfaces y el de inversión de dependencia. Lo que tenéis que hacer:

1. Leer los dos últimos principios, como mínimo. Para recordar lo que ya os explicamos en clase.
2. Os descargáis el .cs con el código del DABRadioCd, para que todos comencéis desde el mismo ejemplo. **Pero tendréis que pasar cada clase a un .cs nuevo.**
3. Tenéis el diagrama de clases, será la solución a la que tenéis que llegar. Como podéis ver hay una clase nueva que sirve de Factoría (Será la clase que servirá para inyectar las dependencias) y os la podéis dejar para el final.
4. Las Interfaces se han segregado, de forma que si eres un medio al que se le pueda insertar o extraer información (CD, USB, etc), habrá que heredar de IExtraibleMedia que a su vez hereda de IMedia
5. IExtraibleMedia es una interfaz genérica, de forma que cuando yo herede de ella deberemos indicar que tipo somos (Disc en este primer caso)
6. Uno de los cambios importantes, será que en la clase DABRADIOCD, ahora le llegará toda la información de tipo IMedia (Cumpliendo el principio de Inversión de dependencia), y además vamos a generalizar de forma que en vez de tener dos variables radioCD y dabRadio (como pone en el diagrama) **tendremos un array de IMedia, y el activeDevice** (este último como siempre), esto será así para poder crearnos DAB con distinto tipo y cantidad de elementos.

Vale, con esto tendréis que modificar el ejercicio para que se quiten todos los errores y que funcione.

Recordar que para crear un DABRadioCD utilizaremos la clase Factory, dentro de ella se creará un array de tipo IMedia (con un cd y una radio) y este array será el que se inyecte a la DABRadioCD.

7. Probar que funcione con el programa que ya existe, pero usando la Factory
8. Lo último será añadir una clase USBPlayer, y probar su funcionamiento con un DABRadio con 3 elementos