

## Práctica 4 (03/12/2021) – Diseño de Patrones & Refactoring

Leer detenidamente hasta el final antes de comenzar.

En esta práctica se realizarán 3 ejercicios relacionados con diferentes patrones.

Al final de la práctica se debe entregar:

- Un documento PDF con los tres ejercicios resueltos (código copiado word + anotaciones del alumno y pasado a PDF).
- Un archivo comprimido con todos los proyectos. Habrá un proyecto por cada ejercicio.

### Detalles a tener en cuenta:

- Limpieza de código
- Buenas prácticas de programación
- Se deberán incluir **comentarios que permitan entender el código** cuando sea necesario.
- Cada clase debe ir en un archivo separado.

### Ejercicio 1

Transformar el código que se facilita del ej1, para que la clase “Cliente” pueda crear sólo 1 objeto de la clase BaseDeDatos. **Modificar sólo la clase BaseDatos y la zona modificable de ExampleCode (2 líneas marcadas).**

- ¿Qué patrón has implementado para ello? Indicarlo al inicio del ejercicio 1.

### Ejercicio 2

En el ejercicio 2 tenemos una clase Cliente (que trabaja con datos tipo ArrayList), y quiere hacer uso del código disponible en Servicio. Sin embargo, existe una incompatibilidad, ya que Servicio trabaja con datos de tipo Array. Crea una estructura tipo Adapter para solventar este problema. **No se deberá modificar el código de la clase Client, ni el de la clase Service.**

- Revisad el código de la clase ExampleCode y la de la clase Service (y su JavaDoc) para definir **el nombre de los métodos y atributos de forma consistente.**
- Cuando tengáis los métodos ya definidos, descomentad las líneas de ExampleCode donde:
  - se crea el adapter (línea 1)
  - las líneas correspondientes al “Use of Service methods from Client (uncomment when ready)” (líneas 14-15)).

### Ejercicio 3

Crea un diagrama UML para cada ejercicio. Insertarlos en el documento a entregar.