

Ejercicios sobre Patrones de Diseño

Diseño y Programación Orientada a Objetos

<http://www.info-ab.uclm.es/asignaturas/42579>

Comentarios: fmontero@info-ab.uclm.es

El siguiente trabajo pretende presentar una batería de ejercicios relacionados con la utilización de patrones de diseño, concretamente con los propuestos por [Gamma y otros, 95] en su libro *Design Patterns*. La solución a los ejercicios propuestos deberá seguir siempre la misma idea; identificación del patrón o patrones útiles para abordar el problema presentado, justificación de la elección, diagrama de clases resaltando aquellos atributos o métodos significativos y, finalmente, implementación mediante un lenguaje con capacidades para orientación a objetos, preferiblemente en Java.

Patrones de Creación

Dentro de esta sección se propondrán ejercicios relacionados con la utilización de patrones de creación. Los patrones de creación propuestos por Gamma y otros son: Abstract Factory, Builder, Factory Method, Prototype y Singleton. Una descripción de los mismos puede encontrarse en la página web y en las transparencias utilizadas en las clases teóricas de la asignatura.

1. Los Juegos Reunidos Geyper, elegidos como el sexto juguete más significativo del siglo XX según la encuesta realizada entre la distribución por Juguetes y Juegos de España, vuelven a España lanzados por la firma Bizak, disponibles en cajas de 25, 35, 45 y 55 juegos. La firma pretende desarrollar una versión virtual que contemple los mismos juegos. Los Juegos Reunidos Geyper, conocidos por millones de españoles, reúnen los juegos de mesa clásicos y más entretenidos, con una imagen renovada y una magnífica presentación.



2 Diseño y Programación Orientada a Objetos

Plantéese el desarrollo del juego virtual mediante la utilización de patrones de diseño. Téngase en cuenta que lo que se pretende desarrollar son diferentes juegos de mesa que comparten fichas, dados, cartas, ruletas, de diferentes tipos que están disponibles en el juego físico.

2. Se está desarrollando una aplicación de edición de textos que será utilizada tanto en Estados Unidos como en Europa. Como se sabe la construcción de las fechas es diferente según donde nos encontremos, en Estados Unidos es mes / día / año, mientras que en Europa es día / mes / año. Desarrollar utilizando patrones de diseño los generadores de fecha para que puedan confeccionarse documentos para uno y otro destino.
3. Siguiendo con los juguetes, queremos construir tres clases de juguetes: Pokemon, BarbieGirl y Superman. Los mismos disponen de acciones de animación asociadas a su encendido y su apagado, estas acciones son diferentes para cada tipo de juguete. Implementétese utilizando patrones estos tres tipos de juguetes.
4. Se quiere desarrollar una aplicación donde los niños puedan dar color a unos dibujos ya prediseñados, de forma que los colores existen en un gestor de colores o paleta restringida de los mismos. El usuario podrá seleccionar entre los colores ya establecidos y, así, proceder a dar color a los dibujos antes mencionados. ¿Qué patrón utilizaría? ¿Por qué?.
5. Dibújese el diagrama de clases correspondiente al patrón Singleton. Y utilizando la sintaxis de Java escríbase una posible implementación del mismo.

Patrones Estructurales

En esta sección se presentarán ejercicios relacionados con los patrones estructurales, con ellos se proporcionan mecanismos de cómo las clases y los objetos se componen para formar estructuras mayores. Un patrón estructural de clases utiliza la herencia para componer interfaces o implementaciones, por ejemplo el patrón Adapter. Un patrón estructural de objetos describe la forma en que se componen objetos para obtener nueva funcionalidad, además se añade la flexibilidad de cambiar la composición en tiempo de ejecución, lo cual no es posible con la composición de clases estáticas, como ejemplo de este tipo de patrones se puede mencionar al patrón Composite. Ejemplos adicionales de este tipo de patrones son Bridge, Decorator, Façade, Flyweight o Proxy.

1. Dado el diagrama de clases que muestra la Figura 1, explica cada una de las entidades en él representadas y el papel que desempeña cada una de ellas.

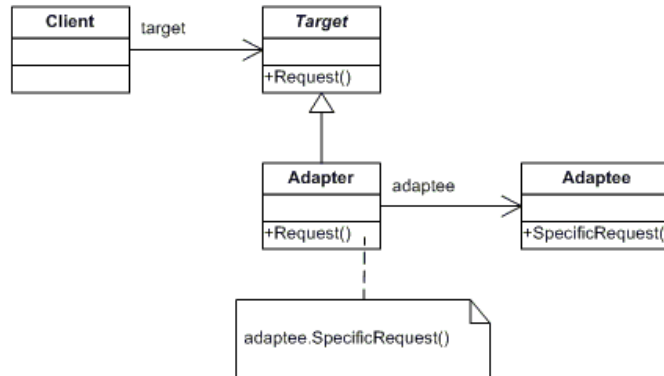


Figura 1. Diagrama de clases correspondiente al patrón Adapter

Bajo que circunstancias podrías utilizar el patrón antes representado.

2. Se quiere desarrollar una aplicación de gestión de stock de una gran superficie comercial. La misma debería proporcionar la información asociada a los productos en stock de diferentes formas, por ejemplo en ASCII o en HTML, ya que no se sabe exactamente que futura utilidad se le dará a la aplicación. Propóngase una posible implementación que de solución al problema planteado, de forma que la posibilidad de dar lugar a diferentes formatos se pueda abordar de forma no traumática para la aplicación principal.
3. Se pretende desarrollar una aplicación para la gestión del mismo stock que el ejercicio anterior. Esta vez se conoce algo más del tipo de productos que se desean gestionar, estamos en la sección de música y en ella podemos encontrar los viejos LPs de vinilo, CDs, DVDs y cassetes, también hay lotes de los productos anteriores y su gestión debería ser similar a los productos individuales. Diseñese el esqueleto de la aplicación necesario para gestionar dicho stock de tal forma que cuando alguno de los productos no tenga un número mínimo de elementos constituyendo el stock se le notifique automáticamente al proveedor.
4. Se desea crear una aplicación para la gestión de la Biblioteca de la Universidad. La misma dispone de libros y discos musicales. Entre los libros hay algunos que pueden prestarse y otros que por el contrario constituyen material de referencia y no tienen esa propiedad. Los discos de música, por el contrario, pueden prestarse. El material que es susceptible de ser prestado también puede ser reservado. Proporcionese una solución al supuesto planteado utilizando patrones de diseño.
5. Se tiene que gestionar la información perteneciente al conjunto de empleados de una empresa. Como se puede suponer alguna de esta información es confidencial y no debería estar accesible a cualquier individuo que tenga acceso a los terminales, por ejemplo, el salario de los diferentes profesionales que pertenecen a la empresa. ¿Cómo podría proporcionarse una solución a este supuesto utilizando patrones de diseño?. ¿Qué patrón utilizarías?. ¿Cómo lo implementarías?

Patrones de Comportamiento

Este tipo de patrones está relacionado con algoritmos y la asignación de responsabilidades entre objetos. Describen, además de los patrones de objetos y clases, los patrones de comunicación entre ellos. Los patrones de comportamiento de clases utilizan la herencia para distribuir el comportamiento entre las clases, como ejemplo se puede citar al patrón `Template Method`. Por su parte los patrones de comportamiento de objetos utilizan la composición de objetos en lugar de la herencia, por ejemplo como un grupo de objetos interconectados cooperan para realizar una tarea que un solo objeto no puede hacer por sí mismo, como representante de este tipo de patrones se puede mencionar al patrón `Observer`. Patrones adicionales de comportamiento son: `Chain of Responsibility`, `Command`, `Interpreter`, `Iterator`, `Memento`, `Mediator`, `State`, `Strategy` o el `Visitor`.

Seguidamente se proponen una serie de supuestos prácticos cuya solución puede verse favorecida por la utilización de patrones de diseño, concretamente por los correspondientes a los patrones de diseño de comportamiento.

1. ¿Qué patrón se está utilizando en el siguiente código?

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Test extends JFrame {
    public static void main(String args[]) {
        Test frame = new Test();
        frame.setTitle("Swing Actions");
        frame.setSize(500, 400);
        frame.setLocation(400, 200);
        frame.show();
    }
    public Test() {
        JMenuBar mb = new JMenuBar();
        JMenu fileMenu = new JMenu("File");

        fileMenu.add(new ShowDialogAction());
        fileMenu.add(new ExitAction());

        mb.add(fileMenu);
        setJMenuBar(mb);
    }
}

class ShowDialogAction extends AbstractAction {
    public ShowDialogAction() {
```

```

        super("show dialog");
    }
    public void actionPerformed(ActionEvent e) {
        JOptionPane.showMessageDialog((Component)e.getSource(),
            "An action generated this dialog");
    }
}
class ExitAction extends AbstractAction {
    public ExitAction() {
        super("exit");
    }
    public void actionPerformed(ActionEvent e) {
        System.exit(0);
    }
}

```

2. Impleméntese el código asociado a un mando de televisión que permita recorrer los canales sintonizados en el aparato receptor.
3. Simúlese el funcionamiento de una subasta mediante la utilización de patrones de diseño. El patrón que puede utilizarse para la dinámica mostrada en un proceso de subasta puede ser el *Observer*. El funcionamiento del proceso es el siguiente, se establece un precio de salida conocido por los participantes, los cuales hacen sus pujas. Las pujas serán conocidas por todos los participantes igualmente, después de un tiempo, una vez establecida la puja más alta el lote será adjudicado a esa puja.
4. Simúlese el comportamiento de una máquina expendedora de productos alimenticios. La misma sólo posee tres estados; recepción del dinero, selección del producto y devolución del cambio en caso de que sea posible. Para ello pueden utilizarse diferentes patrones de diseño, por ejemplo el *State* o el *Strategy*.
5. Se quiere automatizar un horno de pan. Se quieren producir diferentes tipos de pan:
 - **Pan blanco**: está hecho de la harina que contiene sólo la parte central del grano, lo que representa el 75% del grano entero.
 - **Pan integral**: está hecho de grano de trigo entero; conserva todos sus componentes.
 - **Pan negro**: se confecciona con harina de la que se ha quitado salvado y germen de trigo. Representa el 85% del grano de trigo entero.

Y el proceso de producción del pan es el siguiente:

1. Comprobar la temperatura (alrededor de 20°)
2. Preparar agua con sal
3. Preparar la levadura.

6 **Diseño y Programación Orientada a Objetos**

4. Añadir el agua y mezclar con la masa.
5. Añadir la harina y remover.
6. Amasar.
7. Dejar reposar durante horas
8. Trocear la masa
9. Dar cortes a las piezas
10. Llevar al horno.



6. Se dispone de una empresa consultora especializada en dar soluciones desde dos puntos de vista; el financiero o económico y el informático, dentro de esos dos criterios se realizan análisis de costes y análisis de riesgos. Las empresas a auditar estarían organizadas en departamentos y dentro de esos departamentos habría diferentes responsables cuyo trabajo sería el objeto de estudio (Director, Gestor, Ingeniero, etc...). Determinése que patrón de diseño podría utilizarse para representar lo más adecuadamente esta filosofía de trabajo, es decir, la asociada a la consultora.

Albacete, 3 de enero de 2003