PRESENTACIÓN

PATRONES DE DISEÑO ESTRUCTURALES

INTRODUCCIÓN

Los patrones de diseño estructurales son un conjunto de patrones que nos ayudan a organizar los objetos y clases de un sistema de software. Estos patrones se centran en cómo los objetos y clases se relacionan entre sí para formar estructuras más grandes y complejas.

TIPOS DE PATRONES ESTRUCTURALES

- ADAPTER
- BRIDGE
- COMPOSITE
- DECORATOR
- FACADE
- FLYWEIGHT
- PROXY

GRUPO 2

PATRÓN DE DISEÑO

DECORATOR

El patrón de diseño Decorator es un patrón estructural que permite añadir funcionalidades a un objeto existente dinámicamente, sin modificar su estructura básica.

¿CÓMO FUNCIONA?

El patrón Decorator se basa en la composición de objetos para añadir nuevas funcionalidades a un objeto existente. En lugar de heredar de una superclase o interfaz para añadir nuevas funcionalidades, creamos una clase Decorator que implementa la misma interfaz que el objeto a decorar. La clase Decorator contiene una instancia del objeto a decorar, y añade su propia funcionalidad antes o después de invocar los métodos del objeto a decorar.

COMPONENTES

- **Componente:** Interfaz o clase abstracta que define la interfaz para los objetos que se pueden decorar.
- **Objeto Concreto:** Clase que implementa la interfaz del componente y proporciona la funcionalidad básica.
- **Decorador:** Clase abstracta que implementa la interfaz del componente y mantiene una referencia al objeto a decorar.
- **Decorador Concreto:** Clase que extiende el decorador y añade su propia funcionalidad.

VENTAJAS

- El patrón Decorator permite añadir funcionalidades a un objeto existente dinámicamente, sin modificar su estructura básica.
- Es más flexible que la herencia, ya que permite añadir o quitar funcionalidades en tiempo de ejecución simplemente creando o eliminando objetos decoradores.
- Permite crear combinaciones complejas de funcionalidades utilizando objetos decoradores, sin tener que crear una subclase para cada posible combinación.

EJEMPLO

A continuación, veremos un ejemplo de cómo implementar el patrón de diseño Decorator en un videojuego.

GRUPO 2

PATRÓN DE DISEÑO

ADAPTER

El patrón Adapter es un patrón de diseño estructural que se utiliza para adaptar una clase existente a una interfaz requerida por otra clase.

¿CUÁNDO SE UTILIZA EL PATRÓN ADAPTER?

El patrón Adapter se utiliza cuando tenemos dos interfaces que no son compatibles, pero queremos que trabajen juntas.

¿CÓMO FUNCIONA EL PATRÓN ADAPTER?

El Adapter actúa como un intermediario entre las dos interfaces, tomando una interfaz y convirtiéndola en otra que la clase receptora espera.

ADAPTER

El patrón Adapter es útil cuando queremos adaptar una clase existente a una interfaz requerida por otra clase. En Java, esto se puede implementar utilizando una clase que implementa la interfaz requerida y utiliza un objeto de la clase existente para proporcionar los métodos necesarios.

EJEMPLO

A continuación, vamos a ver un ejemplo de cómo implementar el patrón de diseño Adapter.