

README

Juan Aurelio Juarez Ubaldo - 421095568
Sergio Mejia Caballero - 315111149

1 Teoria

1. Menciona los principios de diseño esenciales de los patrones Decorator y Adapter.
2. Menciona una desventaja de cada patron.
 - Decorator.
El patron Decorator nos permite extender facilmente el comportamiento de una clase sin modificar el codigo existente. La desventaja de decorator es que es dificil implementar un Decorator de tal manera que el comportamiento no dependa del orden de encapsulamiento o "wrappers".
 - Adapter.
El patron Adapter, nos permite la colaboracion entre objetos con interfaces incompatibles. La desventaja de este patron es que debemos introducir varias clases e interfaces nuevas, por lo que podria ser mas sencillo cambiar la clase para que corresponda con el resto del codigo.

2 README

Para esta practica, simulamos el lugar WaySub, el cual tiene baguettes que los clientes pueden armar a su gusto con una seleccion de ingredientes. Para esta parte, usamos el patron Decorator para poder añadir y quitar ingredientes facilmente sin necesidad de modificar codigo ya existente. Despues, se nos pide que el lugar tambien venda pizzas de un menu de 5 pizzas, las cuales ya estan establecidas con ciertos metodos, pero que el ticket sea igual. Para esta parte, usamos el Patron Adapter para crear una clase PizzaAdapter que nos permita recrear el comportamiento del ticket de una Baguette utilizando los metodos ya existentes en Pizza.