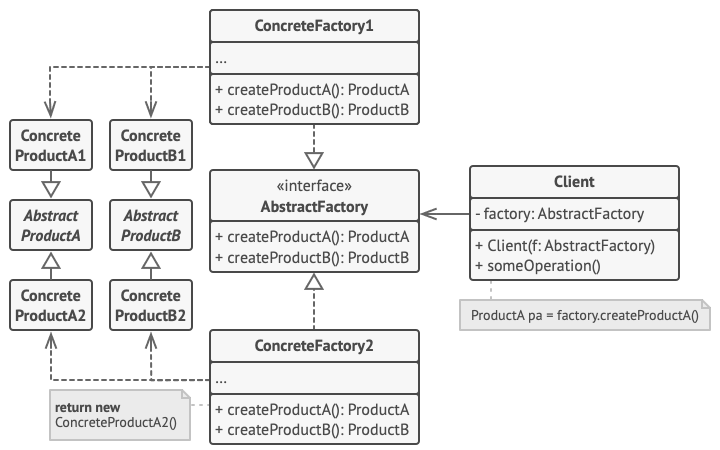
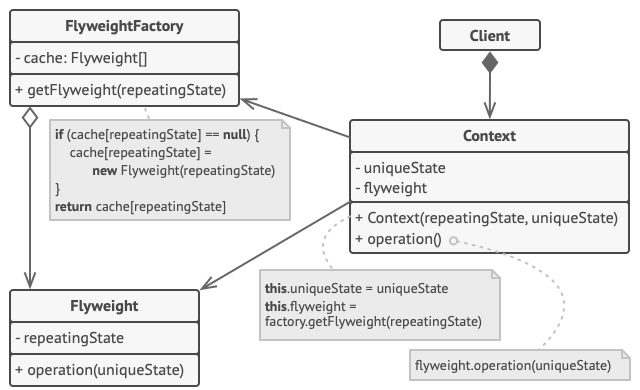
UML PATRON CREACIONAL ABSTRACT FACTORY

Fábrica Abstracta es un patrón de diseño creacional que le permite producir familias de objetos relacionados sin especificar sus clases concretas.



Este código implementa un patrón Abstract Factory para crear diferentes tipos de triángulos (en este caso, Triángulos Equiláteros y Triángulos Rectángulos). Cada fábrica concreta (TrianguloEquilateroFactory y TrianguloRectanguloFactory) crea su tipo correspondiente de triángulo (TrianguloEquilatero y TrianguloRectangulo). Luego, la función mostrar\_informacion\_triangulo se utiliza para imprimir información sobre cada triángulo creado.

UML PATRON ESTRUCTURAL FLYWEIGHT O PESO MOSCA



En este ejemplo, la clase NaveEspacial representa una nave espacial genérica con un tipo específico (como "Enemigo1" o "Enemigo2"). La clase FabricaNaves actúa como una fábrica Flyweight que almacena instancias de naves creadas previamente en un diccionario para su reutilización.

La clase Videojuego representa el videojuego en sí, que puede agregar diferentes tipos de naves a la pantalla y dibujarlas en pantalla. Al agregar y dibujar varias naves del mismo tipo, el ejemplo demuestra cómo se reutilizan las mismas instancias de naves, lo que reduce la cantidad de memoria utilizada y mejora la eficiencia en la gestión de recursos en el videojuego.

Diagrama

Descripción generada automáticamenteUML PATRON TEMPLATE METHOD

El patrón Template Method se utiliza para definir el esqueleto de un algoritmo en una clase base, permitiendo que las subclases implementen pasos específicos del algoritmo según sea necesario.

Supongamos que tenemos una clase base Juego que define un esqueleto para un juego de mesa genérico. Las subclases pueden implementar pasos específicos del juego, como inicialización, jugada y finalización.

En este ejemplo, Juego es una clase base abstracta que define el método jugar, que es el método Template Method. Este método llama a otros métodos abstractos (inicializar, jugar\_turno, finalizar), que deben ser implementados por las subclases concretes.

Las subclases Ajedrez y Damas implementan estos métodos según sea necesario para el juego específico. Al llamar al método jugar en cada una de estas subclases, se ejecuta el algoritmo completo de juego, utilizando los métodos definidos en la clase base.