

Patrón flyweight

DigitalHouse >
Coding School



**Certified Tech
Developer**
The Ultimate Degree

“Imaginemos que tenemos el sistema, una clase con pocos atributos, y que necesitamos instanciar una gran cantidad de veces. El patrón flyweight nos va a permitir tener menos instancias y nos dará la posibilidad de decorar.”



Propósito y solución

Propósito



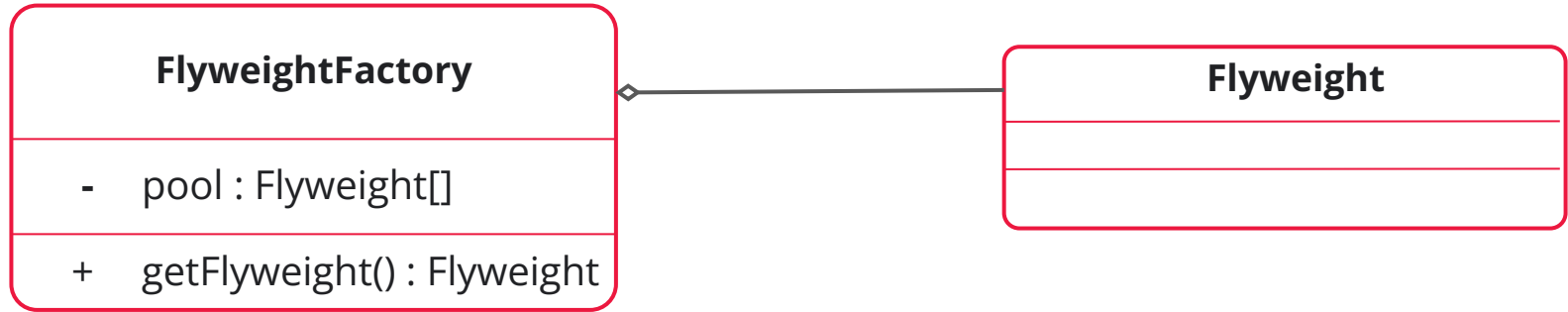
El patrón logra lo dicho anteriormente compartiendo partes del estado del objeto entre varios objetos. Es decir, abstrae las partes reutilizables y, en lugar de crear objetos cada vez que sea requerido, podemos reutilizar objetos creados por otras instancias. Esto permite reducir la capacidad de memoria requerida por la aplicación.

Solución



Este patrón cuenta con varios componentes: el cliente es el objeto que dispara la ejecución. El FlyweightFactory es la fábrica que utilizaremos para crear los objetos flyweight u objetos ligeros. El flyweight corresponde a los objetos que deseamos reutilizar para que estos sean más ligeros.

Diagrama UML



Ventajas y desventajas

Ventaja



Reduce una gran cantidad el peso de los datos en el servidor.

Desventaja



Consume un poco más de tiempo para realizar las búsquedas.

DigitalHouse>
Coding School