



Wzorzec „Dekorator”

Dekorator

- Pozwala na dodanie nowej funkcjonalności do istniejących klas dynamicznie podczas działania programu.
- Wzorzec dekoratora polega na opakowaniu oryginalnej klasy w nową klasę "dekorującą". Zwykle przekazuje się oryginalny obiekt jako parametr konstruktora dekoratora, metody dekoratora wywołują metody oryginalnego obiektu i dodatkowo implementują nową funkcjonalność.

Dekorator

- Dekoratory są alternatywą dla dziedziczenia. Dziedziczenie rozszerza zachowanie klasy w trakcie kompilacji, w przeciwieństwie do dekoratorów, które rozszerzają klasy w czasie działania programu.
- Ponieważ w większości obiektowych języków programowania nie można tworzyć nowych klas podczas działania programu i zwykle nie można przewidzieć z góry wszystkich kombinacji rozszerzeń klas, konieczne byłoby stworzenie nowej klasy dla każdej kombinacji. Dekoratory są obiektami, tworzonymi w czasie działania programu, i mogą być łączone w różne kombinacje bezpośrednio przy użyciu. Przykładem wzorca dekoratora jest implementacja strumieni I/O w Javie.

Pliki

- FileReader
- FileWriter
- BufferedReader
- BufferedWriter
- FileInputStream
- InputStreamReader
- FileOutputStream
- OutputStreamWriter
- File
- PrintWriter

Używają
systemowego
kodowania.

Szybsze, używają
buforowania.

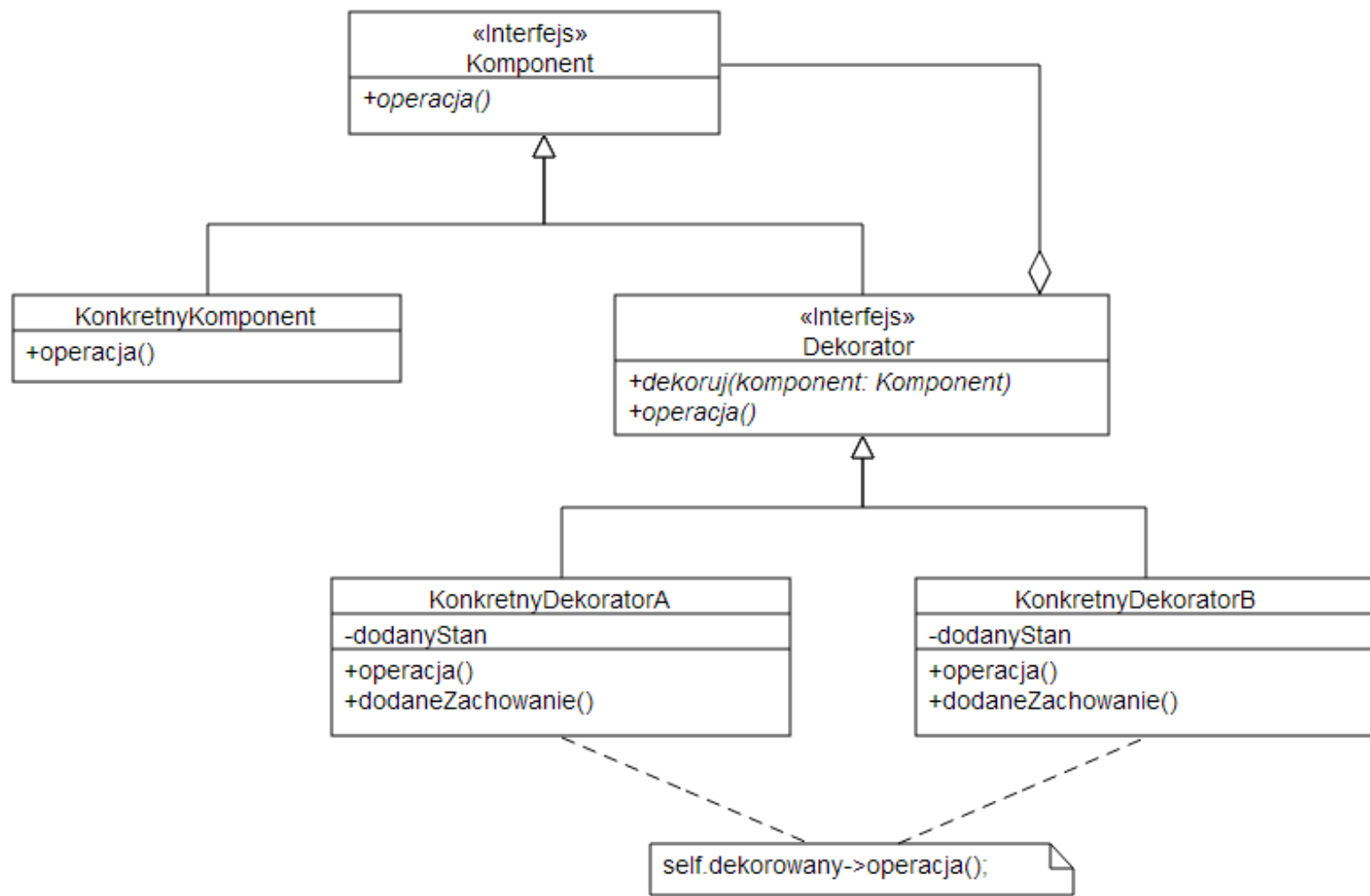
Można ustawić swoje własne
kodowanie, funkcje te są
„łańcuchowane”.

Dekorator w praktyce

```
BufferedReader in = new BufferedReader(new  
InputStreamReader(„plik.txt“));
```

plik.txt jest odczytywany przez `InputStreamReader`; klasa ta odczytuje surowe dane (bajtowo); przekazuje bajty do kolejnej klasy – `BufferedReader`, która konwertuje bajty na znaki, a do tego – buforuje wejście

Dekorator - schemat



[http://pl.wikisource.org/wiki/Dekorator \(wzorzec projektowy\)/kod](http://pl.wikisource.org/wiki/Dekorator_(wzorzec_projektowy)/kod)

Ćwiczenie 1

1. Napisać program, który pozwoli obliczyć liczbę wierszy tekstu. Przeglądane powinny być wszystkie znaki tekstu, a obliczanie liczby wierszy powinno odbywać się dopiero w momencie zakończenia przeglądania danego wiersza.
2. Napisać program obliczający liczbę słów w pliku tekstowym zakładając, że poszczególne słowa są oddzielone spacjami, tabulatorami lub znakami końca linii.

Ćwiczenie 2

- Wykonaj prostą grę polegającą na symulacji zachowań drogowych
- Zdefiniuj klasy użytkowników drogi (pieszych, samochodów, rowerów).
- Zadbaj o ich dekorowanie z abstrakcyjnej klasy bazowej UżytkownikDrogi
- Dzięki dekoratorowi – Pieszy może stać się Rowerzystą (wystarczy, że wsiądzie na rower), itp.
- Każdy z obiektów danej klasy potrafi się poruszać po planszy (konsoli) z różną prędkością. Każdy z obiektów ma również swój odpowiedni znaczek.
- Obiekty poruszają się po planszy w sposób losowy, gra kończy się w momencie „kraksy”.
- Zadbaj o krokową pracę (aby można było podglądać każde stadium).