**Inżynieria oprogramowania**

**prowadzący dr inż. Tomasz Marciniak**

GRUPA 3

Mateusz Nitka

Karol Kunda

Przemysław Grochowski

1.Wzorzec projektowy:  
Uniwersalne, sprawdzone w praktyce rozwiązanie często pojawiających się, powtarzalnych problemów projektowy. Pokazuje powiązania i zależności pomiędzy klasami oraz obiektami i ułatwia tworzenie, modyfikację oraz pielęgnację kodu źródłowego. Stosowanie wzorców tworzy kod znacznie czytelniejszym, przejrzystszym, łatwiejszym do modyfikacji zarówno przez siebie, jak i innego programistę, który miałby za zadanie rozwijać projekt. Dzięki nim nie jesteśmy zmuszani wynajdować na nowo koła i samodzielnie rozwiązywać postawionego przed nami zadania.

2. Wzorce przykłady:

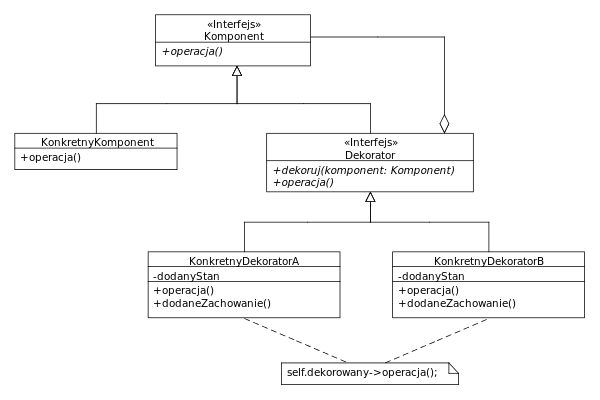
konstrukcyjne:  
tworzenie obiektów, w tym: delegowanie procesu tworzenia do innych klas (ważne ze względu na zmniejszanie współzalezności w kodzie), kontrola nad sposobem tworzenia obiektów  
  
strukturalne:  
zarządzanie strukturą obiektów i strukturami złożonymi z obiektów  
  
behawioralne:  
zachowanie obiektów i komunikacja między nimi

Wzorzec projektowy „Dekorator”:

**Dekorator** – wzorzec projektowy należący do grupy *wzorców strukturalnych*. Pozwala na dodanie nowej funkcji 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 funkcję.

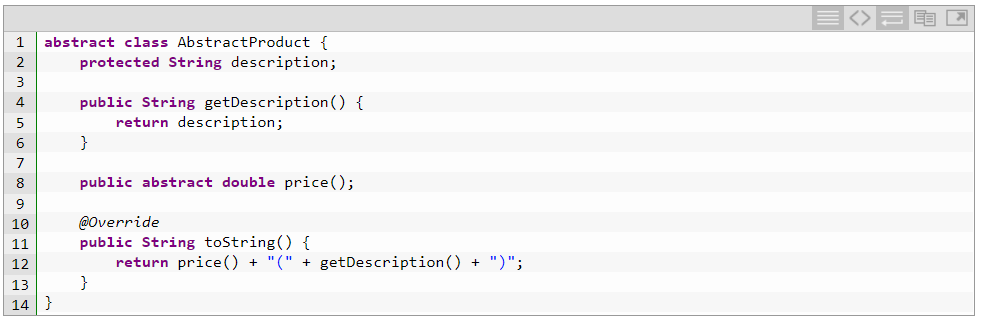
Dekoratory są alternatywą dla dziedziczenia. Dziedziczenie rozszerza zachowanie klasy w trakcie kompilacji, w przeciwieństwie do dekoratorów, które rozszerzają klasy w czasie działania programu.

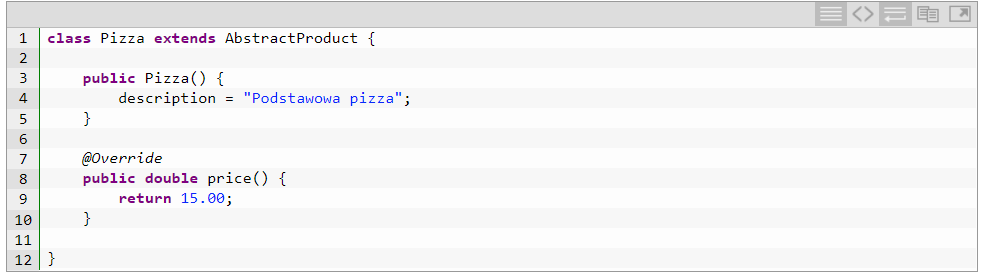


Korzyści z stosowania Dekoratora:

1. Zapewnia większą elastyczność niż statyczne dziedziczenie.
2. Pozwala uniknąć tworzenia przeładowanych funkcjami klas na wysokich poziomach hierarchii.
3. Dekorator i powiązany z nim komponent nie są identyczne.
4. Powstawanie wielu małych obiektów.

Przykładowa implementacja wzorca projektowego:  
  
Dekoratory można wykorzystać np. do oprogramowania kalkulacji ceny pizzy. Definiujemy klasę abstrakcyjnego produktu:



I tworzymy nasz produkt którym będzie pizza:  


Teraz możemy utworzyć abstrakcyjnego dekoratora:

