

WYKŁAD 3

Wzorce projektowe - klasyfikacja

Uodpornianie kodu źródłowego na jego zmiany

Jedną z głównych trudności i wyzwań jest umiejętność przewidywania zmian, które mogą zostać w przyszłości wprowadzone. Do uodpornienia kodu na zmiany stosuje się wzorce projektowe.

Wzorce projektowe

Mogą one być używane do:

- rozwiązywania często spotykanych problemów
- uodporniania kodu na zmiany

Wzorce projektowe

Jedną z istotnych trudności przy wprowadzaniu zmian są rozbudowane hierarchie dziedziczenia.

Wzorce projektowe pokazują w jaki sposób można ograniczać negatywny wpływ dziedziczenia na modyfikowalność kodu.

Wzorce projektowe

Znane są wzorce projektowe „ogólnego przeznaczenia” jak również wzorce projektowe specjalistyczne – specyficzne dla różnych dziedzin zastosowań.

Rodzaje wzorców projektowych

Klasyfikacja ogólnych wzorców projektowych:

- kreacyjne
- strukturalne
- czynnościowe

Wzorce kreacyjne - przykłady

- Builder
- *Abstract Factory*
- Factory Method
- Prototype
- Singleton

Wzorce strukturalne - przykłady

- Adapter
- Decorator
- Facade
- Composite
- *Bridge*
- Proxy
- Flyweight

Wzorce czynnościowe - przykłady

- Interpreter
- *Iterator*
- Chain of Responsibility
- Mediator
- Template Method
- Observer
- *Visitor*

Wzorce czynnościowe - przykłady

- Memento
- *Command*
- State
- *Strategy*

Objaśnienia:

underlined – wzorce najprostsze do nauczania
się i najczęściej używane

italics – wzorce przydatne w case-study

Wzorce projektowe - przykłady

Wzorce kreacyjne:

- Singleton
- Builder
- Abstract Factory

Wzorce projektowe - przykłady

Wzorce strukturalne:

- Facade
- Composite
- Bridge

Wzorce projektowe - przykłady

Wzorce czynnościowe:

- Observer
- Visitor
- State

Sposób opisywania wzorców projektowych

- nazwa wzorca i jego kategoria
- przeznaczenie
- inne nazwy
- uzasadnienie stosowania
- stosowalność
- struktura
- uczestnicy

- współpraca
- konsekwencje
- implementacja
- przykłady
- znane zastosowania
- pokrewne wzorce

Przestrzeń wzorców projektowych

Rodzaj				
		Kreacyjne	Strukturalne	Czynnościowe
Zakres	Klasy	Factory Method	Adaptor (klasy)	Interpreter Template Method
	Obiekty	Builder Abstract Factory Prototype Singleton	Adaptor (obiekty) Decorator Facade Composite Bridge Proxy Flyweight	Iterator Chain of Responsibility Mediator Observer Visitor Memento Command State Strategy

Niezmienniczość wzorców projektowych

Rodzaj	Wzorzec projektowy	Aspekt zmienności
Wzorzec kreatywny	Builder	Sposób tworzenia obiektów złożonych
	Abstract Factory	Rodziny obiektów produktów
	Factory Method	Podklasa tworzonego obiektu
	Prototype	Klasa tworzonego obiektu
	Singleton	Jedyny egzemplarz klasy

Niezmienniczość wzorców projektowych

Rodzaj	Wzorzec projektowy	Aspekt zmienności
Wzorzec strukturalny	Adaptor	Interfejs obiektu
	Decorator	Zobowiązania obiektu bez definiowania podklas
	Facade	Interfejs podsystemu
	Composite	Struktura i układ obiektu
	Bridge	Implementacja obiektu
	Proxy	Sposób dostępu do obiektu
	Flyweight	Koszt przechowywania obiektu

Rodzaj	Wzorzec projektowy	Aspekt zmienności
Wzorzec czynnościowy	Interpreter	Gramatyka i interpretacja języka
	Iterator	Sposób dostępu i przechodzenia elementów agregatu
	Chain of Responsibility	Obiekt, który może zrealizować żądanie
	Mediator	Jak i które obiekty oddziałują na siebie
	Template Method	Kroki algorytmu
	Observer	Ilość obiektów zależących od innego obiektu; jak zależne obiekty utrzymują aktualny stan
	Visitor	Operacje, które można zastosować do obiektu (obiektów) bez zmiany jego klasy (ich klas)
	Memento	Jakie prywatne informacje są przechowywane poza obiektem i kiedy
	Command	Warunki i sposób realizacji żądania
	State	Stany obiektu
	Strategy	Algorytm