

Programowanie obiektowe

Laboratorium 9 - krótkie przypomnienie diagramu klas

mgr inż. Krzysztof Szwarc

krzysztof@szwarc.net.pl

Sosnowiec, 10 maja 2017

Czym jest diagram klas?

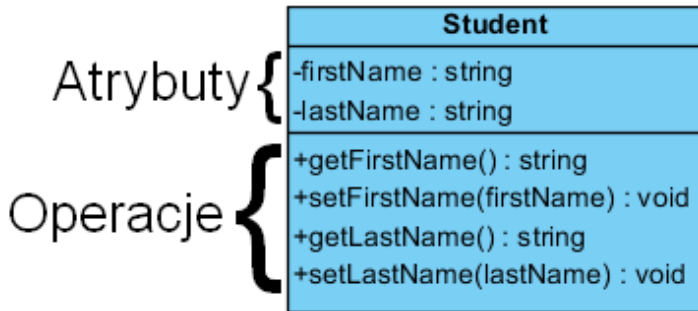
Diagram klas

Diagram klas jest statycznym diagramem strukturalnym przedstawiającym strukturę systemu w modelach obiektowych. Ilustruje on strukturę klas oraz zależności zachodzące między nimi.

Klasy reprezentowane są za pomocą prostokątów zawierających nazwę, atrybuty oraz operacje (metoda to implementacja operacji). W diagramie klas występują następujące symbole reprezentujące poziomy widoczności atrybutów i operacji:

- + (dostęp publiczny).
- # (dostęp chroniony).
- - (dostęp prywatny).
- ~ (dostęp w pakiecie).

Przykład klasy w notacji UML



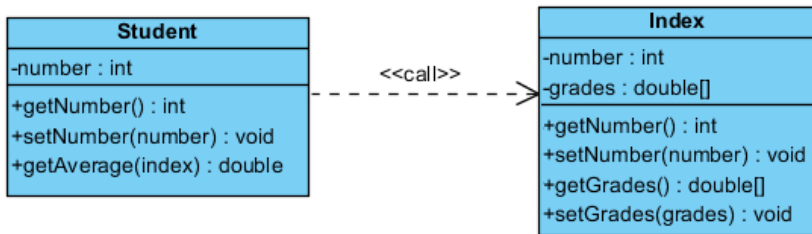
Rodzaje związków w diagramie klas

- Zależność.
- Asocjacja.
- Agregacja.
- Kompozycja.
- Generalizacja.

- 1 (jeden obiekt).
- * (dowolna liczba obiektów).
- 0..4 (od zera do czterech obiektów).
- 2..* (od dwóch do dowolnej liczby obiektów).

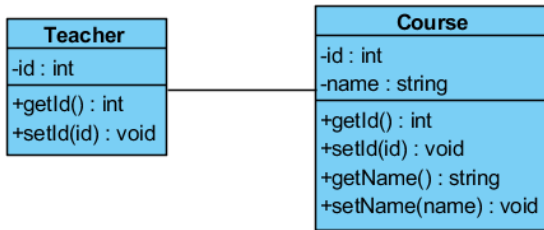
Zależność

Zależność (ang. dependency) jest związkiem występującym między dwoma elementami zakładającym, że jedna klasa wykorzystuje drugą lub wie o jej istnieniu i zmiana w jednym elemencie może (ale nie musi) wymusić zmiany w drugim elemencie. W przykładzie, do wyznaczenia średniej ocen, operacja klasy Student korzysta z operacji klasy Index.



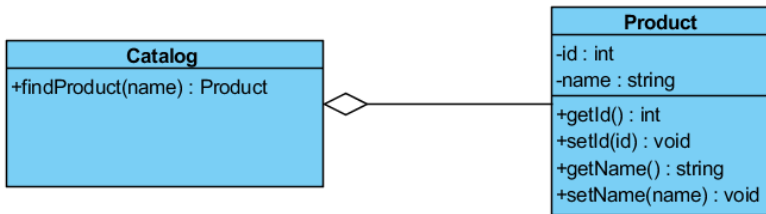
Asocjacja

Asocjacja (ang. association) przedstawia czasowe powiązanie między niezależnymi obiektami dwóch klas (żaden obiekt nie jest właścicielem drugiego). Są silniejszymi relacjami niż zależności. W przykładzie obiekt klasy Teacher przez pewien czas jest związany z obiektem klasy Course.



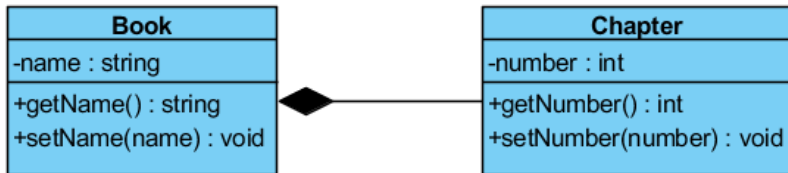
Agregacja

Agregacja (ang. aggregation) jest silniejszą relacją niż asocjacja i reprezentuje relację typu całość-część (część może należeć do kilku całości, ale całość nie zarządza czasem istnienia części - występuje właściciel i obiekt podrzędny, jednakże obiekt podrzędny może mieć wielu właścicieli). W przykładzie obiekt klasy Catalog zawiera obiekty klasy Product, ale nie tworzy ich (nie jest ich wyłącznym właścicielem).



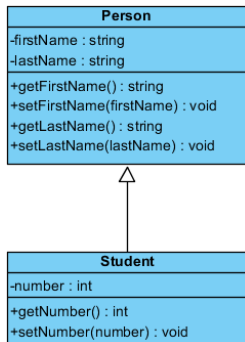
Kompozycja

Kompozycja (ang. composition) jest silniejszą relacją niż agregacja i reprezentuje relację typu całość-część, w której całość jest jedynym właścicielem części, tworzy je oraz zarządza nimi. Po usunięciu klasy głównej usunięta zostanie również klasa częściowa. W przykładzie obiekt klasy Book zawiera obiekty klasy Chapter, a same instance obu klas nie mogą istnieć bez siebie.



Generalizacja

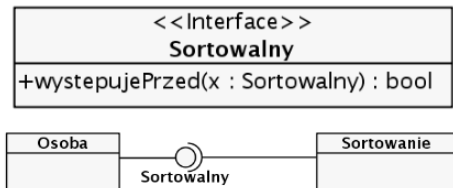
Generalizacja (ang. generalization) tworzy hierarchię klas wydzielając ich części wspólne. W przykładzie klasa Person jest uogólnieniem klasy Student.



Interfejsy w diagramie klas

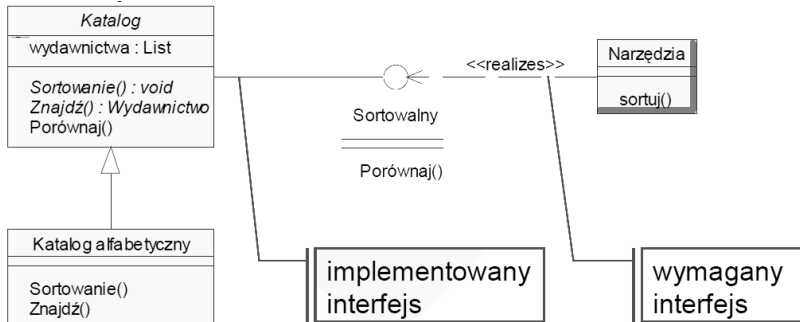
Interfejsy

Interfejs (ang. interface) jest zestawem operacji wyznaczającym usługi, które oferuje dana klasa. Oznaczamy go analogicznie do klasy (wzbogacając o stereotyp <<interface >>) lub w postaci kuli.



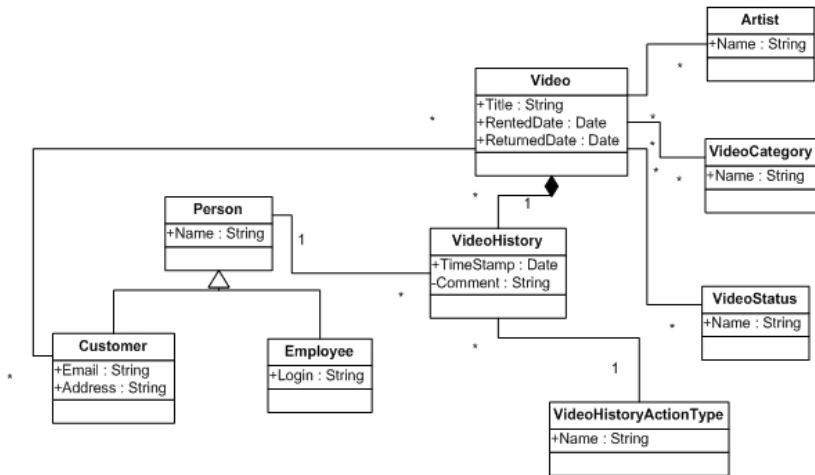
<http://zasoby.open.agh.edu.pl/>

Przykład zastosowania interfejsu



<http://zasoby.open.agh.edu.pl/>

Przykład diagramu klas



<http://zine.net.pl/>

Darmowe narzędzia do diagramów UML

- 1 Visual Paradigm Community Edition (desktopowy)
- 2 <http://argouml.tigris.org/> (desktopowy)
- 3 <https://draw.io/> (online)
- 4 <https://creately.com/> (online, wymaga rejestracji do eksportu diagramu)



<http://wykop.pl/>

- 1 Zaprojektujcie diagram klas dla zadania projektowego.

Dziękuję za uwagę