

Inżynieria oprogramowania

Wykład 11

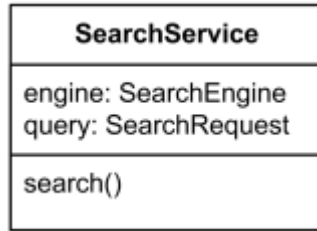
Projektowanie obiektowe: szczegóły diagramu klas

<http://www.uml-diagrams.org/>

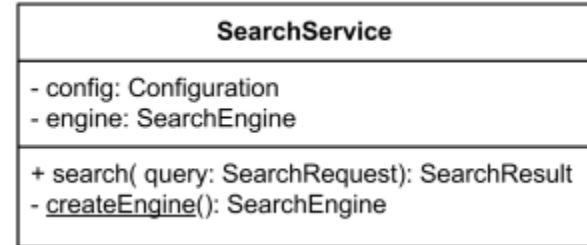
Notacja diagramu klas



Klasa „**Klient**” –
szczegóły stłumione

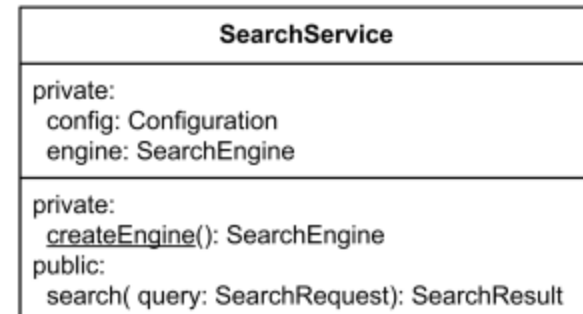


Klasa „**UsługaWyszukiwania**” –
szczegóły na poziomie
analitycznym



Usługa wyszukiwania klasa –
szczegóły na poziomie realizacji.
Operacja „utwórzSilnik()” jest statyczna.

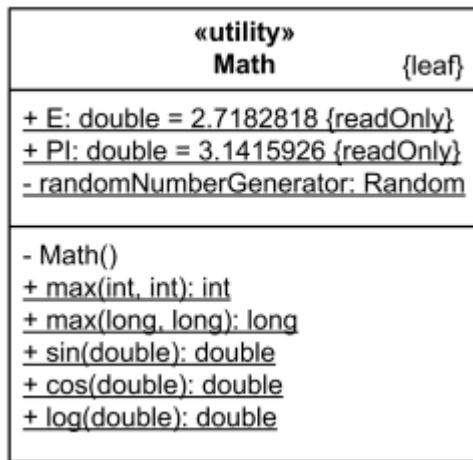
Atrybuty lub działania
mogą być zgrupowane
według widoczności.



Notacja diagramu klas, cd.

SearchRequest

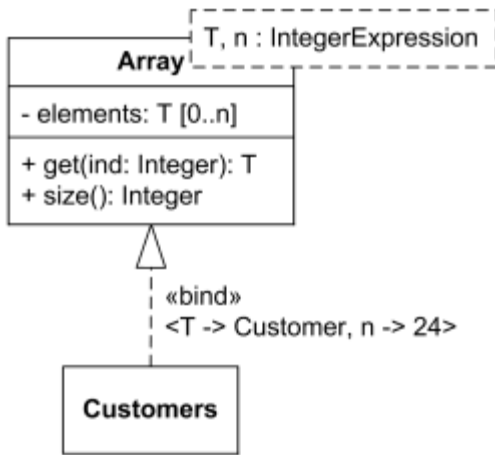
Nazwa klasy abstrakcyjnej jest pokazywana kursywą.



Klasa narzędziowa ma tylko atrybuty i operacje statyczne o zasięgu klasy. Takie klasy zwykle nie mają instancji.

Inne standardowe stereotypy klas:
<<focus>> jest klasą, która definiuje logikę podstawową lub przepływ sterowania dla jednej lub więcej wspieranych klas;
<<auxiliary>> jest klasą, która wspiera inną bardziej centralną lub podstawową klasę zazwyczaj poprzez wdrażanie wtórnej logiki lub wtórnego przepływu sterowania;
<<typ>> jest klasą, która określa domenę obiektów wraz z czynnościami mającymi zastosowanie do obiektów, bez określenia fizycznej realizacji tych obiektów.

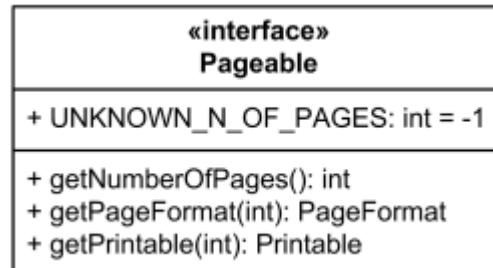
Notacja diagramu klas, cd.



Klasa-szablon **Array** i „zbindowana” klasa Klienci.



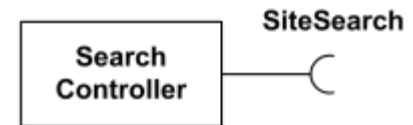
Interfejs **SiteSearch**



Interfejs jest klasyfikatorem, który deklaruje zestaw spójnych funkcji publicznych i obowiązków. Interfejs określa kontrakt.

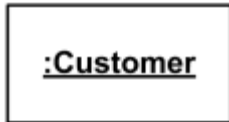


Interfejs **SiteSearch** jest realizowany (wdrażany) przez klasę **SearchService**.

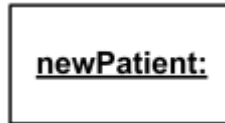


Interfejs **SiteSearch** jest używany (wymagany) przez **SearchController**.

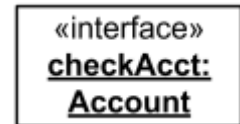
Notacja diagramu obiektów



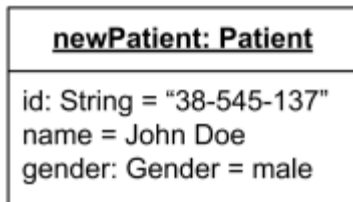
Anonimowa instancja klasy Klienta.



NewPatient instancją klasy bezimiennej lub nieznaniej.

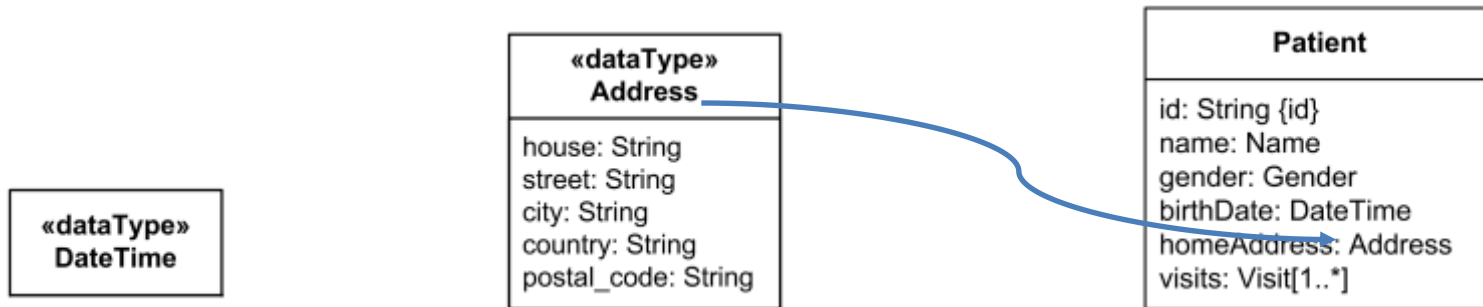


CheckAcct instancją interfejsu Account.



Instancja klasy pacjenta **newPatient** ma gniazda z określonymi wartościami.

Typ danych



Typ danych jest klasyfikatorem – podobnym do klasy – którego wystąpienia są identyfikowane jedynie przez ich wartości.



Prymitywny typ danych „**Waga**”.

«enumeration» jest typem danych, którego wartości są wyliczane w modelu jako zdefiniowane przez użytkownika literały wyliczeniowe.

Działania

SQLStatement
+executeQuery(sql: String): ResultSet #isPoolable(): Boolean ~getQueryTimeout(): int -clearWarnings()

File
+getName(): String +create(parent: String, child: String): File +listFiles(): File[0..*] -slashify(path: String, isDir: Boolean) : String

Działanie jest zachowawczą cechą klasyfikatora, który określa nazwę, typ, parametry i ograniczenia do powoływania skojarzonego zachowania.

Thread
+ setDaemon(in isDaemon: Boolean) - changeName(inout name: char[0..*]) + <u>enumerate</u> (out threads: Thread[0..*]): int + isDaemon(return: Boolean)

Identity
~ check(directive: String) { redefines status} - getPublicKey(): PublicKey {query} + getCerts(): Certificates[*] {unique, ordered}

Diagram modelu domeny

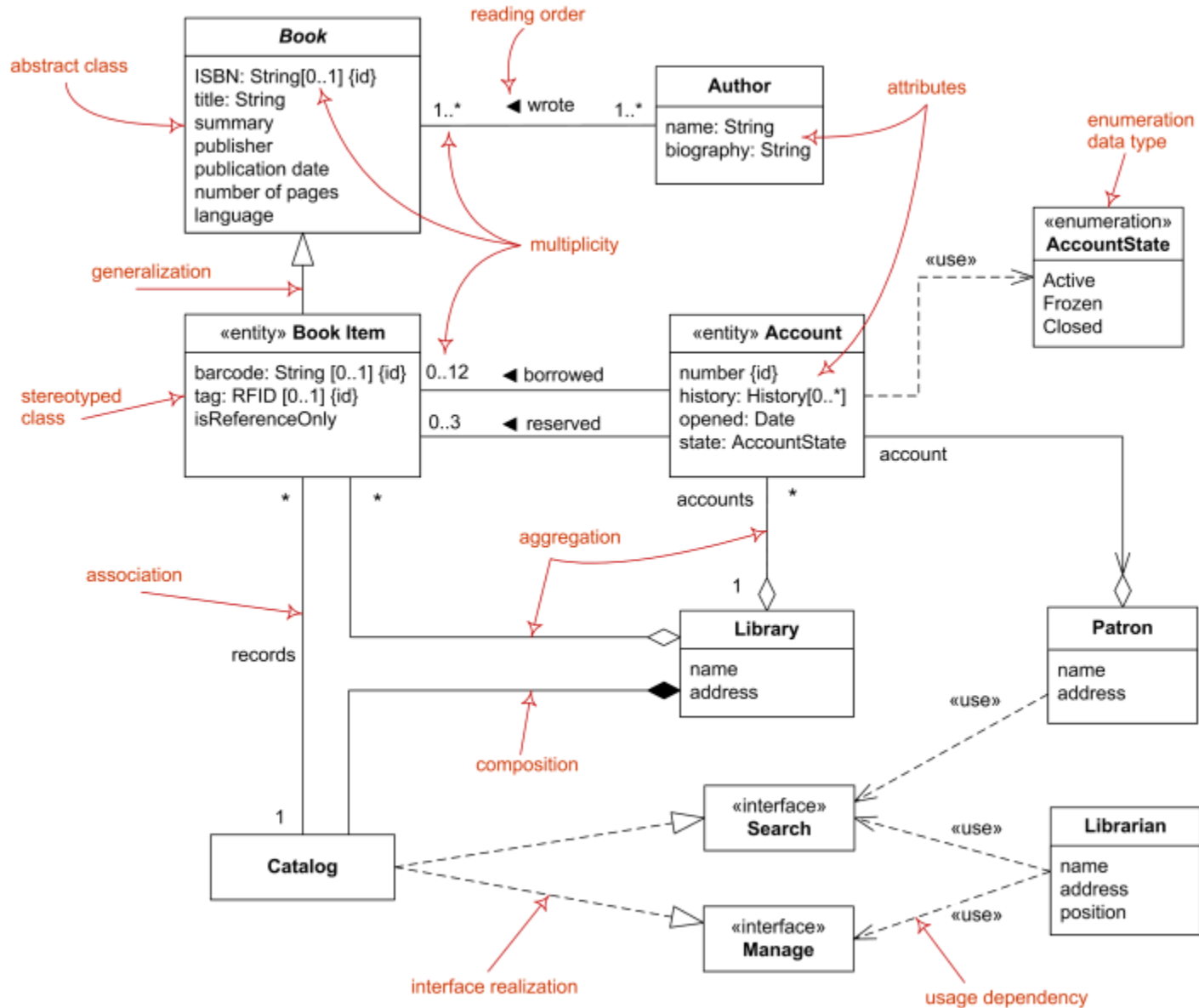


Diagram obiektów

