

Hello

Kasia Bogusławska

1. Klasy i dziedziczenie

Klasa

Wikipedia

Częściowa lub całkowita definicja dla obiektów. Definicja obejmuje dopuszczalny stan obiektów oraz ich zachowania [...] Klasy mogą być typami języka programowania - przykładowo, instancja klasy Owoc będzie mieć typ Owoc.

Tak na chłopski rozum

Klasa definiuje, **czym coś jest**: jakie składowe oraz zachowania charakteryzują dany byt.

Pamiętacie generator?

```
Sample6 generateRandomString()
package utils;

import ...

public class Sample6 {
    public static String generateFirstname(){
        int firstNameLength = 8;
        return Sample6.generateRandomString(firstNameLength);
    }

    public static String generateLastname(){
        int lastNameLength = 14;
        return Sample6.generateRandomString(lastNameLength);
    }

    private static String generateRandomString(int length){
        RandomStringGenerator generator =
            new RandomStringGenerator.Builder().withinRange('a', 'z').build();
        return StringUtils.capitalize(generator.generate(length));
    }
}
```

Części wspólne

- generateFirstName
- generateLastName
- generateStreetAddress
- generateCity
- ...

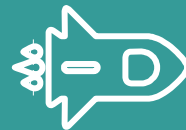
Każda z tych metod u swoich podstaw ma **generowanie losowego ciągu znaków lub cyfr**. A jakby tak zamknąć te podstawy w osobnym pudełku?

KLASA BAZOWA



Jej pola i metody mają zbierać to, co będzie charakterystyczne dla wszystkich klas pochodnych (czyli tych, które po niej dziedziczą);

ABSTRAKCJA



... pewnego rodzaju uproszczenie rozpatrywanego problemu, polegające na **ograniczeniu zakresu** cech manipulowanych obiektów wyłącznie **do cech kluczowych** dla algorytmu...

To czemu nie...

Interfejs?

Interfejs to zobowiązanie, że wszystkie klasy, które go implementują będą wystawiały pewne metody, czyli będą się w pewien sposób **zachowywały**. Każda implementująca klasa będzie dysponowała swoją wersją danego zachowania. Metody interfejsu należałoby więc implementować osobno w każdej klasie.

Klasa abstrakcyjna?

Klasa abstrakcyjna to forma wymuszenia lub poinformowania, że każda klasa pochodna **jest** tym samym, co klasa bazowa, pod względem pól i metod (czyli elementów i zachowań). Ze względu na wysoki poziom tego wymuszenia, nie podajemy dokładnie implementacji – dajemy sobie elastyczność.

“ *Simplicity and elegance are unpopular because they require hard work and discipline to achieve and education to be appreciated.*

Ale mamy wyabstarhowane...

CEL: stworzenie generatora adresów będącego pochodną naszej klasy losowego generatora

JAK:

- 1) Jakich danych potrzebujemy?
- 2) Jak je uzyskać przy wykorzystaniu mechanizmów, które właśnie stworzyliśmy?

Jakich danych potrzebujemy?

Checkout

Logon or signup to simplify the online purchase process!

BILLING INFORMATION

First name *

Last name *

Billing company

Street address *

City *

Country *

State / province *

Postal code *

Email address *

Phone number *

☐ Create an account?

SHIP TO A DIFFERENT ADDRESS? ☐

Personal information

First name *

Last name *

Country *

State / province *

Sign-in information

Email address *

Password *

Repeat password *

CREATE AN ACCOUNT

Jakich danych potrzebujemy?

	Format	Dowolność
Street Address	Liczba spacja tekst	TAK
City	tekst	TAK
Country	tekst	NIE
Postal code	Liczba myślnik liczba	TAK

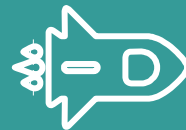
Co już mamy bez dziedziczenia?

	Format	Dowolność
Street Address	Liczba spacja tekst	TAK
City	tekst	TAK
Country	tekst	NIE
Postal code	Liczba myślnik liczba	TAK

Co już mamy z dziedziczeniem?

	Format	Dowolność
Street Address	Liczba spacja tekst	TAK
City	tekst	TAK
Country	tekst	NIE*
Postal code	Liczba myślnik liczba	TAK

HERMETYZACJA



Hermetyzacja polega na **ukrywaniu** pewnych danych składowych lub metod obiektów danej klasy tak, aby były one dostępne tylko **metodom wewnętrznym** danej klasy lub **klasom dziedziczącym**.

Po co tyle zachodu?

1

Cały podział kodu na klasy służy wydzieleniu logicznych odpowiedzialności porcji kodu.

2

Robi się to po to, by kod był łatwiejszy w utrzymaniu. W przypadku jasnego podziału odpowiedzialności, łatwo zidentyfikować miejsce wymagające zmiany/ poprawki.

3

Hermetyzacja ma na celu upewnienie się, że jak już znajdziemy to miejsce, to mamy pewność, że tylko to jedno miejsce należy zmienić/ poprawić, bo dana część kodu używana jest tylko w tym miejscu.

Jakby tu teraz tego użyć?

CEL: stworzenie modelu adresu, który pozwoli zapakować niezbędne informacje w jedną samowystarczalną całość

WYKORZYSTANIE: obiekty tej klasy będą przekazywane metodom testowym do wprowadzania danych adresowych klienta; asercje będą sprawdzały, czy dane wyświetlone w aplikacji zgadzają się z oczekiwanymi dostarczonymi w obiekcie

Jakby tu teraz tego użyć?

JAK:

- 1) Klasa ta będzie jedynie przechowywać dane: DTO – *Data Transfer Object*
- 2) Identyfikacja niezbędnych pól klasy i ich typów
- 3) Konstruktor, który wywołuje metody statyczne odpowiedniego generatora
- 4) Konstruktor z parametrami pozwalający na stworzenie obiektu z danych odczytanych z aplikacji
- 5) Prywatne pola udostępnione poprzez gettery i settery
- 6) Nadpisana metoda equals, która pozwoli porównać adresy w asercji

Dzięki