

# Klasy, obiekty i konstruktory; hermetyzacja, metody ustawiające i zwracające

Karol Kulinowski, Agnieszka Kazimierska

19 kwietnia 2023

## Klasy

**Klasa** - szablon składający się z pól oraz metod, na podstawie którego można tworzyć obiekty (instancje) tej klasy.

- **Pola** – zmienne należące do danej klasy, służą do przechowywania informacji o obiektach.
- **Metody** - „funkcje” wewnątrz danej klasy opisujące zachowanie obiektów (instancji) danej klasy.
- **Konstruktory** - mają taką samą nazwę jak klasa, służą do tworzenia obiektów danej klasy oraz inicjalizacji pól. Wywoływany przez użycie komendy **new**.
- Klasę należy umieścić w pliku o tej samej nazwie, na plik powinna przypadać jedna klasa.
- Dla każdej klasy należy utworzyć klasę „demo” która ją przetestuje. Powinna ona zawierać wyłącznie metodę „main”.
- Modyfikatory dostępu:
  - **Private** - dostępny tylko wewnątrz danej klasy, dostęp wyłącznie przez metody z modyfikatorem *public*.
  - **Public** - dostępny dla całego świata, wszyscy mają dostęp do klas, pól lub metod z tym modyfikatorem.
  - *Domyślny (package-private)* - dostępny w ramach pakietu.
- Do prywatnych elementów klas można uzyskać dostęp przez napisanie tzw. *setterów* i *getterów*, metod umożliwiających ustawianie wartości oraz ich zwracanie. Ich zastosowanie pozwala zabezpieczyć się przed zapisaniem niewłaściwej wartości za pomocą kontroli wyjątków (patrz metoda `setWiek` w przykładzie z klasą `Pies`). Metody te powinny mieć nazwę `setPole(...)` i `getPole()`.

**Pakiety** - struktury umożliwiające grupowanie związanych ze sobą klas.

- Kasę można dodać do pakietu poprzez zapisanie pliku `.java` w folderze o odpowiedniej nazwie, oraz dodanie komendy `package nazwaPakietu;`, przy czym nazwa folderu musi odpowiadać nazwie pakietu.

## Schemat tworzenia klas: klasa NazwaKlasy

```
1 package nazwaPakietu; // Nazwa pakietu, którego klasa jest częścią
2 // Uwaga: plik z klasą musi się znajdować w folderze "nazwaPakietu"
3
4 // modyfikator_dostępu class NazwaKlasy {...}
5 public class NazwaKlasy {
6     //
7     // Pola klasy
8     // modyfikator_dostępu typ nazwa_pola;
9     private int pole_1; // pole prywatne
10    public double pole_2; // pole publiczne
11    // ...
12    String pole_n; // Brak modyfikatora -> modyfikator domyślny (package-
private)
13    //
14    // Konstruktory
15    // modyfikator_dostępu NazwaKlasy(parametry) {...}
16    public NazwaKlasy() {...} // Konstruktor domyślny
17    public NazwaKlasy(int x, ...) {pole_1 = x; ...} // Konstruktor z
parametrami
18    //
19    // Metody
20    // modyfikator_dostępu typ nazwa_metody(parametry) {...}
21    private void metoda_1() {...} // metoda prywatna
22    public int metoda_2(...) {...} // metoda publiczna
23    float metoda_3(...) {...} // metoda z modyfikatorem domyślnym
24    // ...
25 }
```

  

```
1 // Klasa Main tworząca obiekt klasy NazwaKlasy oraz testująca jej metody
2 // Klasa ta już nie jest częścią pakietu, nie będzie więc ona widzieć
3 //      pól/metod z modyfikatorem domyślnym (package-private)
4 public class Main {
5     static public main(String[] args) {
6         NazwaKlasy obiekt = new NazwaKlasy(); // Domyślny konstruktor
7         obiekt.metoda_1();
8     }
9 }
```

## Przykład tworzenia klas: klasa Pies

```
1 package psy; // nazwa pakietu (i folderu w którym znajduje się plik)
2
3 // Klasa Pies zawarta w pliku Pies.java
4 public class Pies {
5     /**
6      * Klasa Pies
7      */
8     private int wiek;
9     private String rasa;
10    private String imie;
11    public Pies () {
12        /**
13         * Domyślny konstruktor
14         */
15        wiek = 1;
16        rasa = "kundel";
17        imie = "Azor";
18    }
19    public Pies (int init_wiek, String init_rasa, String init_imie) {
20        /**
21         * Konstruktor pobierający trzy parametry
22         */
23        wiek = init_wiek;
24        rasa = init_rasa;
25        imie = init_imie;
26    }
27    public void setImie(String nowe_imie) {
28        /**
29         * Zmień imię psa
30         * @param nowe_imie imię psa
31         */
32        imie = nowe_imie;
33    }
34    public void setRasa(String nowa_rasa) {
35        /**
36         * Zmień rasę psa
37         * @param nowa_rasa rasa psa
38         */
39        rasa = nowa_rasa;
40    }
41    public void setWiek(int nowy_wiek) {
42        /**
43         * Zmień wiek pieska
44         * @param nowy_wiek wiek w latach
45         * @throws IllegalArgumentException jeśli wiek <= 0
46         */
47        try {
48            if (nowy_wiek <= 0) {
49                throw new IllegalArgumentException("Niepoprawny wiek!");
50            }
51        }
```

```

51         else {
52             wiek = nowy_wiek;
53         }
54     } catch (IllegalArgumentException e) {
55         System.out.println(e);
56     }
57 }
58 public void getInfo() {
59     /**
60      * Informacje o piesku
61      */
62     System.out.println("Hau hau, mam na imię " + imie + " jestem psem
rasy " + rasa + " i mam " + wiek + " lat, szczek szczek"); // (5)
63 }
64 }

1 package psy; // nazwa pakietu (i folderu w którym znajduje się plik)
2
3 // Klasa PiesDemo zawarta w pliku PiesDemo.java
4 public class PiesDemo {
5     public static void main(String[] args) {
6         Pies pies_1 = new Pies(5, "welsh corgi", "Kokos");
7         Pies pies_2 = new Pies(2, "samojed", "Okami");
8         pies_1.setWiek(6);
9         pies_2.setWiek(-1);
10        pies_1.getInfo();
11        pies_2.getInfo();
12    }
13 }

```