

Klasa i obiekt

Klasa i obiekt to podstawowe terminy związane z programowaniem obiektowym.

Klasa jest szablonem lub prototypem, na podstawie którego tworzone są obiekty, nazywane też instancjami danej klasy. Przy pomocy zmiennych i metod definiuje ona własności oraz możliwe do wykonania na każdej instancji klasy operacje.

Składowe klasy w języku Java

- Opcjonalny modyfikator dostępu klasy
- Nazwa klasy
- Opcjonalnie, nazwa klasy bazowej i implementowanych interfejsów wymienione po słowie kluczowym *extends*
- Ciało klasy – pola, konstruktory, metody, klasy zagnieżdżone

Modyfikatory dostępu

W języku Java występują 4 modyfikatory dostępu:

- *private* – dostęp tylko z wnętrza klasy
- domyślny (brak słowa kluczowego) – dostęp z wnętrza danej klasy i klas znajdujących się w tym samym pakiecie,
- *protected* – dostęp z wnętrza danej klasy, klasy znajdujących się w tym samym pakiecie i klas pochodnych
- *public* – dostęp z każdego miejsca.

Pola i metody klasy mogą wykorzystywać wszystkie z nich. Same klasy natomiast mogą posiadać modyfikator domyślny lub *public*.

Pola klasy

Pola klasy to zmienne zadeklarowane w jej ciele. Mogą być skojarzone z samą klasą (pola statyczne, poprzedzone słowem kluczowym *static*) lub z konkretną instancją (pola instancji). Dostęp do pól statycznych nie wymaga utworzenia instancji klasy. Każda instancja klasy posiada własne pola niestacyjne. Dostęp do pól klasy określany jest przy pomocy wymienionych wcześniej modyfikatorów dostępu.

Metody klasy

Metody klasy to funkcje zadeklarowane w jej ciele. Mogą przyjmować parametry i zwracać wartość. Podobnie jak pola, mogą być statyczne lub niestacyjne. Metody statyczne mogą być wywoływane bez uprzedniego utworzenia obiektu danej klasy. Nie mają jednak dostępu do jej niestacyjnych pól. Metody niestacyjne wywoływane są na obiektach i mają dostęp do pól instancji. Dostęp do metod określany jest w ten sam sposób co do pól.

Konstruktory

Konstruktory to specjalne metody wykorzystywane do inicjalizacji obiektów tworzonych przy użyciu słowa kluczowego *new*. Jeżeli w klasie nie zdefiniowano żadnego konstruktora, posiada ona domyślny, niejawny konstruktor, nieprzyjmujący żadnych parametrów. Konstruktory, podobnie jak inne metody, mogą posiadać modyfikator dostępu. W przeciwieństwie do nich jednak nie mogą one zwracać żadnej wartości.

Przykład klasy

```
public class Rower {  
  
    private final static int LICZBA_KOL = 2; // pole statyczne  
    private String nazwa; // pole instancji  
  
    // konstruktor bezargumentowy  
    public Rower() {  
        this.nazwa = "Jubilat"; // inicjalizacja pola instancji nazwa  
    }  
  
    // konstruktor z parametrem  
    public Rower(String nazwa) {  
        this.nazwa = nazwa; // inicjalizacja pola instancji nazwa  
    }  
  
    // metoda niestaticzna  
    public String zwrocNazwe() {  
        return nazwa;  
    }  
  
    // metoda niestaticzna  
    public void ustawNazwe(String nazwa) {  
        this.nazwa = nazwa;  
    }  
  
    // metoda statyczna  
    public static int zwrocLiczbeKol() {  
        return LICZBA_KOL;  
    }  
  
}
```

Przykładowe obiekty

```
Rower romet = new Rower( nazwa: "Romet"); // tworzenie nowego obiektu  
Rower jubilat = new Rower(); // tworzenie nowego obiektu  
romet.zwrocNazwe(); // wywołanie metody instancji  
Rower.zwrocLiczbeKol(); // wywołanie metody statycznej
```