

Objektovo orientované programovanie

Učiteľ:
Ing. Jozef Wagner PhD.

Učebnica:
<https://oop.wagjo.com/>

OPG

Teória 9

1. Viditeľnosť premenných
2. Zatienenie atribútov
3. Inferencia typov
4. Názvy premenných

Viditeľnosť premenných

Premenná má svoju viditeľnosť (visibility) a rozsah (scope)

Viditeľnosť atribútov je daná modifikátorom prístupu, napr. **public**, **private**.

V Jave rozlišujeme nasledovné rozsahy, v ktorých existujú premenné a atribúty:

- Lokálny rozsah (lokálne premenné)
- Inštančný rozsah
- Triedny rozsah (Statické atribúty)

Viditeľnosť statických atribútov

Dostupné počas celého programu

K statickým atribútom pristupujem cez názov triedy

```
class Example {  
    // statický atribút  
    static int staticVar = 20;  
  
    void metoda() {  
        System.out.println(staticVar);  
    }  
}  
  
System.out.println(Example.staticVar);
```

Viditeľnosť inštančných atribútov

Inštančné atribúty sú viditeľné z metód priamo, alebo k nim vieme pristupovať cez premennú **this**.

```
class Example {  
    // inštančný atribút  
    int instanceVar = 10;  
  
    void metoda() {  
        System.out.println(instanceVar);  
        System.out.println(this.instanceVar);  
    }  
}
```

```
Example obj = new Example();  
System.out.println(obj.instanceVar);
```

Lokálny rozsah

Parametre metód a lokálne premenné sú viditeľné iba v rámci metódy resp. bloku, v ktorom boli definované

```
void metoda(int param) {  
    if (true) {  
        int blockVar = 10;  
        System.out.println(blockVar); // OK  
    }  
    // System.out.println(param); // OK  
    // System.out.println(blockVar); // Chyba  
}
```

Lokálny rozsah

```
void metoda(int param) {  
    for (int i = 0; i < 5; i++) {  
        System.out.println(i);  
    }  
    System.out.println(i); // chyba, i neexistuje  
}
```

Inicializácia

- Atribúty sa inicializujú automaticky, ak sa to neurobí manuálne
- Lokálne premenné musíme inicializovať sami

Názvy premenných a atribútov

- Názov musí začínať písmenom, podčiarkníkom `_` alebo dolárom `$`
- Názov nesmie byť rezervované slovo (`class`, `if`, `static`...)

Zatienenie premenných - shadowing

Ak má parameter alebo lokálna premenná rovnaký názov ako atribút
Na prístup k zatienenému atribútu použijeme **this** alebo názov triedy
(pri statických atribútoch)

```
class Example {  
    int x = 10;
```

```
    void metoda(int x) {  
        System.out.println(x); // Vypíše lokálnu premennú x  
        System.out.println(this.x); // Vypíše atribút x  
    }  
}
```

Lokálna inferencia typov

Na automatické pridelenie typu premennej sa používa kľúčové slovo **var**.

```
void example() {  
    var counter = 10; // int  
    System.out.println(counter);  
}
```

```
for (var i = 0; i < 5; i++) {  
    System.out.println(i); // i je typu int  
}
```

```
var names = List.of("Alice", "Bob");  
for (var name : names) {  
    System.out.println(name); // name je typu String  
}
```