

07. Parametry metod (předávané hodnotou vs. odkazem, jak je to v jazyce JAVA)

Volání metody

Existují sice i metody, které proběhnou pokaždé stejným způsobem, a tudíž jí při zavolání žádná data nepotřebují (bez parametrické). Nicméně častěji se vyskytuje, že metoda potřebuje pro svůj běh nějaká zdrojová data, podle kterých se bude při svém průběhu řídit. Tato data se jí předají právě prostřednictvím parametru při jejím zavolání.

Příkladem metody bez parametrů je například typický getter- závorky pro parametry jsou prázdné.

```
public class cosi {  
    private int test;  
  
    public cosi(int test) {  
        this.test = test;  
    }  
    public int getTest() {  
        return test;  
    }  
}
```

Parametry

Formální parametry

Parametry, které jsou deklarovány v metodě. Tyto parametry zastupují skutečné hodnoty, které se převezmou při volání metody.

Skutečné parametry

Hodnoty, které jsou předány metodě při volání. Skutečné protože při volání je jasná jejich hodnota.

Parametry předávané hodnotou

```
public static int sectiCisla(int a, int b){  
    a = 2 * a;  
    int soucet = a + b;  
    return soucet;  
}
```

Volání předchozí metody

```
int paraA = 12;  
int paraB = 13;  
int vysledek = sectiCisla(paraA, paraB);
```

Při volání metody **sectiCisla** se v paměti alokuje místo pro tuto metodu a při tom současně místo pro formální parametry (int a, int b). Následně se do formálních parametrů zkopíruje obsah skutečných parametrů paraA, paraB.

I když se v metodě měnila hodnota parametru a, v proměnné paraA se nic nezmění.

Parametry předávané referencí

Parametrem ovšem nemusí být jen proměnná primitivního datového typu, ale také pole, nebo přímo objekt. V takovém případě už nedochází ke kopírování skutečných parametrů do formálních, ale je zde využívána jen reference na konkrétní pole nebo objekt.

Při předávání parametrů odkazem (referencí) se předává adresa místa skutečného parametru. To znamená, že hodnota skutečného parametru se při volání metody může změnit.

Odkazem se předávají parametry objektového typu. Výhodou je vyšší rychlost práce s parametrem, protože se nepořizuje kopie skutečných parametrů, ale pouze jejich adresa.