

# JAVA - Proměnné, datové typy a operátory

## Proměnné:

zastupuje libovolný myslitelný objekt z dané třídy.

## Primitivní datové typy Javy:

Java je silně typově orientovaný jazyk, to znamená, že u každé proměnné musí být její typ deklarován. Typ definuje soubor hodnot, jež mohou být v proměnné uloženy, včetně souboru operací, které aplikace (applet) s daty může provádět. Java podporuje následujících osm primitivních datových typů.

### Celočíselné datové typy:

Datový typ	Velikost paměti	Rozsah
<i>long</i>	8 bajtů	-9223372036854775808 až 9223372036854775808
<i>int</i>	4 bajty	-2 147 483 648 až 2 147 483 647
<i>short</i>	2 bajty	-32 768 až 32 767
<i>byte</i>	1 bajt	-128 až 127

### Datové typy s desetinnou částí:

Datový typ	Velikost paměti	Rozsah
<i>double</i>	8 bajtů	+/- 1.79769313486231570 E+308 (15 platných číslic)
<i>float</i>	4 bajty	+/- 3.40282347 E+38 (7 platných číslic)

## Datový typ char pro uložení alfanumerických znaků a znaků Unicode:

Datový typ	Velikost paměti	Rozsah
<i>char</i>	2 bajty	65 536 možných různých znaků

## Datový typ boolean pro logické testování:

Datový typ	Velikost paměti	Rozsah
<i>boolean</i>	1 bit	true nebo false

## Používání operátorů v Javě:

Java obsahuje několik typů operátorů. V následujícím výkladu uvedu pouze jejich přehled a příklad použití, pro podrobnější informace se obraťte na nápovědu.

### Aritmetické operátory

Operátor +	- Sčítání	$výsledek = výraz1 + výraz2$
Operátor -	- Odečítání	$výsledek = číslo1 - číslo2$
Operátor *	- násobení	$výsledek = číslo1 * číslo2$
Operátor /	- dělení	$výsledek = číslo1 / číslo2$
Operátor %	- zbytek po celočíselném dělení	$výsledek = číslo1 \% číslo2$

Při použití aritmetických operátorů Java obecně vrací výsledek v typu, který je dostatečně velký, aby hodnotu výsledku správně vyjádřila. Příklad pokud vynásobíme číslo typu *float* číslem typu *int* pak výsledek je typu *float*.

Pokud dělíme mezi sebou dvě čísla typu *int*, pak Java vrací standartně výsledek typu *int*. To znamená, že se chová jako při celočíselném dělení. Příklad  $7 / 2 = 3$ .

Pokud potřebujete dostat výsledek v pohyblivé řádové čárce musíme to kompilátoru sdělit (přidáme za operand *f*- značí *float* nebo *d* – značí *double*). Příklad  $10f / 4f = 2.5$ .

## Operátory přiřazení

Operátor =	- prosté přiřazení $x = x + y$
Operátor +=	- součet $x = x + y$ $x += y$
Operátor -=	- rozdíl $x = x - y$ $x -= y$
Operátor /=	- dělení $x = x / y$ $x /= y$
Operátor *=	- násobení $x = x * y$ $x *= y$
Operátor % =	- zbytek po celočíselném dělení $x = x \% y$ $x \% = y$
Operátor ^=	- bitově XOR $x = x \wedge y$ $x \wedge = y$
Operátor &=	- bitově AND $x = x \& y$ $x \& = y$
Operátor  =	- bitově OR $x = x   y$ $x  = y$
Operátor <<=	- posun doleva $x = x << y$ $x << = y$
Operátor >>=	- posun doprava $x = x >> y$ $x >> = y$
Operátor >>>=	- posun doprava se zaplněním nulami $x = x >>> y$ $x >>> = y$

## Relační operátory

Operátor <	- menší než
Operátor <=	- menší nebo rovno
Operátor >	- větší než
Operátor >=	- větší nebo rovno
Operátor ==	- rovná se
Operátor !=	- nerovná se

## Logické operátory

Operátor&&	- logický součin
Operátor	- logický součet
Operátor !	- negace výrazu

Pozor Java vyhodnocuje pouze tolik operandů, kolik je nezbytně nutné pro vyhodnocení výrazu!

## Deklarace proměnné v Javě

Proměnná je dočasný prostor pro uložení dat v programu. Proměnné je možné deklarovat v Javě na začátku každého programového bloku. Obvykle se proměnné deklarují na začátku třídy nebo metody. Usnadňuje to pozdější orientaci v kódu.

### Příklad explicitní deklarace proměnné *Prijmeni* datového typu String v metodě `init()`

```
public void init()
{
    string Prijmeni;    // proměnné Prijmeni
}
```

Každá proměnná musí mít jednoznačné jméno, které se neshoduje s žádným příkazem Javy. Jméno proměnné:

- musí začínat písmenem (rozumí se znaky a .. Z, \_, \$),
- mimo prvního znaku může obsahovat jméno libovolné písmena i čísla,
- ve jménu proměnné se nesmí vyskytovat znaky +, @ apod.

Přiřazení hodnoty do proměnné se uskuteční pomocí znaku přiřazení =.

### Příklad přiřazení hodnoty do proměnné *Prijmeni*

```
Prijmeni = "Novák";
```

Přiřazení proměnné je možné i při deklaraci proměnné (obdobně jako v C/C++).

### Příklad přiřazení hodnoty do proměnné *Prijmeni* při deklaraci

```
string Prijmeni = "Novák";
```