

Dátové typy

<code>int cislo = 5;</code>	Vytvorí premennú <code>cislo</code> typu celé číslo a nastaví jej hodnotu 5. Alternatívy: <code>long</code> , <code>short</code> , <code>byte</code>
<code>double cislo = 2.8;</code>	Vytvorí premennú <code>cislo</code> typu reálne číslo a nastaví jej hodnotu 2.8. Alternatíva: <code>float</code>
<code>boolean suhlas = true;</code>	Vytvorí premennú <code>suhlas</code> typu <code>boolean</code> a nastaví jej hodnotu <code>true</code>
<code>char pismeno = 'a';</code>	Vytvorí premennú <code>pismeno</code> typu znak a nastaví jej hodnotu <code>a</code> .
<code>String meno = "Jozef";</code>	Vytvorí premennú <code>meno</code> typu reťazec znakov a nastaví jej hodnotu „Jozef“ (bez úvodzoviek).
<code>int[] pole = new int[] {}</code>	Vytvorí premennú <code>pole</code> typu pole celých čísel.

Aritmetické operácie

<code>+</code> <code>-</code> <code>/</code> <code>*</code>	Klasické matematické operácie pre typy <code>int</code> , <code>long</code> , <code>short</code> , <code>byte</code> , <code>double</code> a <code>float</code> .
<code>+</code>	Funguje aj na <code>String</code> , „A“ + „B“ = „AB“
<code>%</code>	Modulo – zvyšok po delení: <code>7 % 5 = 2</code> .

Logické operácie a porovnávania

<code>!</code>	Negácia	<code>! (5 == 4)</code>
<code>&&</code>	A zároveň	<code>(vek > 0 && vek < 100)</code> // vek je medzi 0 a 100
<code> </code>	Alebo	<code>(vek < 0 vek > 100)</code> // vek je buď menší ako 0 alebo väčší ako 100
<code><=</code> <code><</code>	Menšie alebo rovné Menšie	<code>4 <= 5</code> <code>4 < 5</code>
<code>>=</code> <code>></code>	Väčšie alebo rovné iba ostro väčšie	<code>5 >= 5</code> <code>5 > 4</code>
<code>==</code> <code>!=</code>	Rovné nerovné	<code>5 == 5</code> <code>4 != 5</code>

String

<code>indexOf(znak)</code>	Vráti pozíciu znaku v texte. Počíta sa od nuly. Ak nenájde, vráti -1	<code>"Text".indexOf('e') = 1</code>
<code>length()</code>	Vráti počet znakov v texte.	<code>"Toto".length() = 4</code>
<code>charAt(pozícia)</code>	Vráti znak z textu na danej pozícii.	<code>"Java".charAt(1) = 'a'</code>
<code>substring(beginIndex, endIndex)</code>	Vráti časť reťazca.	<code>"Java".substring(1,3) = "av"</code>

Príkazy

<code>for (začiatok; podmienka; inkrement) { príkazy; }</code>	Cyklus, ktorý je definovaný začiatkom a beží dovtedy, kým je splnená podmienka. V cykle sa vykonávajú príkazy. Po každom behu sa vykoná inkrement.
<code>if (podmienka) { príkazy; } else { príkazy2; }</code>	Príkazy sa vykonávajú iba vtedy, ak je splnená podmienka. Ak je uvedené aj slovíčko „else“, znamená to, že „príkazy2“ sa vykonávajú vtedy, ak podmienka splnená nie je.
<code>while (podmienka) { príkazy; } }</code>	Cyklus, ktorý beží, kým je podmienka splnená
<code>int metoda(int a) { príkazy; return 0; }</code>	Zadeklaruje metódu, ktorá na vstupe dostane celé číslo, vykoná „príkazy“ a na výstupe vráti celé číslo.

Iné

<code>System.out.println(vyraz);</code>	Vypíše výraz na konzolu.
<code>Math.sqrt(vyraz);</code>	Vypočíta druhú odmocninu z daného výrazu.
<code>Math.random();</code>	Vráti náhodné číslo z intervalu 0 až 1.