

Prosti tipovi podataka

Značaj tipova podataka

- **Veća preglednost i jasnoća programa** - programer radi sa apstraktnim pojmovima brojeva, znakova, i sl., ne opterećujući se načinom njihovog unutrašnjeg predstavljanja
- **Provera usklađenosti tipova** - prevodilac proverava kako se koriste konstante i promenljive u izrazima, u skladu sa operatorima koji su dozvoljeni za pojedine tipove podataka
- **Veća efikasnost prevedenog koda** - pošto je usklađenost tipova već proverena, Java prevodilac može generisati efikasniji kod koji ne vrši nepotrebne provere u toku izvršavanja programa

Prosti tipovi podataka u Javi

- U Javi postoji osam prostih (primitivnih) tipova podataka:
 - `boolean` (logičke vrednosti)
 - `char` (znakovi)
 - `byte` (celi brojevi)
 - `short` (celi brojevi)
 - `int` (celi brojevi)
 - `long` (celi brojevi)
 - `double` (realni brojevi)
 - `float` (realni brojevi)

Tip boolean

- Tip `boolean` predstavlja dve moguće logičke vrednosti: `true` (tačno) i `false` (netačno)
- Primer:

```
boolean nasao = true;
```
- U gornjem primeru se deklariše promenljiva sa imenom `nasao`, tipa `boolean`, i dodeljuje joj se vrednost `true`
- Ovo je tip podataka sa najmanje mogućih vrednosti, što mu ne umanjuje važnost

Tip boolean

Operatori (nad vredostima logičkog tipa)	Relacioni operatori == , !=
	Logički operatori ! (NOT), & (AND), (OR), ^ (XOR) & & (AND) , (OR)
	? : (uslovni operator) + (konkatenacija stringova)

- Logičke vrednosti se dobijaju i kao rezultat primene relacionih operatora na vrednosti celobrojnih ili realnih tipova podataka

Tip boolean

Ispravni logički izrazi

<code>tekucaDuzina <= maxDuzina</code>	(promenljive tipa <code>int</code>)
<code>a && b</code>	(<code>a</code> i <code>b</code> su tipa <code>boolean</code>)
<code>a b == (x <= y)</code>	(<code>x</code> i <code>y</code> su tipa <code>double</code>)

Neispravni logički izrazi

<code>broj == true</code>	(<code>broj</code> je tipa <code>int</code>)
<code>broj + 1 == !(1 < 3)</code>	

Tip char

- Tip `char` (skraćeno od engl. *character*) predstavlja jedan znak iz Unicode rasporeda
- Literal ovog tipa navode se korišćenjem jednostrukih navodnika `' '`
- Primeri:
 - `'a'` // Slovo a
 - `' '` // Razmak
 - `'\t'` // Znak za tabulator
 - `'\''` // jednostruki navodnik
 - `'\"'` // dvostruki navodnik
- Znakovni literal obrađeni su na prethodnom predavanju
- Tip `String`, koji služi za predstavljanje sekvenci znakova, nije prost tip (u pitanju je klasa)
 - Kod njegovih literala koriste se dvostruki navodnici `" "`
- Operatori nad tipom `char` su isti kao za celobrojne tipove, štaviše `char` se pri primeni operatora i tretira kao celobrojni tip (sa vrednostima koje predstavljaju redne brojeve znakova u Unicode rasporedu)

Celobrojni tipovi podataka

- **byte** – 1 bajt (8 bita), celi brojevi iz intervala
 $[-2^7, 2^7-1] = [-128, 127]$
- **short** – 2 bajta (16 bita)
 $[-2^{15}, 2^{15}-1] = [-32768, 32767]$
- **int** – 4 bajta (32 bita)
 $[-2^{31}, 2^{31}-1] = [-2147483648, 2147483647]$
- **long** – 8 bajtova (64 bita)
 $[-2^{63}, 2^{63}-1] = [-9223372036854775808, 9223372036854775807]$
- (Ako je n broj bitova kojim se predstavlja ceo broj, tada su moguće vrednosti celobrojnog tipa $[-2^{n-1}, 2^{n-1}-1]$)

Celobrojni literali

- Celi brojevi se mogu navoditi u nekoliko različitih brojnih sistema, sa osnovama 10, 16, 8 i 2
- Decimalni zapis (sa osnovom 10) – standardan zapis
Primer: 42
- Heksadecimalni zapis (osnova 16) – počinje znakovima 0x
Primer: 0x2A
- Oktalni zapis (osnova 8) – počinje sa 0
Primer: 052
- Binarni zapis (osnova 2) – počinje sa 0b
Primer: 0b101010
- Ako želimo da naglasimo da je literal tipa `long`, to radimo dodavanjem `L` ili `l` na kraj literala (pri čemu se ne preporučuje `l` zbog sličnosti sa cifrom 1)
Primeri: 42L 0x2AL 052L 0b101010L
- Inače se podrazumeva da je literal tipa `int` (mada su usklađeni i sa drugim brojevnim tipovima)

Operatori nad celobrojnimi tipovima

- **Relacioni operatori:** `<` , `<=` , `>` , `>=` , `==` , `!=`
- **Aritmetički operatori:** `+` , `-` , `*` , `/` , `%` , `++` , `--`
- **Operatori nad bitovima:** `<<` , `>>` , `>>>` , `~` , `&` , `|` , `^`
- **Operatori dodele:** `=` , `+=` , `-=` , `*=` , `/=` , `%=` , `<<=` , `>>=`
`>>>=` , `&=` , `|=` , `^=`
- **Uslovni operator** `?` `:`
- **“Cast” operatori za konverziju tipova:** `(imeTipa)`
- **Operator konkatencije stringova** `+`
- **Klase za rad sa tipovima:** `Character` , `Byte` , `Short` , `Integer` , `Long` , klasa `Math` – matematičke funkcije

Celobrojni tipovi podataka: primeri

Ispravni celobrojni izrazi	Neispravni celobrojni izrazi
<code>19 >> 3</code>	<code>(int) true</code>
<code>0x33 + 033 + 33</code>	<code>099 + 1 // los oktalni zapis</code>
<code>12L + 45 / 2 % 3</code>	
<code>'a' + 1</code>	

Primena klasa `Byte`, `Integer` i `Math`

```
System.out.println(Byte.MAX_VALUE);  
  
int i = Integer("123").intValue();  
  
System.out.println(Math.sin( Math.PI / 2 ));
```