

JAVA



ПЛАТФОРМЕННО-НЕЗАВИСИМИ ПРОГРАМНИ ЕЗИЦИ

Лекция 2. Изрази и операции

доц. д-р инж. Румен П. Миронов



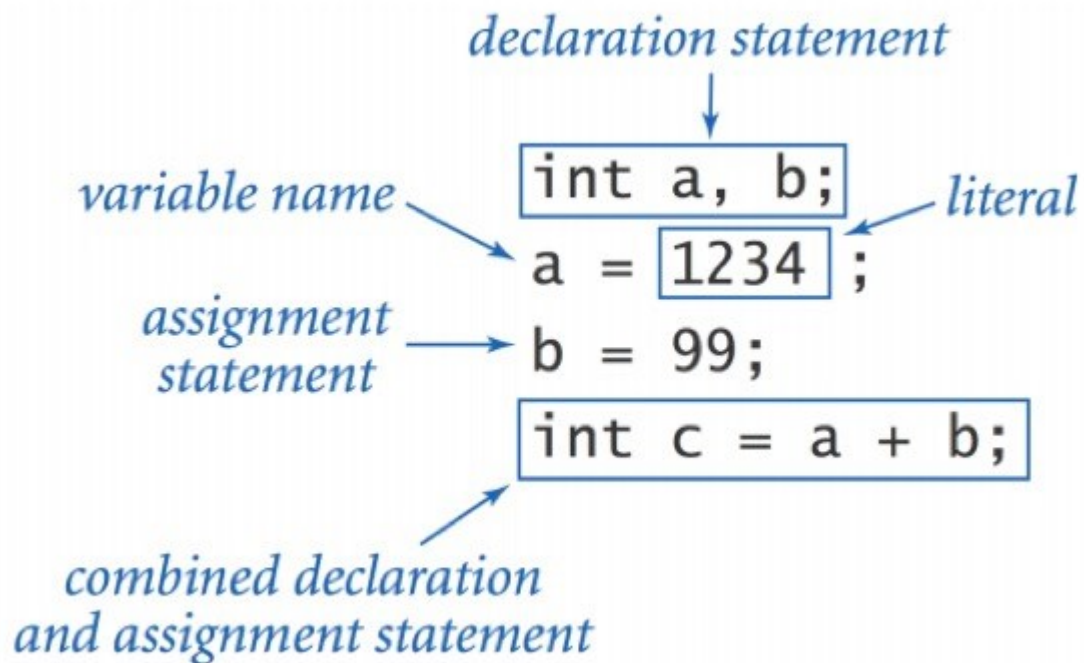
1. Типове данни и променливи



Резервирани ключови думи в Java

<u>abstract</u>	<u>do</u>	<u>implements</u>	public	transient
assert	<u>double</u>	<u>import</u>	requires	<u>try</u>
<u>boolean</u>	<u>else</u>	<u>instanceof</u>	return	var
<u>break</u>	<u>enum</u>	<u>int</u>	short	<u>void</u>
<u>byte</u>	exports	<u>interface</u>	static	volatile
<u>case</u>	<u>extends</u>	<u>long</u>	strictfp	<u>while</u>
<u>catch</u>	<u>final</u>	module	super	
<u>char</u>	<u>finally</u>	native	switch	
<u>class</u>	<u>float</u>	<u>new</u>	synchronized	
<u>continue</u>	<u>for</u>	<u>package</u>	this	
const	goto	<u>private</u>	throw	
<u>default</u>	<u>if</u>	<u>protected</u>	throws	

1. Типове данни и променливи



1.2 Литерали



- логически – **true** или **false**;
- символни – Unicode символ заграден в кавички;

Пример за логически и символни литерали:

<code>boolean istrue = true;</code>	<code>char symbol = 'a';</code>	// символ a
<code>boolean isfalse = false;</code>	<code>char number = '4';</code>	// символ 4
	<code>char newline = '\n';</code>	
	<code>char tabulator = '\t';</code>	

1.2. Литерали



➤ Целочислени

- десетичен - по подразбиране (пример: 1024)
- осмичен - префикс 0 (пример: 0777)
- шестнадесетичен - префикс 0x (пример: 0x1F5A)

ИЗПОЛЗВАНИ ТИПОВЕ

int - по подразбиране

long – суфикс L (пример: 523987234L)

```
int i1 = 0x2f; // Hexadecimal (lowercase)
int i2 = 0X2F; // Hexadecimal (uppercase)
int i3 = 0177; // Octal (leading zero)
int i4 = 1024;
```

```
byte b = 0x7f; // max byte hex value
short s = 0x7fff; // max short hex value
```

```
long n1 = 200L; // long suffix
long n2 = 200l; // long suffix (can be confusing)
long n3 = 200;
```

1.2. Литерали



- Дробни – с плаваща запетая
 - нормален (например: 0.5)
 - експоненциален – запис: мантика, суфикс E, експонента (2.34E+15)
 - шестнадесетичен - префикс 0x (0x1F5A)

ИЗПОЛЗВАНИ ТИПОВЕ

float - суфикс F или f (например: 1.025f)

double – суфикс d или D

```
float f1 = 1;  
float f2 = 1F; // float suffix  
float f3 = 1f; // float suffix
```

```
// Uppercase and lowercase 'e' are the same:  
float f4 = 1e-43f;           // 10 to the power  
float expFloat = 1.39e-43f;
```

```
double d1 = 1d; // double suffix  
double d2 = 1D; // double suffix
```

```
double expdouble = 4.76E100d;  
double expdouble = 4.76E-100d;  
double expDouble = 47e47d; // 'd' is optional  
double expDouble2 = 47e47; // Automatically double
```

1.2. Литерали



➤ Низови литерали

- последователност от Unicode символи, заградена в кавички (примери: “абв”, “abc”)

```
String s = "abcsdf";  
String s = „абвгд“;
```

Типове данни

- Примитивни типове
 - Булев (boolean) тип
 - Числени (numeric) типове
 - Целочислени (integer) типове
 - Типове за числа с плаваща запетая (floating-point)
- Референтни типове (reference types)
- Обвиващи типове (wrapper types)
 - Представяват референтни аналози на примитивните типове
 - Използват се:
 - където синтаксисът на езика изисква обект, а не примитивен тип;
 - когато ни трябват константи или помощни функции, които са имплементирани в съответния wrapper клас;
 - Имплицитно се конвертират към съответния си примитивен тип, и обратно

1.3 Основни типове данни



Primitive type	Size	Minimum	Maximum	Wrapper type
boolean	—	—	—	Boolean
char	16 bits	Unicode 0	Unicode $2^{16}-1$	Character
byte	8 bits	-128	+127	Byte
short	16 bits	-2^{15}	$+2^{15}-1$	Short
int	32 bits	-2^{31}	$+2^{31}-1$	Integer
long	64 bits	-2^{63}	$+2^{63}-1$	Long
float	32 bits	IEEE754	IEEE754	Float
double	64 bits	IEEE754	IEEE754	Double
void	—	—	—	Void

1.3 Основни типове данни



Default values for primitive members

Primitive type	Default
boolean	false
char	'\u0000' (null)
byte	(byte)0
short	(short)0
int	0
long	0L
float	0.0f
double	0.0d

1.3 Основни типове данни



Защо ни трябват типове?

За да ни помага компилаторът в откриването на грешки



През 1996, ракетата Ариана 5 експлодира след излитане поради софтуерна грешка в конвертирането на типове (опит да „набута“ 64-битово число в 16 бита).

1.3 Основни типове данни



Конвертиране на типовете

- Имплицитно - без загуба на точност
- Експлицитно - чрез cast

Израз	Тип на израза	Стойност на израза
"1234" + 99	String	"123499"
(int) 2.71828	int	2
11 * 0.3	double	3.3
(int) 11 * 0.3	double	3.3
11 * (int) 0.3	int	0
(int) (11 * 0.3)	int	3

4. Оператори на езика JAVA



□ Унарни – един операнд

- унарни плюс и минус: +операнд, -операнд

$x = +4, y = +0.7, x = -3, y = -0.5, z = -(y-5);$

- инкрементиране и декрементиране: ++, -- (целочислени типове)

- постфиксна нотация (postfix notation) : операнд++, операнд --

- префиксна нотация (prefix notation): ++операнд, --операнд

$x++, y--, ++x, --y$

- побитово инвертиране: ~операнд (целочислени типове)

$x = \sim y, (y \rightarrow 01001000, y \rightarrow 10110111)$

- логическо отрицание: !операнд (boolean тип)

$a = !b, (b \rightarrow \text{true}, a \rightarrow \text{false})$

- преобразуване на типове, типизирне, cast: () операнд

примитивни типове: `int x = (int) 0.25; long y = (long) (Math.PI*123.4);`

референции към обекти: `Vector list = new Vector();`

`list.add("Символен низ");`

`String str = (String) list.get(0);`

4. Оператори на езика JAVA



□ Аритметични – два операнда (инфиксна нотация (infix notation))

- умножение: операнд1 * операнд2 $x = y * 5;$
- деление: операнд1 / операнд2 $y = z / x;$
- деление по модул: операнд1 % операнд2 $x = 10 \% 3; (x = 1)$
- събиране: операнд1 + операнд2 $x = y + z;$
- изваждане: операнд1 – операнд2 $x = y - z;$

□ Оператори за преместване – два операнда

- на ляво със знак (signed): операнд1 << операнд2
 - новите битове се заместват с 0
- на дясно със знак: операнд1 >> операнд2
 - новите битове копират най-старшия бит
- на дясно без знак (unsigned): операнд1 >>> операнд2
 - новите битове се заместват с 0

4. Оператори на езика JAVA



□ Оператори за сравнение – връщат резултат от тип `boolean` – два операнда

- равенство: операнд1 `==` операнд2 `x == 5; myObject1 == myObject2 ;`
- не-равенство: операнд1 `!=` операнд2 `x != y;`
- по-малко `<`
- по-голямо `>`
- по-малко или равно `<=`
- по-голямо или равно `>=`
- принадлежност към тип операнд1 `instanceof` операнд2
 операнд1 – референция към обект, операнд2 – име на клас
 `myObject instanceof String`

4. Оператори на езика JAVA



❑ Логически оператори (boolean тип)

- логическо И: &&
- логическо ИЛИ: ||

❑ Побитови оператори

- логическо И: &
- логическо ИЛИ: |
- изключващо ИЛИ: ^

❑ Тернарен оператор

- [логически израз] ? оператор1: оператор2;

a = x ? b : c;

```
if (x) {  
    a = b;  
} else {  
    a = c;  
}
```


4. Оператори на езика JAVA



□ Оператори за присвояване

■ `=`, `+=`, `-=`, `*=`, `/=`, `%=`

`x = 5`, `x += 5` (`x = x + 5`) , `y += 5` (`y = y + 5`), `z *= 5` (`z = z * 5`);

`a /= 5` (`a = a / 5`), `x %= 5` (`x = x % 5`);

4. Оператори на езика JAVA



Оператори	Приоритет
postfix	expr++ expr--
unary	++expr --expr +expr -expr ~ !
multiplicative	* / %
additive	+ -
shift	<< >> >>>
relational	< > <= >= instanceof
equality	== !=
bitwise AND	&
bitwise exclusive OR	^
bitwise inclusive OR	
logical AND	&&
logical OR	
ternary	? :
assignment	= += -= *= /= %= &= ^= = <<= >>= >>>=

Java Online Compiler (Editor)

With our online Java compiler, you can edit Java code, and view the result in your browser.

```
public class Main {  
    public static void main(String[] args) {  
        System.out.println("Hello World!");  
    }  
}
```

Hello World!

Click on the "Try it Yourself" button to see how it works.

Java Tutorial

<https://www.javatpoint.com/history-of-java>

<https://www.w3schools.com/java/>