Backend Java Basics

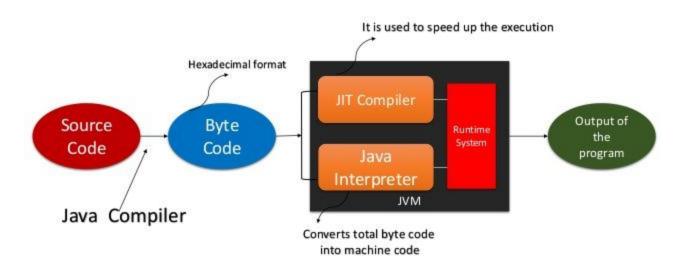
Особенности языка Java

- Простой
- Объектно-ориентированный
- Кроссплатформенный
- Интерпретируемый
- Распределенный
- Надежный
- Безопасный
- Многопоточный
- Высокопроизводительный

Как работает Java

How Java Works?





Simple Program

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

Java Syntax

- Синтаксис унаследован от C++
- Case sensitive
- Каждый statement заканчивается точкой с запятой ;
- Блоки помещаются в фигурные скобки '{' и '}'
- Пробел, табуляция, и перевод строки используются для форматирования кода, удобства чтения и понимания
- Комментарии (// one-line, /* Multiple line */, /** JavaDoc */)

Идентификаторы

- Это имена, которые даются различным элементам языка для упрощения доступа к ним.
- Имена имеют пакеты, классы, интерфейсы, поля, методы, аргументы и локальные переменные.
- ▶ Правило: любая комбинация латинских букв (uppercase, lowercase), чисел (но не начинается с числа) и _, \$.
- Character, c, D, x1, x2, Math, sqrt, x,
- PI, condition, getWidth, getHeight,
- lang, stack, Stack, STACK_SIZE, wav2snd, _snd, \$snd

Keywords

Complete List of Java Keywords

abstract	boolean	break	byte	case	catch
char	class	const	continue	default	do
double	else	extends	extends final		float
for	goto	if	f implements		instanceof
int	interface	long	native	new	package
private	protected	public	return	short	static
strictfp	super	switch	synchronized	this	throw
throws	transient	try	void volatile		while
assert					

Примитивные типы данных

Category	Types	Size (bits)	Min Value	Max Value	Example
Integer	byte	8	-128	127	byte b = 65;
	char	16	0	2 ¹⁶ -1	char c = 'A'; char c = 65;
	short	16	-2 ¹⁵	2 ¹⁵ -1	short s = 65;
	int	32	-2 ³¹	2 ³¹ -1	int i = 65;
	long	64	-2 ⁶³	2 ⁶³ -1	long I = 65L;
Floating- point	float	32	3.4 ⁻³⁸	3.4 ³⁸	float f = 65f;
	double	64	1.7 ⁻³⁰⁸	1.7308	double d = 65.55;
Other	boolean	1	false	true	boolean b = true;
	void				_

Переменные

Переменные используются в программе для хранения данных.

Любая переменная имеет три базовые характеристики:

- **RMN** ◀
- ▶ТИП
- ▶ значение

```
int a;
int b = 0, c = 3+2;
int d = b+c;
int e = a = 5;
boolean result = true;
```

Операторы

Category	Operators		
postfix	expr++ expr		
unary	++exprexpr +expr -expr - !		
casting	(type)		
multiplicative	* / %		
additive	+ -		
shift	<< >> >>>		
relational	<> <= >= instanceof		
equality	!		
bitwise AND	&		
bitwise exclusive OR	^		
bitwise inclusive OR	Ti .		
logical AND	&&		
logical OR	H		
ternary	? :		
assignment	= += -= *= /= %= &= ^= = <<= >>		

Примитивные и ссылочные типы данных

Что произойдет со второй переменной?

Условный оператор if

```
if (условие) оператор; // если условие истинно, то выполняется оператор
```

```
if (условие) оператор; // если условие истинно, то выполняется первый оператор else оператор; // если условие ложно, то выполняется оператор после else
```

```
if (условие)
{
    onepatop1;
    onepatop2;
}
else
{
    onepatop1;
    onepatop2;
}
```

```
if(i == 10)
{
    if(j < 20) a =b;
    if(k > 100) c =d;
    else a = c; // else относится κ if(k > 100)
}
else a = d; // else относится κ if(i == 10)
```

```
int lowNum = 9;
int highNum = 27;
int largerNum = lowNum < highNum ? highNum : lowNum;</pre>
```

```
if (condition)
    statement;
else if (condition)
    statement;
else if (condition)
    statement;
.
.
.
else
    statement;
```

Условный оператор if

Описание
Меньше чем
Меньше или равно
Больше чем
Больше или равно
Равно
Не равно

Оператор множественного выбора switch

```
switch(ВыражениеДляСравнения) {
    case Совпадение1:
            команда;
                break;
        case Совпадение2:
            команда;
                break;
        case Совпадение3:
            команда;
                 break;
        default:
            оператор;
            break;
```

```
int month = 3;
String monthString:
switch (month) {
    case 1: monthString = "Январь";
             break:
    case 2: monthString = "Февраль";
             break;
    case 3: monthString = "Mapt";
             break;
    case 4: monthString = "Апрель";
             break;
    case 5: monthString = "Май";
             break:
    case 6: monthString = "Июнь";
             break:
    case 7: monthString = "Июль";
             break;
    case 8: monthString = "ABryct";
             break;
    case 9: monthString = "Сентябрь";
             break;
    case 10: monthString = "Октябрь";
             break:
    case 11: monthString = "Ноябрь";
             break:
    case 12: monthString = "Декабрь";
             break;
    default: monthString = "Не знаем такого";
             break;
```

Циклы

for

```
for (int i = 0; i < 10; ++i){
    System.out.print ("Ku-Ku ");//Обратите внимание на метод print
    //В отличие от println метод print не переводит курсор на новую строку
}
```

while

```
int i = 0;
while (i < 10){
    System.out.print (i++);
}</pre>
```

do-while

```
int i = 0;
do{
   System.out.print (i++);
} while (i < 10);</pre>
```

Циклы

continue

```
for(int i = 0; i < data.length; i++) {
    if (data[i] == -1)
        continue;
    square(data[i]);
 return
 double square (double x) {
     return x * x;
  break
 for(int i = 0; i < data.length; i++) {</pre>
      if(data[i] == target) {
          index = i;
          break;
```

Массивы

В отличие от обычных переменных, которые хранят только одно значение, массивы (arrays) используются для хранения целого набора значений.

```
int[] cats; // мы объявили переменную массива
Возможна и альтернативная запись:
 int cats[]; // другой вариант
 cats = new int[10];
Можно одновременно объявить переменную и определить массив (в основном так и делают):
 int[] cats = new int[10];
int[][] matrixA;
matrixA = new int[2][3];
```

Массивы

```
Object arr[] = new Object[3];
arr[0] = new Object();
arr[1] = null;
arr[2] = arr;
int len = arr.length;
Point p = new Point(1, 3);
Point arr[] = {p, new Point(2,2), null, p};
String strs[] = { "aaa", "bbb", "cde" +
"xyz"};
int pithagor table[][]=new int[5][5];
for (int i = 0; i < 5; i++) {
    for (int j = 0; j < 5; j++) {
       pithagor table [i][j] = i * j;
       System. out.print(pithagor table[i][j] + "\t");
    System.out.println();
```

Домашнее задание

- 1. Создать репозиторий на Bitbucket и подключить к IDE.
- 2. Посчитать среднее арифметическое массива.
- 3. Факториал числа.
- 4. Вывести цену за определенный товар опираясь на его код (предусмотреть возможность введение неверного кода).
- 5. Вывести на экран информацию о том является ли целое число записанное в переменную n, чётным либо нечётным.

Links

- Intellij Idea https://www.jetbrains.com/idea/download
- Java SDK http://www.oracle.com/technetwork/java/javase/downloads/jdk8-down loads-2133151.html
- Java code conventions http://www.oracle.com/technetwork/java/codeconventions-150003.pdf
- Bitbucket https://bitbucket.org/