



Язык программирования Java. Начальный уровень

Мультимедийный курс

автор: Васильев А.Н.

www.vasilev.kiev.ua

Киев 2017



Лекция 4. Массивы



- Одномерные массивы
- Двумерные массивы
- Инициализация массивов
- Копирование и присваивание массивов
- Основные операции с массивами

*Я представитель
цивилизованной планеты, и
требую, чтобы вы проследили
бы за своим лексиконом!
из к/ф "Кин-дза-дза"*

Массивы: основные определения



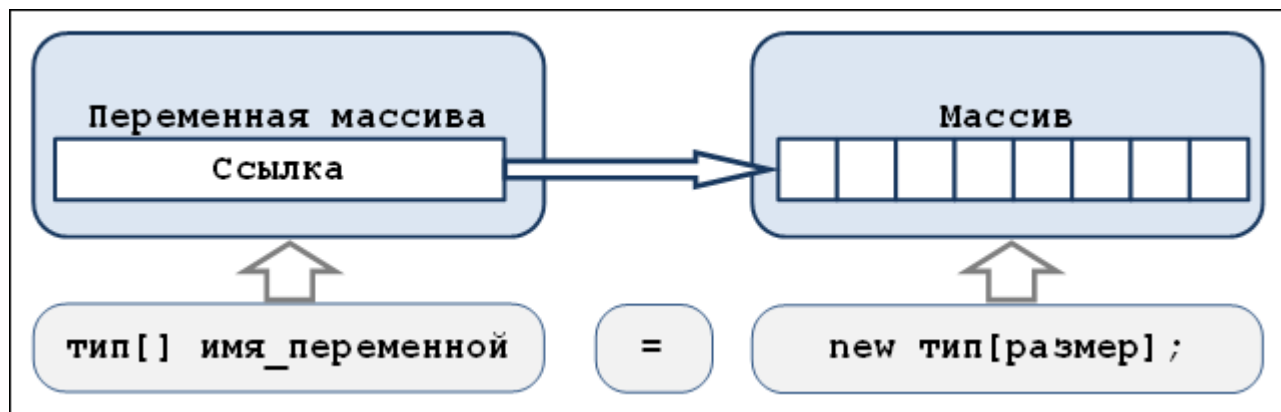
- Массив - набор однотипных элементов, объединенных общим именем, доступ к которым выполняется с помощью целочисленного индекса (или индексов).
- Размер массива - количество элементов в массиве.
- Размерность массива - количество индексов, которые указываются при идентификации элемента массива.
- Одномерный массив - массив, в котором доступ к элементу выполняется с помощью одного индекса (в формате `имя_массива[индекс_элемента]`).



Одномерный массив: создание



Схема реализации одномерного массива:



Переменная массива: указывается тип элементов в массиве, пустые квадратные скобки [] и имя переменной

Массив: создается с помощью инструкции `new`, после которой следует указывается тип элементов и размер массива (в квадратных скобках)

```
int[] nums=new int[12];
```

```
int nums[]=new int[12];
```

```
int[] nums;  
nums=new int[12];
```

Создание массива из 12 целочисленных элементов



Одномерный массив: элементы и размер



Доступ к элементу массива:

- Указывается имя массива и индекс элемента в квадратных скобках.
- Индексация начинается с нуля (индекс первого элемента равен нулю).
- Индекс последнего элемента на единицу меньше размера массива).

Размер массива можно определить с помощью свойства `length`, которое через точку указывается после имени массива.

Например:

```
int[] nums=new int[12];
```

Тогда:

<code>nums[0]</code>	- первый элемент массива
<code>nums.length</code>	- размер массива (значение равно 12)
<code>nums[nums.length-1]</code>	- последний элемент массива

Программа: одномерный массив



```
class ArrayDemo{
    public static void main(String[] args){
        // Создание массива из 12 чисел:
        int[] nums=new int[12];
        // Перебор элементов массива:
        for(int k=0;k<nums.length;k++){
            // Присваивание значения элементу массива:
            nums[k]=2*k+1;
            // Отображение значения элемента массива:
            System.out.print("| "+nums[k]+" ");
        }
        System.out.println("|");
    }
}
```

Результат выполнения программы:

| 1 | 3 | 5 | 7 | 9 | 11 | 13 | 15 | 17 | 19 | 21 | 23 |

Массив случайных чисел

```
import java.util.Random;
class RandArrayDemo{
    public static void main(String[] args){
        // Объект для генерирования случайных чисел:
        Random rnd=new Random();
        // Размер массива:
        int n=10;
        // Создание массива:
        int[] nums=new int[n];
        // Заполнение и отображение массива:
        for(int k=0;k<nums.length;k++){
            // Присваивание значения элементу массива:
            nums[k]=rnd.nextInt(21)+1;
            // Отображение значения элемента массива:
            System.out.print(nums[k]+" ");
        }
        System.out.println("");
        // Отображение массива в обратном порядке:
        for(int k=nums.length-1;k>=0;k--){
            System.out.print(nums[k]+" ");
        }
        System.out.println("");
    }
}
```

Результат программы:

```
1 12 3 18 11 12 13 8 3 3
3 3 8 13 12 11 18 3 12 1
```

© Васильев А.Н.


```
import java.util.Random;
class CharArrayDemo{
    public static void main(String[] args){
        Random rnd=new Random();
        int n=8,k;
        // Создание массива:
        char[] a=new char[n];
        for(k=0;k<a.length;k++){
            // Присваивание значения элементу массива:
            a[k]=(char) ('a'+rnd.nextInt(28));
            // Отображение значения элемента массива:
            System.out.print("| "+a[k]+" ");
        }
        System.out.println("|");
        // Массив случайного размера:
        char[] b=new char[rnd.nextInt(6)+3];
        // Значение первого элемента:
        b[0]='A';
        System.out.print("| "+b[0]+" | ");
        for(k=1;k<b.length;k++){
            // Присваивание значения элементу массива:
            b[k]=(char) (b[k-1]+1);
            // Отображение значения элемента:
            System.out.print(b[k]+" | ");
        }
        System.out.println("");
    }
}
```

Результат программы:

	r		p		f		x		s		b		q		c	
	A		B		C		D		E		F		G		H	

Инициализация массива



```
class InitArrayDemo{
    public static void main(String[] args){
        // Создание массивов:
        int[] nums={1,3,7,2,8,5,9};
        char[] symbs={'A','Z','Q','R'};
        String[] txts={"красный", "желтый", "зеленый"};
        // Отображение содержимого массивов:
        System.out.println("Массив nums:");
        for(int k=0;k<nums.length;k++){
            System.out.print(nums[k]+" ");
        }
        System.out.println("\nМассив symbs:");
        for(int k=0;k<symbs.length;k++){
            System.out.print(symbs[k]+" ");
        }
        System.out.println("\nМассив txts:");
        for(int k=0;k<txts.length;k++){
            System.out.print(txts[k]+" ");
        }
        System.out.println("");
    }
}
```

Результат программы:

Массив nums:

1 3 7 2 8 5 9

Массив symbs:

A Z Q R

Массив txts:

красный желтый зеленый

Изменение порядка элементов в массиве



```
class ReverseArrayDemo{
    public static void main(String[] args){
        // Создание массива:
        int[] nums={1,3,7,2,8,5,9};
        // Отображение содержимого массива:
        System.out.println("До инверсии:");
        for(int k=0;k<nums.length;k++){
            System.out.print(nums[k]+" ");
        }
        System.out.println("\nПосле инверсии:");
        // Меняем порядок элементов в массиве:
        for(int k=0,s;k<nums.length/2;k++){
            s=nums[nums.length-1-k];
            nums[nums.length-1-k]=nums[k];
            nums[k]=s;
        }
        // Отображение содержимого массива:
        for(int k=0;k<nums.length;k++){
            System.out.print(nums[k]+" ");
        }
        System.out.println("");
    }
}
```

Результат программы:

До инверсии:

1 3 7 2 8 5 9

После инверсии:

9 5 8 2 7 3 1

Васильев А.Н.

```
class SortArrayDemo{
    public static void main(String[] args){
        int i,j,k,s;
        // Создание массива:
        int[] nums={1,3,7,2,8,5,9};
        // Отображение содержимого массива:
        System.out.println("До сортировки:");
        for(k=0;k<nums.length;k++){
            System.out.print(nums[k]+" ");
        }
        System.out.println("\nПосле сортировки:");
        // Сортировка элементов в массиве:
        for(i=1;i<nums.length;i++){
            for(j=0;j<nums.length-i;j++){
                if(nums[j]>nums[j+1]){
                    s=nums[j+1];
                    nums[j+1]=nums[j];
                    nums[j]=s;
                }
            }
        }
        // Отображение содержимого массива:
        for(k=0;k<nums.length;k++){
            System.out.print(nums[k]+" ");
        }
        System.out.println("");
    }
}
```

Результат программы:

До сортировки:

1 3 7 2 8 5 9

После сортировки:

1 2 3 5 7 8 9

Поиск максимального значения

```
import java.util.Random;
class FindMaxDemo{
    public static void main(String[] args){
        int k,val;
        // Объект для генерирования случайных чисел:
        Random rnd=new Random();
        // Создание массива:
        int[] nums=new int[12];
        // Заполнение и отображение массива:
        for(k=0;k<nums.length;k++){
            // Присваивание значения элементу массива:
            nums[k]=rnd.nextInt(100)+1;
            // Отображение значения элемента массива:
            System.out.print(nums[k]+" ");
        }
        System.out.print("\nMax value is ");
        // Поиск наибольшего значения:
        val=nums[0];
        for(k=1;k<nums.length;k++){
            if(nums[k]>val) val=nums[k];
        }
        System.out.println(val);
    }
}
```

Результат программы:

8 88 93 82 78 35 35 15 59 82 83 73

Max value is 93

© Васильев А.Н.

Копирование и присваивание массивов



```
class CopyArrayDemo{
    public static void main(String[] args){
        // Массив:
        int[] a={1,3,5,7,9};
        // Переменные массивов:
        int[] b,c;
        // Присваивание массивов:
        b=a;
        c=new int[a.length];
        for(int k=0;k<a.length;k++){
            c[k]=a[k];
        }
        a[0]=0;
        b[b.length-1]=0;
        System.out.println("a:\t b:\t c:");
        for(int k=0;k<a.length;k++){
            System.out.println(a[k]+\t "+b[k]+\t "+c[k]);
        }
    }
}
```

Результат программы:

a:	b:	c:
0	0	1
3	3	3
5	5	5
7	7	7
0	0	9

Цикл по массиву



```
class ForeachDemo{
    public static void main(String[] args){
        // Массивы:
        int[] a={1,2,3,4,5};
        char[] b={'A','B','C','D'};
        String[] c={"один","два","три"};
        // Циклы по массивам :
        for(int s:a){
            System.out.print(s+" ");
        }
        System.out.println();
        for(char s:b){
            System.out.print(s+" ");
        }
        System.out.println();
        for(String s:c){
            System.out.print(s+" ");
        }
        System.out.println();
    }
}
```

Цикл по коллекции (массиву):

```
for(тип переменная: массив){
    // команды
}
```

Результат программы:

```
1 2 3 4 5
A B C D
один два три
```

Двумерные массивы



Двумерный массив:

Массив, элементами которого являются переменные массива.
Другими словами, это массив массивов.

Создание двумерного массива:

```
int[][] nums=new int[3][5];
```

```
nums[i][j]
```

↑
Переменная
массива

↑
Массив из 3 строк и 5
столбцов

↑
Элемент на пересечении строки с
индексом *i* и столбца с индексом *j*

```
nums[i]
```

← Строка с индексом *i*

```
nums[i].length
```

← Количество элементов в строке с индексом *i*

```
nums.length
```

← Количество строк в массиве




```
class Array2DDemo{
    public static void main(String[] args){
        int m=2,n=4,i,j;
        // Двумерные массивы:
        int[][] a=new int[3][5];
        char[][] b;
        b=new char[m][n];
        int val=1;
        char s='A';
        // Заполнение и отображение массивов:
        System.out.println("Числовой массив:");
        for(i=0;i<a.length;i++){
            for(j=0;j<a[i].length;j++){
                a[i][j]=val; // Значение элемента массива
                val++;        // Значение для следующего элемента
                System.out.print(a[i][j]+"\\t"); // Отображение элемента
            }
            System.out.println();
        }
        System.out.println("Символьный массив:");
        for(i=0;i<b.length;i++){
            for(j=0;j<b[i].length;j++){
                b[i][j]=s; // Значение элемента массива
                s++;        // Значение для следующего элемента
                System.out.print(b[i][j]+" "); // Отображение элемента
            }
            System.out.println();
        }
    }
}
```

Результат программы:

Числовой массив:

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15

Символьный массив:

A	B	C	D
E	F	G	H

Инициализация двумерного массива

```
class IntitArray2DDemo{
    // Метод для отображения содержимого массива:
    static void show(int[][] a){
        for(int i=0;i<a.length;i++){
            for(int j=0;j<a[i].length;j++){
                System.out.print(a[i][j]+"\\t");
            }
            System.out.println();
        }
    }
    public static void main(String[] args){
        // Массивы:
        int[][] A={{1,2,3},{4,5,6}};
        int[][] B={{1,2},{3,4},{5,6}};
        int[][] C={{1},{2,3},{4,5,6}};
        // Отображение содержимого массивов:
        System.out.println("Массив A:");
        show(A);
        System.out.println("Массив B:");
        show(B);
        System.out.println("Массив C:");
        show(C);
    }
}
```

Результат программы:

Массив A:

1	2	3
4	5	6

Массив B:

1	2
3	4
5	6

Массив C:

1	
2	3
4	5

```

class TransposeDemo{
    // Метод для отображения содержимого массива:
    static void show(int[][] a){
        for(int i=0;i<a.length;i++){
            for(int j=0;j<a[i].length;j++){
                System.out.print(a[i][j]+"\\t");
            }
            System.out.println();
        }
    }
    public static void main(String[] args){
        // Массив (матрица):
        int[][] A={{1,2,3},{4,5,6},{7,8,9}};
        int i,j,n;
        System.out.println("До транспонирования:");
        show(A);
        // Транспонирование:
        for(i=1;i<A.length;i++){
            for(j=0;j<i;j++){
                n=A[i][j];
                A[i][j]=A[j][i];
                A[j][i]=n;
            }
        }
        System.out.println("После транспонирования:");
        show(A);
    }
}

```

До транспонирования:

1	2	3
4	5	6
7	8	9

После транспонирования:

1	4	7
2	5	8
3	6	9

"Рваный" массив



```
class StrangeArrayDemo{
    public static void main(String[] args){
        // Массив с размерами строк:
        int[] n={2,5,3,4};
        int val=1;
        // Массив строк:
        int[][] A=new int[n.length][];
        // Создание строк и их заполнение:
        for(int i=0;i<A.length;i++){
            // Создание новой строки:
            A[i]=new int[n[i]];
            // Заполнение строки:
            for(int j=0;j<A[i].length;j++){
                A[i][j]=val++;
                // Отображение значения элемента:
                System.out.print(A[i][j]+"\\t");
            }
            System.out.println();
        }
    }
}
```

Результат программы:

1	2			
3	4	5	6	7
8	9	10		
11	12	13		

Методы для работы с массивами



Статические методы класса `Arrays`, используемые при работе с массивами:

`Arrays.copyOf(A, num)`



Копия из `num` начальных элементов массива `A`

`Arrays.copyOfRange(A, i, j)`



Копия массива `A`. Задействованы элементы с индексами от `i` (включая) до `j` (не включая)

`Arrays.fill(A, i, j, v)`
или
`Arrays.fill(A, v)`



Заполнение массива `A` значениями `v`. Задействованы элементы с индексами от `i` (включая) до `j` (не включая)

`Arrays.sort(A)`
или
`Arrays.sort(A, i, j)`



Сортировка (в порядке возрастания) массива `A`. Задействованы элементы с индексами от `i` (включая) до `j` (не включая)

Для использования класса `Arrays` добавляем в программу инструкцию `import java.util.Arrays`



Методы для работы с массивами



Статические методы класса `Arrays`, используемые при работе с массивами:

`Arrays.equals(A,B)`

Проверка на предмет равенства массивов `A` и `B`

`Arrays.toString(A)`

Текстовое представление для массива `A`

Для использования класса `Arrays` добавляем в программу инструкцию
`import java.util.Arrays`

Статический метод класса `System`:

`System.arraycopy(A,i,B,j,num)`

Из массива `A` в массив `B` копируется `num` элементов начиная с позиции `i` в массиве `A`, копирование в массив `B` выполняется с позиции `j`

Класс `System` доступен автоматически, поэтому можно не использовать инструкцию
`import java.lang.System`

Программа: методы работы с массивами



```
import java.util.Arrays;
class ArrayMethodsDemo{
    public static void main(String[] args){
        int[] A={10,3,7,11,2,5};           // Исходный массив
        int[] B=Arrays.copyOf(A,A.length); // Копия массива
        // Отображение массивов:
        System.out.println("A: "+Arrays.toString(A));
        System.out.println("B: "+Arrays.toString(B));
        // Сравнение массивов:
        System.out.println("A==B: "+Arrays.equals(A,B));
        Arrays.sort(B);                    // Сортировка массива
        System.out.println("A: "+Arrays.toString(A));
        System.out.println("B: "+Arrays.toString(B));
        System.out.println("A==B: "+Arrays.equals(A,B));
        Arrays.fill(B,2,4,0);              // Заполнение части массива
        System.out.println("B: "+Arrays.toString(B));
        int [] C=Arrays.copyOfRange(A,1,5); // Копия части массива
        System.out.println("C: "+Arrays.toString(C));
        int[] D=new int[4];                // Новый массив
        System.arraycopy(B,1,D,0,C.length); // Копия части массива
        System.out.println("D: "+Arrays.toString(D));
    }
}
```


Программа: результат выполнения



Результат программы:

A: [10, 3, 7, 11, 2, 5]

B: [10, 3, 7, 11, 2, 5]

A==B: true

A: [10, 3, 7, 11, 2, 5]

B: [2, 3, 5, 7, 10, 11]

A==B: false

B: [2, 3, 0, 0, 10, 11]

C: [3, 7, 11, 2]

D: [3, 0, 0, 10]

Домашнее задание



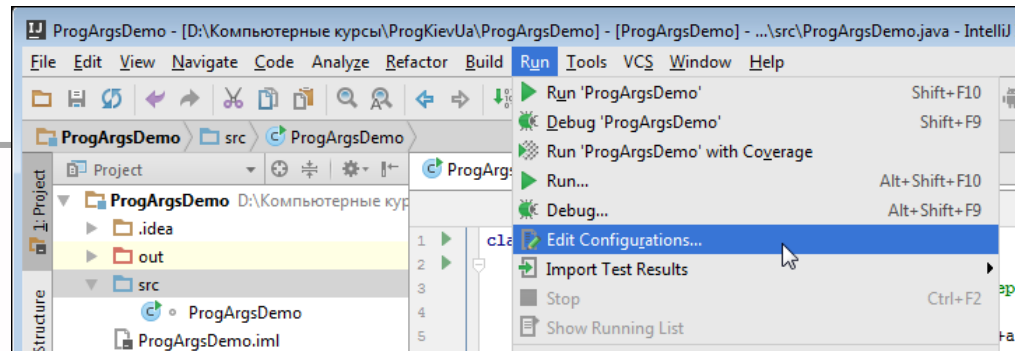
- Напишите программу, в которой создается одномерный массив и заполняется четными числами.
- Напишите программу, в которой создается одномерный массив и заполняется степенями двойки (числа 1, 2, 4, 8, 16, 32, и так далее).
- Напишите программу, в которой создается одномерный массив и заполняется числами Фибоначчи (первые два равны 1, а каждое следующее - сумма двух предыдущих).
- Напишите программу, в которой создается одномерный массив, заполняется случайными числами, а затем сортируется в порядке убывания значений элементов.
- Напишите программу для поиска элемента массива с минимальным значением.
- Напишите программу, в которой создается двумерный массив и построчно заполняется нечетными числами.



Аргументы командной строки



```
class ProgArgsDemo{
    public static void main(String[] args){
        // Определение количества переданных параметров:
        System.out.println("При вызове в программу передано "+args.length+"
        параметров:");
        // Отображение значений параметров:
        for(int k=0;k<args.length;k++){
            System.out.println((k+1)+"-й параметр: "+args[k]);
        }
    }
}
```



Результат программы:

При вызове в программу
передано 5 параметров:

- 1-й параметр: первый
- 2-й параметр: 123
- 3-й параметр: A
- 4-й параметр: 32.5
- 5-й параметр: последний

© Васильев А.Н.

Информация к размышлению



```
class ObjectArrayDemo {  
    public static void main(String[] args){  
        // Массив объектов класса Object:  
        Object[] objs=new Object[3];  
        // Значения разных типов:  
        objs[0]=123;  
        objs[1]='A';  
        objs[2]="Java";  
        // Отображение содержимого:  
        for(int k=0;k<objs.length;k++){  
            System.out.println(objs[k]);  
        }  
        System.out.println();  
        // Новые значения элементов:  
        objs[0]=(int)objs[0]+1;  
        objs[1]="Hello";  
        objs[2]=32.1;  
        // Отображение содержимого:  
        for(int k=0;k<objs.length;k++){  
            System.out.println(objs[k]);  
        }  
    }  
}
```

Результат:

123

A

Java

124

Hello

32.1

```

import javax.swing.*;
import static javax.swing.JOptionPane.*;
class UsingListDemo{
    public static void main(String[] args){
        String[] txt={"Волк","Лиса","Медведь","Енот"};           // Названия животных
        // Названия файлов с изображениями животных:
        String[] files={"wolf.jpg","fox.jpg","bear.jpg","raccoon.jpg"};
        String folder="d:/books/pictures/";                      // Путь к файлам
        String msg="Кого показать?";                              // Текст в окне
        String title="В мире животных";                          // Название окна
        ImageIcon img=new ImageIcon(folder+"giraffe.png");        // Пиктограмма
        String animal;                                           // Имя выбранного животного
        animal=(String)showInputDialog(null,
            msg,                                                  // Текст над раскрывающимся списком
            title,                                                // Название окна
            PLAIN_MESSAGE,   // Тип окна (не влияет на результат)
            img,            // Пиктограмма, отображаемая в окне
            txt,             // Элементы раскрывающегося списка
            txt[0]           // Выбранный по умолчанию элемент
        );
        if(animal==null){                                         // Если пользователь отменил ввод
            System.exit(0);                                       // Завершение выполнения программы
        }
        for(int k=0;k<txt.length;k++){                          // Определение пиктограммы
            if(animal.equals(txt[k])){
                img=new ImageIcon(folder+files[k]);              // Создание пиктограммы
                break;                                           // Завершение оператора цикла
            }
        }
        // Отображение диалогового окна:
        showMessageDialog(null,
            img,          // Изображение
            animal,        // Название окна
            PLAIN_MESSAGE // Тип окна
        );
    }
}

```

Внимание!!!

**В каталоге d:/books/pictures/
должны находиться файлы с
изображениями животных!!!**

Результат выполнения
программы

