

ОСНОВЫ ПРОГРАММИРОВАНИЯ НА Java

МАССИВЫ



Массив

Массив - структура данных, представляющая собой набор элементов одного типа, имеющий общее имя

Массив определяется именем и порядковым номером каждого своего элемента.

Обычно порядковый номер элемента называют индексом

Объявление массива (1)

Размер или длина массива — это общее количество элементов в массиве.

Размер массива задаётся при создании массива и не может быть изменён в дальнейшем

```
тип <имя массива> [размер1][размер2]...[размер N];
```

Одномерные массивы (вектор):

```
тип<имя массива>[размер1];
```

- **int arrayInt[10];**

Двумерные массивы:

```
тип<имя массива>[размер1][размер2];
```

```
float arrayFloat[5][5]; (матрица)
```

Объявление массива (2)

- нумерация элементов начинается с нуля
- первый элемент массива - это элемент с нулевым номером.

```
int a[100];
```

Это означает, что массив содержит следующие элементы: $a[0]$, $a[1]$, $a[2]$, ..., и так далее, $a[99]$ – последний элемент массива a .

Объявление

```
тип[] имя;  
тип имя[];
```

Создание массива (инициализация)

`new тип_данных[количество_элементов]`

где `new` - ключевое слово, выделяющее память для указанного в скобках количества элементов.

`int[] numbers = new int[10];`

создается массив из 10 элементов `int` и каждый элемент по умолчанию равен нулю(!!!).

Значения по умолчанию

- для числовых типов (в том числе для типа `char`) это число 0,
- для типа `boolean` это значение `false`,
- а для остальных объектов это значение `null`

Пример. Создание массива

```
int numbers[];  
numbers = new int[10]; // массив из 10 чисел  
  
int[] numbersArray = new int[10];  
  
//можно задать конкретные значения для элементов массива при его создании  
int[] nums1 = new int[]{1, 4, 3, 5, 1};  
int[] nums2 = {1, 4, 3, 5, 1};  
  
int[] nums3 = new int[3];  
nums3[0] = 1;  
nums3[1] = 2;  
nums3[2] = 4;  
  
//Размерность массивов предпочтительнее задавать с помощью типизированных констант  
final int SIZE=10; ✓  
int[] arrayInt=new int[SIZE];
```

Примеры

```
int[] numbersArray = new int[10];
for (int i = 0; i < numbersArray.length; i++) {
    numbersArray[i] = (int) (Math.random() * 100);
    System.out.print(numbersArray[i] + " ");
}
```

```
int[] numbersArray = new int[10];
Scanner scanner=new Scanner(System.in);
for (int i = 0; i < numbersArray.length; i++) {
    System.out.printf("Enter the %d number: ",i+1);
    numbersArray[i]=scanner.nextInt();
}
```

```
for (int i = 0; i < numbersArray.length/2; i++) {
    int temp=numbersArray[i];
    numbersArray[i]=numbersArray[numbersArray.length-i-1];
    numbersArray[numbersArray.length-i-1]=temp;
}
```

```
int sum=0;
for (int i = 0; i < numbersArray.length; i++) {
    sum+=numbersArray[i];
}
System.out.println("Sum="+sum);
```

```
int min=Integer.MAX_VALUE; ✓
for (int i = 0; i < number
```

Max?

```
int min=numbersArray[0];
for (int i = 1; i < numbersArray.length; i++) {
    if(min>numbersArray[i]){
        min=numbersArray[i];
    }
}
System.out.println("Minimum: "+min);
```

for each

это разновидность цикла for

используется для перебора элементов массива или коллекции

```
for (Type varName : collectionName)
```

```
int[] numbersArray = new int[10];  
for (int element : numbersArray) {  
    System.out.println(element);  
}
```


Пример

```
int sum = 0;
for (int element : numbersArray) {
    sum += element;
}
System.out.println("Sum=" + sum);

int min = Integer.MAX_VALUE;
for (int element : numbersArray) {
    if (min > element) {
        min = element;
    }
}
System.out.println("Minimum: " + min);
```

```
60 20 35 95 39 23 35 87 35 71
Sum=500
Minimum: 20
```

Вопросы



Задание

1. Найти в одномерном массиве те элементы, значение которых меньше среднего арифметического, взятого от всех элементов массива.
2. В одномерном массиве целых чисел определить два наименьших элемента.

Спасибо за внимание
