

IT ШКОЛА SAMSUNG

Тема: *Функции в Java.*

Мария Александровна Сокольская

Задание



Вася, Петя и Аня решили найти максимальный элемент в массиве, но так как каждый из них был яркой индивидуальностью, то каждый решил выбрать собственный массив данных. Сможете ли вы написать программу, которая находит максимум **в каждом из массивов** данных Васи, Пети и Ани?

Решение



1. Копируем код цикла, ищущего максимум в массиве три раза, подставляем нужные массивы.
2. Копируем трижды код генерации массивов.

```
public static void main(String[] args) {  
    Random rand = new Random();  
    // Создаем Васин массив  
    int[] vas_ar = new int[10];  
    // Заполняем Васин массив случайными значениями  
    for (int i = 0; i < vas_ar.length; i++){  
        vas_ar[i] = rand.nextInt(200);  
        System.out.print("\t" + vas_ar[i]);  
    }  
    // Создаем Петин массив  
    int[] pet_ar = new int[10];  
    // Заполняем Петин массив случайными значениями  
    for (int i = 0; i < pet_ar.length; i++){  
        pet_ar[i] = rand.nextInt(100);  
        System.out.print("\t" + pet_ar[i]);  
    }  
}
```

Решение



```
// Создаем Анин массив
int[] an_ar = new int[10];
// Заполняем Анин массив случайными значениями
for (int i = 0; i < an_ar.length; i++){
    an_ar[i] = rand.nextInt(300);
    System.out.print("\t" + an_ar[i]);
}

// Создаем переменную для максимума в Васином массиве
int vas_max = vas_ar[0];
// Ищем максимум в Васином массиве
for (int i = 0; i < vas_ar.length; i++){
    if (vas_ar[i] > vas_max){
        vas_max = vas_ar[i];
    }
}
// Выводим максимум Васиного массива
System.out.println(vas_max);
```

Решение



```
// Создаем переменную для максимума в Петинном массиве
int pet_max = pet_ar[0];
// Ищем максимум в Петинном массиве
for (int i = 0; i < pet_ar.length; i++){
    if (pet_ar[i] > pet_max){
        pet_max = pet_ar[i];
    }
}
// Выводим максимум Петинного массива
System.out.println(pet_max);
// Создаем переменную для максимума в Анином массиве
int an_max = an_ar[0];
// Ищем максимум в Анином массиве
for (int i = 0; i < an_ar.length; i++){
    if (an_ar[i] > an_max){
        an_max = an_ar[i];
    }
}
// Выводим максимум Аниного массива.
System.out.println(an_max);
}
```

Как сделать программу короче?



Хорошо бы написать код поиска максимума и заполнения массива только 1 раз, а потом попросить программу выполнить его с нужным массивом.

```
for (int i = 0; i < an_ar.length; i++){  
    if (an_ar[i] > an_max){  
        an_max = an_ar[i];  
    }  
}
```



```
int an_max = maxArray;
```

Как сделать программу короче?



Хорошо бы написать код поиска максимума и заполнения массива только 1 раз, а потом попросить программу выполнить его с нужным массивом.

```
for (int i = 0; i < an_ar.length; i++){  
    if (an_ar[i] > an_max){  
        an_max = an_ar[i];  
    }  
}
```



```
int an_max = maxArray();
```

*Отличие команды
(функции) от
простой переменной*

Как сделать программу короче?



Хорошо бы написать код поиска максимума и заполнения массива только 1 раз, а потом попросить программу выполнить его с нужным массивом.

```
for (int i = 0; i < an_ar.length; i++){  
    if (an_ar[i] > an_max){  
        an_max = an_ar[i];  
    }  
}
```



```
int an_max = maxArray(an_ar);
```

*Сообщим максимум
какого массива
искать*

Функции



Если выделить отдельным **специально оформленным** блоком поиск максимума в массиве, то мы получим **функцию**.

Функция – это логически завершенный фрагмент кода программы, имеющий собственное имя и решающий часть задачи.

Для классов функции называют **методами**.

Синтаксис определения функций (методов)



```
тип_данных_которые_метод(функция)_вернет  
имя_метода(функции)  
(данные_передающиеся_методу){  
    здесь размещаем тот код,  
    который мы хотим, чтобы выполнялся  
    при вызове этого метода  
}
```

Функция поиска максимума



```
maxArray(int[] ar){  
    for(int i = 0; i < ar.length; i++){  
        if(ar[i] > max){  
            max = ar[i];  
        }  
    }  
}
```

Какие ошибки в
этом фрагменте?



*В объектно-ориентированных языках функции обязательно определены (прикреплены) в каком-либо классе, эта особенность (принадлежность функции классу) отразилась в том, что функцию называют **методом**.*

Функция поиска максимума



```
maxArray(int[] ar){  
    for(int i = 0; i < ar.length; i++){  
        if(ar[i] > max){  
            max = ar[i];  
        }  
    }  
}
```

1. Откуда взялась переменная max?
2. Как максимум вернется в функцию main?

Функция поиска максимума

```
maxArray(int[] ar){  
    for(int i = 0; i < ar.length; i++){  
        if(ar[i] > max){  
            max = ar[i];  
        }  
    }  
}
```

```
int maxArray(int[] ar){  
    int max = ar[0];  
    for(int i = 0; i < ar.length; i++){  
        if(ar[i] > max){  
            max = ar[i];  
        }  
    }  
    return max;  
}
```

Функция поиска максимума

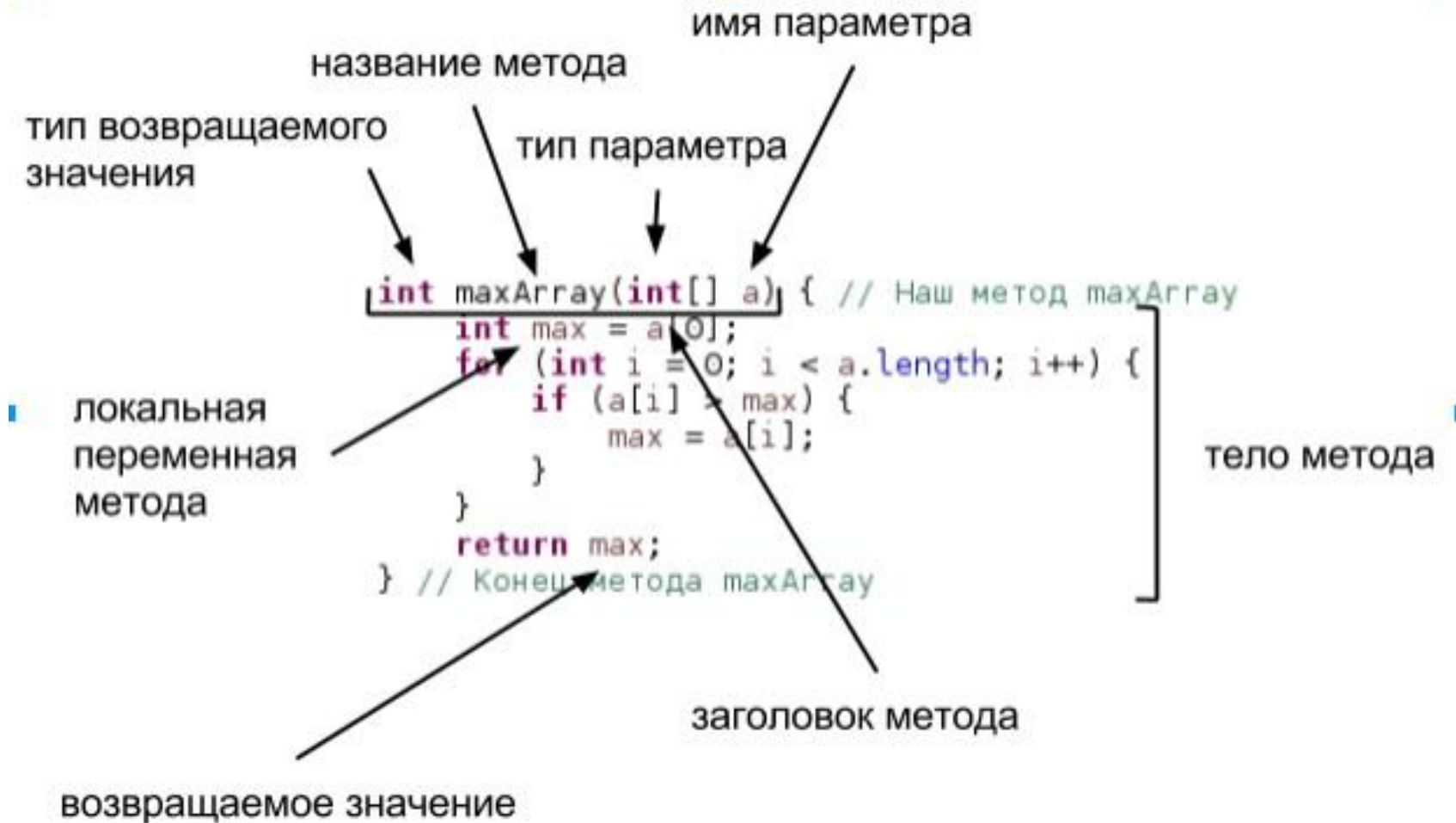
Тип
возвращаемого
значения

Объявление и
инициализация
переменной max

Точка возврата из
функции и описание
возвращаемого
значения
Операторов return
может быть
несколько

```
int maxArray(int[] ar){  
    int max = ar[0];  
    for(int i = 0; i < ar.length; i++){  
        if(ar[i] > max){  
            max = ar[i];  
        }  
    }  
    return max;  
}
```

«Строение» функции



Собственные методы можно создавать внутри класса, в любом месте.

Создание метода (функции) класса



```
public class Max {  
  
    static int maxArray(int[] ar){ // Наш метод maxArray  
        int max = ar[0];  
        for(int i = 0; i < ar.length; i++){  
            if(ar[i] > max){  
                max = ar[i];  
            }  
        }  
        return max;  
    } // конец метода maxArray  
  
    public static void main(String[] args) { // Начало метода main  
        ...  
    }
```


Создание метода (функции) класса



```
public class Max {
```

```
    static int maxArray(int[] ar){ // Наш метод maxArray
    int max = ar[0];
    for(int i = 0; i < ar.length; i++){
        if(ar[i] > max){
            max = ar[i];
        }
    }
    return max;
} // конец метода maxArray
```

```
    public static void main(String[] args) { // Начало метода main
```

```
    ...
}
```

Задание



Завершите задачу с описанием и применением функции поиска максимума

Замечание: метод `main` помечен как **статический** модификатором **`static`**. Из подобных методов, методы этого же класса, можно вызывать только, если они тоже статические. Нужно добавить модификатор **`static`** перед объявлением метода.



Задание

Создать метод для заполнения массива. Этот метод должен принимать массив и вернуть его уже заполненным. Применить второй метод в программе.

Подсказка:

```
static int[] randArray(int[] ar){  
    // Метод randArray заполнения массива  
    ...  
}
```



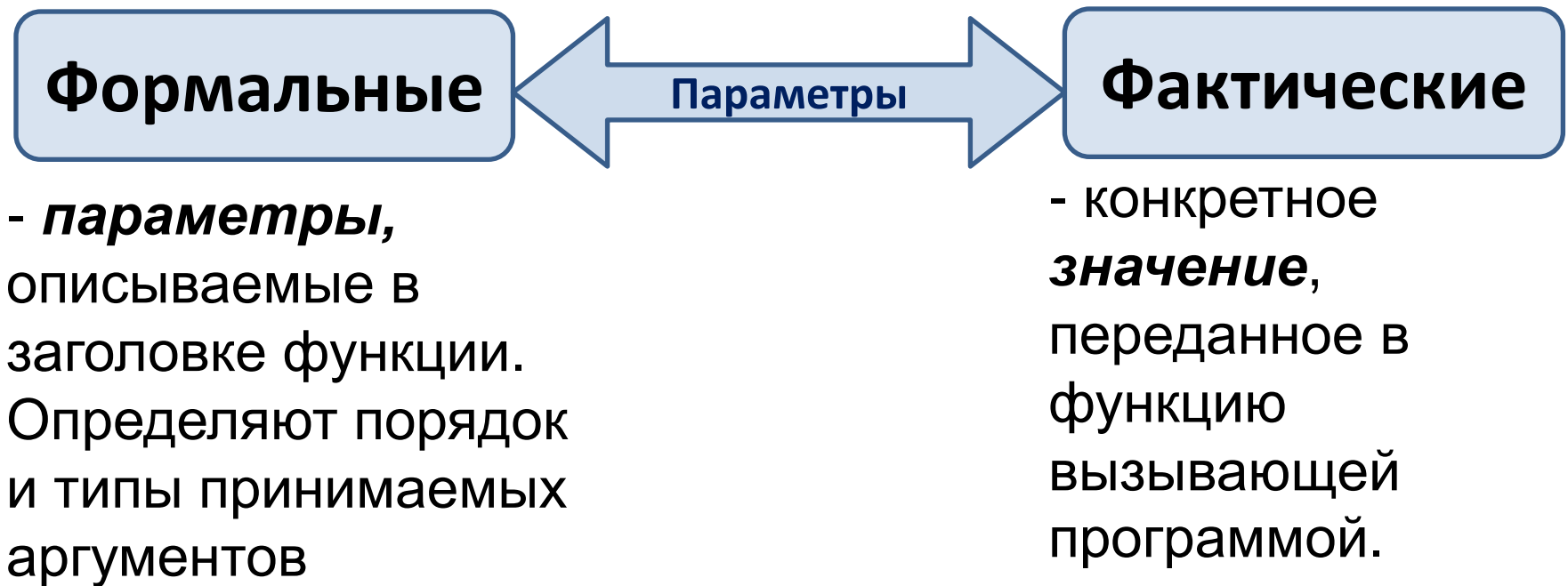
Параметры метода

Предположим, Вася, Петя и Аня не просто хотят иметь свой массив положительных целых чисел, но еще и чтобы диапазон возможных значений в них был разный.

Т.е. необходимо сделать так, чтобы Вася, Петя и Аня в методе **randArray** могли задавать каждый свое значение переменной **maxRange**.

Для решения нужно локальную переменную **maxRange** определить как параметр метода.

Формальные и фактические параметры



Фактический параметр может быть константой, переменной или более сложным выражением. Независимо от типа фактического параметра он в начале вычисляется, а затем его величина передается функции (методу).

Задание



1. Напишите метод для вывода массива, метод принимает массив и разделитель. Заголовок метода:

```
static void printArray(int[] a, char delimiter)
```

2. Решите задачи №307, 310, 199.

Использование функций – обязательное условие для решения.

Домашнее задание



Закончить решение задач, предложенных на занятии