

Plan

2023-09-19

1. Homework Review
2. Array

-
1. Разбор домашнего задания
 2. Массив

Theory

► **Open in English**

▼ **На русском**

Массивы

1. Зачем нужны массивы данных?

Главная функция массивов - **хранение больших объемов информации.**

Примеры массивов, которые вам известны из жизни:

- список сотрудников
- перечень дел/поручений
- телефонная книжка
- список покупок
- библиотека (книги)
- архив документов
- фотоархивы
- склады
- каталог товаров
- аудиотека
- списки покупателей, клиентов
- реестр недвижимости (аренда, продажа, покупка)
- перечень имущества, активов
- кадастр по земельным участкам
- счета клиентов в банках
- ...

Есть переменная и тип данных => как их сохранять???? `int X = 1000; int Y = 2000; int Z = 2500; ...` так неудобно.

Удобно так, данные можно превратить в массив: `int X[0], X[1], X[2], ... double Y[0], Y[1], Y[2] ..., Y[30]` - длина массива 31

Определение:

- Массив – это объект-контейнер, содержащий **фиксированное количество** значений **одного типа**.
- В массиве могут содержаться данные **ТОЛЬКО ОДНОГО типа**.
- Тип того, что хранится в массиве - это типы данных, которые у нас есть в языке Java, других типов нет.

Главные характеристики массива:

- **тип** помещённых в него данных (`int`, `double`, `boolean`, `long`, `String`, ...)
- имя массива (как имяПеременной)
- **длина** (натуральное число, не может быть отрицательным) = размер массива (количество ячеек, длина), всегда имеет **тип int**.

ВАЖНО! - изменить длину массива после его создания нельзя.

В массивах, как и везде в Java, **элементы нумеруются с нуля**.

2. Синтаксис объявления массива:

Объявление массива Примеры

```
dataType[] arrayName; int[] myArray;
```

```
dataType arrayName[]; int myArray[];
```

- В обоих случаях `dataType` — тип переменных в массиве.
- Шаг **1**: определение/задание **типа хранимых данных** (в массиве хранятся данные **ОДНОГО** типа),
- Шаг **2**: объявление и инициация массива `Тип[] имяМассива = new Тип[размер];`

Примеры:

```
int[] arrayNum = new int[5]; // определили массив целых чисел, длина массива = 5  
arrayNum[0] - это первый элемент массива, arrayNum[4] - это последний элемент
```

```
String[] arrayCity = new String[10]; // определили массив строк, длина массива = 10  
arrayCity[0] - это первый элемент массива, arrayCity[9] - это последний элемент
```

```
double[] arrayResults = new double[100]; // определили массив дробных чисел, длина массива = 100 arrayResults[0] - это первый элемент массива, arrayResults[99] - это последний элемент
```

1. Заполнение массива После процедуры создания (инициализации) массива он наполнен **значениями по умолчанию**.

- для int все элементы = 0
- для float, double все элементы = 0.0
- для String все элементы = "null"
- для char все элементы = '\0'
- для boolean все элементы = false

Прямое (ручное) заполнение массива:

```
int[] myArray = new int[3];
```

массив называется myArray, тип данных - int , длина массива = 3

```
myArray[0] = 5;  
myArray[1] = -19;  
myArray[2] = 155;
```

```
String[] directions = new String[4];
```

```
directions = {"left", "right", "up", "down"};  
direction[0] = "left"  
direction[1] = "right"  
.....  
direction[3] = "down"
```

```
direction[4] = "up-side-down" - ошибка, попытка обращения к несуществующему элементу  
direction[5] - ошибка, попытка обращения к несуществующему элементу массива,  
out of range - выход за пределы массива
```

```
int[] marks = {2, 5, 4, 3, 5, 3, 5, 4, 1}; // так тоже можно задавать массив  
marks.length - возвращает длину массива marks
```

```
double[] temperature = {16.0, 18.0, 17.5, 22.6, 25.0, 28.0, 26.8}  
temperature.length - возвращает длину массива temperature
```

4. Доступ к элементам массива: Доступ к элементу массива получают по **имени массива** и **индексу элемента** в нем: arrayName[n] - это n-й элемент в массиве с именем arrayName, нумерация начинается с 0

Стандартные задачи и операции с массивами:

- поиск элемента массива;

- добавление нового элемента (в начало, в конец, в середину);
- редактирование(изменение) элемента массива;
- удаление элемента массива;
- поиск максимального/минимального значения;
- сортировка по возрастанию/убыванию;
- удаление дубликатов;
- сдвиг массива;
- удаление пустых (нулевых) элементов.

Постепенно мы пройдем все эти пункты.

Homework

► English

▼ На русском

Задача 1

Найти максимальный элемент в массиве

Создайте массив из 12 случайных целых чисел из отрезка от **-15** до **15**. Определите какой элемент является в этом массиве максимальным и сообщите индекс его последнего вхождения в массив.

Задача 2

Заменить каждый элемент массива с нечетным индексом на ноль.

1. Создайте массив из 8 случайных целых чисел из отрезка от **-3** до **10**
2. Выведите массив на экран в строку
3. Замените каждый элемент с нечётным индексом на ноль
4. Снова выведете массив на экран на отдельной строке

Задача 3 *

Среднее арифметическое элементов массива

1. Создайте 2 массива из 5 случайных целых чисел из отрезка от **0** до **5** каждый
2. Выведите массивы на экран в двух отдельных строках
3. Посчитайте среднее арифметическое элементов каждого массива и сообщите, для какого из массивов это значение оказалось больше (либо сообщите, что их средние арифметические равны)

Code

code/HwSolution_10/src/DepositForLoop.java

```
/**
 * @author Andrej Reutow
 * created on 17.09.2023
 * Вычисление конечной суммы вклада с учетом процентов (используя цикл for)
 */
public class DepositForLoop {
    public static void main(String[] args) {
        float initialDeposit = 100;
        int months = 1;
        float interestRate = 12;

        for (int month = 1; month <= months; month++) {
            initialDeposit += initialDeposit * (interestRate / 100);
        }

        System.out.println("Final deposit after 1 month: " + initialDeposit
    }
}
```

code/HwSolution_10/src/DepositWhileLoop.java

```
/**
 * @author Andrej Reutow
 * created on 17.09.2023
 * Вычисление конечной суммы вклада с учетом процентов (используя цикл while)
 */
public class DepositWhileLoop {
    public static void main(String[] args) {
        float initialDeposit = 100;
        int months = 10;
        float interestRate = 12;
        int currentMonth = 1;

        calculateLoop(initialDeposit, months, interestRate, currentMonth);
        System.out.print("\n-----");
        calculateWithoutLoop(initialDeposit, months, interestRate, currentMonth);
    }
}
```

```
public static void calculateLoop(float initialDeposit,
                                int months,
                                float interestRate,
                                int currentMonth) {

    while (currentMonth <= months) {
        System.out.println(currentMonth + " месяц");

        float summ = initialDeposit * (interestRate / 100);
        System.out.println("Сумма начисления " + summ);

        initialDeposit += summ;
        System.out.println("Текущий депозит " + initialDeposit);
        currentMonth++;
    }

    System.out.println("Final deposit after " + months + " month: " + i
}

public static void calculateWithoutLoop(float initialDeposit,
                                         int months,
                                         float interestRate,
                                         int currentMonth) {

    // 1 месяц
    if (currentMonth <= months) {
        System.out.println(currentMonth + " месяц");

        float summ = initialDeposit * (interestRate / 100);
        System.out.println("Сумма начисления " + summ);

        initialDeposit += summ;
        System.out.println("Текущий депозит " + initialDeposit);
        currentMonth++;
    }

    // 2 месяц
    if (currentMonth <= months) {
        System.out.println(currentMonth + " месяц");

        float summ = initialDeposit * (interestRate / 100);
        System.out.println("Сумма начисления " + summ);
```

```
        initialDeposit += summ;
        System.out.println("Текущий депозит " + initialDeposit);
        currentMonth++;
    }

    // 3 месяц
    if (currentMonth <= months) {
        System.out.println(currentMonth + " месяц");

        float summ = initialDeposit * (interestRate / 100);
        System.out.println("Сумма начисления " + summ);

        initialDeposit += summ;
        System.out.println("Текущий депозит " + initialDeposit);
        currentMonth++;
    }

    // 4 месяц
    if (currentMonth <= months) {
        System.out.println(currentMonth + " месяц");

        float summ = initialDeposit * (interestRate / 100);
        System.out.println("Сумма начисления " + summ);

        initialDeposit += summ;
        System.out.println("Текущий депозит " + initialDeposit);
        currentMonth++;
    }

    // 5 месяц
    if (currentMonth <= months) {
        System.out.println(currentMonth + " месяц");

        float summ = initialDeposit * (interestRate / 100);
        System.out.println("Сумма начисления " + summ);

        initialDeposit += summ;
        System.out.println("Текущий депозит " + initialDeposit);
        currentMonth++;
    }

    // 6 месяц
```

```
        if (currentMonth <= months) {
            System.out.println(currentMonth + " месяц");

            float summ = initialDeposit * (interestRate / 100);
            System.out.println("Сумма начисления " + summ);

            initialDeposit += summ;
            System.out.println("Текущий депозит " + initialDeposit);
            currentMonth++;
        }

        System.out.println("Final deposit after " + months + " month: " + i
    }
}
```

code/HwSolution_10/src/MultiplicationTable.java

```
// Напишите программу, которая выводит на консоль таблицу умножения
public class MultiplicationTable {
    public static void main(String[] args) {
        for (int i = 1; i <= 9; i++) {
            for (int j = 1; j <= 9; j++) {
                System.out.print(i * j + "\t");
            }
            System.out.println();
        }
    }
}
```

code/HwSolution_10/src/PopulationGrowth.java

```
/**
 * @author Andrej Reutow
 * created on 17.09.2023
 * Расчет численности населения
 */
public class PopulationGrowth {
    public static void main(String[] args) {
        int population = 14_000_000;
        int birthRate = 14;
        int deathRate = 8;
        int years = 5;
```



```

        populationCalcLoop(population, birthRate, deathRate, years);
        System.out.println("Without loop");
        populationCalcWithoutLoop(population, birthRate, deathRate, years);
    }

    public static void populationCalcLoop(int population, int birthRate, int deathRate, int years) {
        for (int year = 1; year <= years; year++) {
            int births = (population * birthRate) / 1000;
            int deaths = (population * deathRate) / 1000;
            int grow = (births - deaths);
            population += grow;
            System.out.println("Год " + year);
            System.out.println("Рождаемость " + births + " в " + year + " году");
            System.out.println("Смертность " + deaths + " в " + year + " году");
            System.out.println("Прирост " + grow + " в " + year + " году");
            System.out.println("Население " + population + " в " + year + " году");
            System.out.println();
        }

        System.out.println("Population after " + years + " years: " + population);
    }

    public static void populationCalcWithoutLoop(int population, int birthRate, int deathRate, int years) {
        int currentYear = 1;
        // 1 год
        if (currentYear <= years) {
            int births = (population * birthRate) / 1000; // 196_000 ((14_000 * 14))
            int deaths = (population * deathRate) / 1000; // 11_2000 ((14_000 * 8))
            int grow = (births - deaths); // 84_000
            population += grow; // 14_084_000 (14_000_000 + 84_000)
            System.out.println("Год " + currentYear);
            System.out.println("Рождаемость " + births + " в " + currentYear);
            System.out.println("Смертность " + deaths + " в " + currentYear);
            System.out.println("Прирост " + grow + " в " + currentYear);
            System.out.println("Население " + population + " в " + currentYear);
            System.out.println();
            currentYear++;
        }
        // 2 год
        if (currentYear <= years) {
            // population 14_084_000
            int births = (population * birthRate) / 1000; // 197_176 ((14_084 * 14))
            int deaths = (population * deathRate) / 1000; // 112_672 ((14_084 * 8))

```

```
int grow = (births - deaths); // 84_504 (197_176 - 112_672)
population += grow; // 14_168_504 (14_084_000 + 84_504)
System.out.println("Год " + currentYear);
System.out.println("Рождаемость " + births + " в " + currentYear);
System.out.println("Смертность " + deaths + " в " + currentYear);
System.out.println("Прирост " + grow + " в " + currentYear + " ");
System.out.println("Население " + population + " в " + currentYear);
System.out.println();
currentYear++;
}
// 3 год
if (currentYear <= years) {
    int births = (population * birthRate) / 1000;
    int deaths = (population * deathRate) / 1000;
    int grow = (births - deaths);
    population += grow;
    System.out.println("Год " + currentYear);
    System.out.println("Рождаемость " + births + " в " + currentYear);
    System.out.println("Смертность " + deaths + " в " + currentYear);
    System.out.println("Прирост " + grow + " в " + currentYear + " ");
    System.out.println("Население " + population + " в " + currentYear);
    System.out.println();
    currentYear++;
}
// 4 год
if (currentYear <= years) {
    int births = (population * birthRate) / 1000;
    int deaths = (population * deathRate) / 1000;
    int grow = (births - deaths);
    population += grow;
    System.out.println("Год " + currentYear);
    System.out.println("Рождаемость " + births + " в " + currentYear);
    System.out.println("Смертность " + deaths + " в " + currentYear);
    System.out.println("Прирост " + grow + " в " + currentYear + " ");
    System.out.println("Население " + population + " в " + currentYear);
    System.out.println();
    currentYear++;
}
// 5 год
if (currentYear <= years) {
    int births = (population * birthRate) / 1000;
    int deaths = (population * deathRate) / 1000;
    int grow = (births - deaths);
```

```

        population += grow;
        System.out.println("Год " + currentYear);
        System.out.println("Рождаемость " + births + " в " + currentYear);
        System.out.println("Смертность " + deaths + " в " + currentYear);
        System.out.println("Прирост " + grow + " в " + currentYear + " ");
        System.out.println("Население " + population + " в " + currentYear);
        System.out.println();
        currentYear++;
    }
    // 6 год
    if (currentYear <= years) {
        int births = (population * birthRate) / 1000;
        int deaths = (population * deathRate) / 1000;
        int grow = (births - deaths);
        population += grow;
        System.out.println("Год " + currentYear);
        System.out.println("Рождаемость " + births + " в " + currentYear);
        System.out.println("Смертность " + deaths + " в " + currentYear);
        System.out.println("Прирост " + grow + " в " + currentYear + " ");
        System.out.println("Население " + population + " в " + currentYear);
        System.out.println();
        currentYear++;
    }
}
}

```

code/Lesson_11/src/MultiplyTable.java

```

/**
 * @author Andrej Reutow
 * created on 19.09.2023
 */
public class MultiplyTable {

    public static void main(String[] args) {

        // i = 1
        // i = 1, j = 1 - 1 * 1
        // i = 1, j = 2 - 1 * 2
        // i = 1, j = 3 - 1 * 3
        // i = 1, j = ...
        // i = 1, j = 9
        // i = 2
        // i = 2, j = 1 - 2 * 1
    }
}

```

```

        // i = 2, j = 2 - 2 - 2
        // i = 2, j = 3
        // i = 2, j = ...
        // i = 2, j = 9
    // i = 3
        // i = 3, j = 1 - 3 * 1
        // i = 3, j = 2 - 3 * 2
        // i = 3, j = 3
        // i = 3, j = ...
        // i = 3, j = 9

    for (int i = 1; i < 10; i++) {
        for (int j = 1; j < 10; j++) {
            System.out.print(i * j + "\t");
        }
        System.out.println();
    }
}
}

```

code/Lesson_11/src/Array.java

```

/**
 * @author Andrej Reutow
 * created on 19.09.2023
 */
public class Array {

    public static void main(String[] args) {
        String garage[] = new String[11];        // 0 ... 10
        int size = garage.length;
        // System.out.println("Размер массива: " + size);
        // int intArray[] = new int[100];        // 0 ... 99
        // boolean booleanArray[] = new boolean[100];
        // double doubleArray[] = new double[100];
        // char charArray[] = new char[100];
        // LocalDate LocalDate[] = new LocalDate[1000];

        garage[5] = "B-AR 1555";
        garage[0] = "M-TT 2012";
        garage[10] = "WN-GG 777";
        garage[6] = "OS-BN 666";
        String carNumber = "S-WG 2012";
        garage[4] = carNumber;
    }
}

```

```

    printArray(garage);
    System.out.println();

    int parkNumber = 5;
    garage[parkNumber] = carNumber;

    int requestParkNumber = 2;
    String result = garage[requestParkNumber];
    // System.out.println(result); // null

    // System.out.println(garage[0]);

    printArray(garage);
}

// печать массива
public static void printArray(String[] arr) {
    for (int i = 0; i < arr.length; i++) {
        System.out.println("На парковочном месте: " + i + " находится м
    }
}
}

```

code/Lesson_11/src/ArrayMaxElement.java

```

/**
 * @author Andrej Reutow
 * created on 19.09.2023
 */
public class ArrayMaxElement {

    // Найти максимальный элемент в массиве.
    public static void main(String[] args) {
        long[] longArray = {1, 2, 5, 8, 100, 9576, -1};

        for (int index = 0; index < longArray.length; index++) {

            //          if (){
            //
            //          }

        }
    }
}

```

```
public static void printArray(long[] arr) {
    for (int i = 0; i < arr.length; i++) {
        System.out.println("Index " + i + " ,value: " + arr[i]);
    }
}
}
```

code/Lesson_11/src/ArraySum.java

```
/**
 * @author Andrej Reutow
 * created on 19.09.2023
 */
public class ArraySum {

    public static void main(String[] args) {
        int[] ints = new int[5];
        ints[0] = 100;
        ints[1] = 5;
        ints[3] = 7;
        ints[4] = 8;

        // 100 + 5 + 0 + 7 + 8 = 120
        int result = 0;
        for (int i = 0; i < ints.length; i++) {
            result += ints[i]; // ints[0] = 1, ints[1] = 5, ints[2] = 5
        }

        System.out.println(result);
        printArray(ints);
    }

    // печать массива
    public static void printArray(int[] arr) {
        for (int i = 0; i < arr.length; i++) {
            System.out.println("Индекс: " + i + " значение: " + arr[i]);
        }
    }
}
```