

# *Многомерные массивы*



*Многомерный массив – это массив, в котором каждый элемент является другим массивом.*

*На практике редко используются массивы с более чем третьим уровнем вложенности.*



# Общий вид многомерных массивов

*Тип []...[размерностьN] Имя = new Тип []...[Кол-воЭлементовМассиваN]*

*Тип – тип элементов массива.*

*РазмерностьN – размерность массива (количество пар скобок).*

*Имя – имя массива.*

*Кол-воЭлементовМассиваN – количество элементов в каждом из измерений массива.*



# Объявление, создание и инициализация двумерных массивов

```
public class Task {  
    public static void main(String[] args) {  
        int [][] array1 = new int[3][3];  
        int array2 [][] = new int[3][3];  
    }  
}
```

*В первой [] указывается количество строк в массиве.*

*Во второй [] указывается количество столбцов в массиве.*



```
public class Task {  
    public static void main(String[] args) {  
        int [][]array3 = {{1,2,3}, {4,5,6}};  
        int [][]array4 = new int[2][3];  
        array4[0][0] = 1;  
        array4[0][1] = 2;  
        array4[0][2] = 3;  
        array4[1][0] = 4;  
        array4[1][1] = 5;  
        array4[1][2] = 6;  
    }  
}
```

Индексы столбцов

	0	1	2
0	1	2	3
1	4	5	6

Индексы строк



# *Вывод двумерного массива*

```
public class Task {  
    public static void main(String[] args) {  
        int [][]array = {{1,2,3}, {4,5,6}};  
        for(int i = 0; i<array.length; i++) {  
            for(int j = 0; j<array[0].length; j++) {  
                System.out.print(array[i][j]+"\\t");  
            }  
            System.out.println();  
        }  
    }  
}
```



# Вывод двумерного массива

```
public class Task {  
    public static void main(String[] args) {  
        int [][]array = {{1,2,3}, {4,5,6}};  
        for(int[] i : array) {// идем по строкам.  
            for(int j : i) { // идем по столбцам.  
                System.out.print(j+"\t");  
            }  
            System.out.println();  
        }  
    }  
}
```



# Вывод двумерного массива

```
import java.util.Arrays;  
  
public class Task {  
    public static void main(String[] args) {  
        int [][]array = {{1,2,3}, {4,5,6}};  
        System.out.println(Arrays.deepToString(array));  
    }  
}
```





*Дан двумерный массив, содержащий отрицательные и положительные числа. Выведите на экран номера тех ячеек массива, которые содержат отрицательные числа.*

```
public class Task {  
    public static void main(String[] args) {  
        int [][] array = {{1,-2,-4}, {2,6,-7},{2,-3,8}};  
  
        for (int i = 0; i < array.length; i++) {  
            for (int j = 0; j < array.length; j++) {  
                if(array[i][j]<0) {  
                    System.out.println("Элемент: "+ array[i][j] +" Строка: "+i +" Столбец: "+j);  
                }  
            }  
        }  
    }  
}
```



*Дан двумерный массив, заполненный случайными числами. Вывести на экран максимальный элемент массива, а также номер его столбца и строки.*



```
public class Task {
    public static void main(String[] args) {
        int [][] array = {{1,6,7},{9,-6,19},{4,6,2}};
        int max = array[0][0];
        int maxIndexI = 0;
        int maxIndexJ = 0;

        for (int i = 0; i < array.length; i++) {
            for (int j = 0; j < array.length; j++) {
                if(max <= array[i][j]){
                    max = array[i][j];
                    maxIndexI = i;
                    maxIndexJ = j;
                }
            }
        }
        System.out.println("Максимальный элемент: " + max);
        System.out.println("Индекс строки максимального элемента: " + maxIndexI);
        System.out.println("Индекс столбца максимального элемента: " + maxIndexJ);
    }
}
```



*Дан двумерный массив, заполненный целыми случайными числами. Необходимо вычислить сумму положительных элементов, расположенных в столбцах с четными номерами.*



```
import java.util.Arrays;
import java.util.Random;

public class Task {
    public static void main(String[] args) {
        int [][] array = new int [4][4];
        Random rnd = new Random();

        int count = 0;
        for (int i = 0; i < array.length; i++) {
            for (int j = 0; j < array.length; j++) {
                array[i][j]= rnd.nextInt(31)-10;
            }
        }

        System.out.println(Arrays.deepToString(array));

        int sum = 0;
        for (int i = 0; i < array.length; i++) {
            for (int j = 1; j < array.length; j++) {
                if(j%2 ==0) {
                    sum+=array[i][j];
                }
            }
        }
        System.out.print(sum);
    }
}
```



# Типовые алгоритмы обработки двумерного массива относительно диагоналей

1	5	6
8	9	1
2	6	7

Главная диагональ

$$i=j$$

`array[i][i]`

1	5	6
8	9	1
2	6	7

Побочная диагональ

$$j=n-i-1$$

`array[i][n-i-1]`



*Дан двумерный квадратный массив, заполненный случайными числами.*

*Необходимо найти сумму нечетных отрицательных элементов, стоящих на главной и побочной диагоналях.*





```
import java.util.Arrays;
import java.util.Random;

public class Task {
    public static void main(String[] args) {
        int [][] array = new int [3][3];
        Random rnd = new Random();

        for (int i = 0; i < array.length; i++) {
            for (int j = 0; j < array.length; j++) {
                array[i][j]= rnd.nextInt(31)-10;
            }
        }

        System.out.println(Arrays.deepToString(array));

        int sum1=0, sum2=0;
        for (int i = 0; i < array.length; i++) {
            if(array[i][i]%2!=0 && array[i][i]<0)
                sum1+=array[i][i];
        }
        for (int i = 0; i < array.length; i++) {
            if(array[i][array.length-i-1]%2!=0 && array[i][array.length-i-1]<0)
                sum2+=array[i][array.length-i-1];
        }
        System.out.print(sum1+sum2);
    }
}
```





*Дан двумерный квадратный массив, заполненный случайными числами.*

*Найти минимальный элемент, лежащий ниже побочной диагонали.*



```
import java.util.Arrays;
import java.util.Random;

public class Task {
    public static void main(String[] args) {
        int [][] array = new int [3][3];
        Random rnd = new Random();

        for (int i = 0; i < array.length; i++) {
            for (int j = 0; j < array.length; j++) {
                array[i][j]= rnd.nextInt(31)-10;
            }
        }

        System.out.println(Arrays.deepToString(array));

        int min = array[1][array.length-1];
        for (int i = 0; i < array.length; i++) {
            for (int j = array.length-i; j < array.length; j++) {
                if(array[i][j]<min) {
                    min = array[i][j];
                }
            }
        }
        System.out.println(min);
    }
}
```



*Напишите программу, заполняющую двумерный массив по следующему правилу: на побочной диагонали стоят 0, выше диагонали находятся 2, ниже  $-1$ .*

2	2	2	2	0
2	2	2	0	1
2	2	0	1	1
2	0	1	1	1
0	1	1	1	1



```
public class Task {  
    public static void main(String[] args) {  
        int [][] array = new int [6][6];  
  
        for (int i = 0; i < array.length; i++) {  
            for (int j = 0; j < array.length; j++) {  
                if(i+j == array.length-1) array[i][j] = 0;  
                if(i+j < array.length-1) array[i][j] = 2;  
                if(i+j > array.length-1) array[i][j] = 1;  
            }  
        }  
  
        for (int i = 0; i < array.length; i++) {  
            for (int j = 0; j < array.length; j++) {  
                System.out.print(array[i][j] +"\t");  
            }  
            System.out.println();  
        }  
    }  
}
```



# *Домашнее задание по двумерным массивам*

*Задача 1. Вычислить сумму квадратов отрицательных элементов массива, расположенных в столбцах с четными номерами.*

*Задача 2. Найти минимальный и максимальный элементы массива и поменять их местами.*

*Задача 3. Найти среднее арифметическое массива. Вывести на экран только те элементы, которые больше найденного среднего арифметического.*

