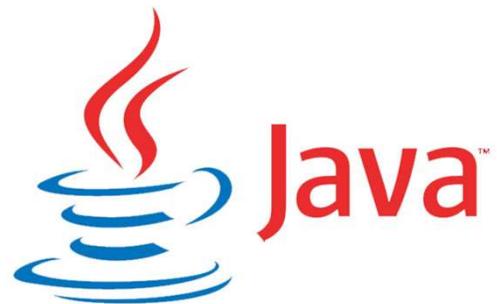


Тема 4

Многомерные массивы





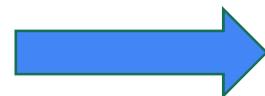
Двухмерные массивы

Ранее мы рассматривали одномерные массивы, которые можно представить как цепочку или строку однотипных значений.

Но кроме одномерных массивов также бывают и многомерные.

Наиболее известный многомерный массив - таблица, представляющая **двухмерный массив**.

0	1	2
3	4	5



Вот так выглядит двухмерный массив



Двухмерные массивы

Создание и инициализация

- Создание и инициализация двухмерного массива выглядит следующим образом:

int[][] nums2 = { { 0, 1, 2 }, { 3, 4, 5 } };

Или

int[][] nums2 = new int[2][3]; (без инициализации)

- А вот так может выглядеть объявление трехмерного массива:

int[][][] nums3 = new int[2][3][4];



Двухмерные массивы

Заполнение значениями

- Как и в одномерных массивах, для выполнения операции над массивом целиком приходится пользоваться циклами;
- При работе с многомерными массивами используется вложенное итерирование, т.е. несколько циклов, вложенных в друг друга:

```
for (int i = 0; i < m.length; i++)
{
    for (int j = 0; j < m[i].length; j++)
    {
        m[i][j]=1;
    }
}
```

Пример заполнения двухмерного массива единицами.



Двухмерные массивы

Заполнение значениями

Length вызывается первый раз от самого массива, а во втором – от его строки.

В первом случае оно возвращает количество строк в массиве, а во втором – количество элементов в строке.



Двухмерные массивы

Неправильные массивы

- В Java разные измерения одного и того же массива могут иметь **разные размеры**;
- Такие массивы называются **неправильными** или **зубчатыми**.

Пример:

```
int[][] nums = new int[3][];
nums[0] = new int[2];
nums[1] = new int[3];
nums[2] = new int[5];
```

Трёхмерные массивы

Трёхмерный массив можно представить себе как набор матриц, каждую из которых мы записали на библиотечной карточке.

Тогда чтобы добраться до конкретного числа сначала нужно указать номер карточки (первый индекс трёхмерного массива), потому указать номер строки (второй индекс массива) и только затем номер элемент в строке (третий индекс).

n -мерные массивы

Соответственно, для того, чтобы обратиться к элементу n -мерного массива нужно указать n индексов.

Создание 2-мерного массива с клавиатуры

```
System.out.println("Введите размерность массива NxM");
Scanner sc=new Scanner(System.in); //создаём объект типа Scanner
Random r= new Random(); // создаём объект типа Random
int n = 0,m = 0; //объявление переменных
if(sc.hasNextInt()){
    n=sc.nextInt(); // вводим n с клавиатуры
    m=sc.nextInt(); // вводим m с клавиатуры
}
int mass [][]=new int[n][m]; //инициализация массива размером NxM
```

Заполнение 2-мерного массива случайными числами

```
for(int i=0; i<n;i++){
    for(int j=0; j<m; j++){
        mass[i][j]=r.nextInt(9);
    }
}
```

Вывод 2-мерного массива на экран

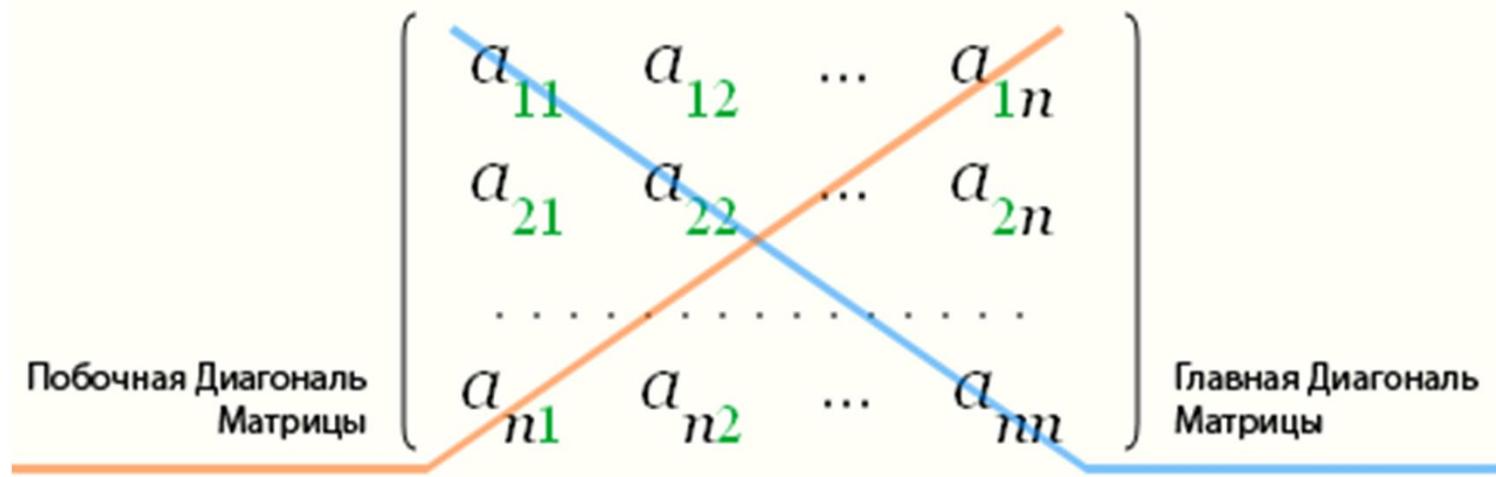
```
for(int i=0; i<n;i++){// вывод массива в консоль
    for(int j=0; j<m; j++){
        System.out.print(mass[i][j]+" ");
    }
    System.out.println();
}
```

Практика

Задание: найдите максимальное значение в 2-мерном массиве.

Практика

Задание: найдите сумму элементов на главной диагонали.



Практика

Задание: найдите сумму элементов на побочной диагонали.

