



# **Багатовимірні масиви**

---

Багатовимірні масиви по суті – це масив масивів. Робота з багатовимірними масивами подібна до роботи з одновимірними. Відмінність лише в тому, що використовуються додаткові квадратні дужки. Переважно використовуються двовимірні масиви, які служать для роботи з табличними даними та трьохвимірні масиви. Двовимірний масив та трьохвимірний, можна оголосити наступним чином:

```
int twoD[][] = new int [4][5]; //створення масиву 4x5  
int threeD[][][] = new int[5][5][5]; //створення масиву 5x5x5
```


Для двовимірного лівий індекс означає номер рядка, а правий номер стовпця. Це можна уявити наступним чином:

[0,0][0,1][0,2][0,3][0,4]

[1,0][1,1][1,2][1,3][1,4]

[2,0][2,1][2,2][2,3][2,4]

[3,0][3,1][3,2][3,3][3,4]



Трьохвимірний масив можна уявити у вигляді куба. Крім номера рядка і номера стовпця, додається ще індекс елемента вглибину.

Наступна програма створює масив 5 на 4, заповнює його випадковими числами і виводить на екран.

```
import java.util.Random; // імпортуємо клас Random
public class RandomArray {
    public static void main(String[] args)
    {
        int m = 5, n = 4; //оголошуємо і ініцілізуємо змінні з розмірами масиву
        int Array[][] = new int[m][n]; //оголошуємо і ініціалізуємо масив
        Random generator = new Random(); // створюємо генератор випадкових чисел

        int gn; //змінна в яку буде записуватися згенероване генератором число
        /* заповнюємо масив випадковими числами */
        for (int i = 0; i < m; i++) //проходимося по стовпцях
            for (int j = 0; j < n; j++) { //проходимося по рядках
                gn = generator.nextInt(100); //генерація випадкового числа від 0 до 100;
                Array[i][j] = gn; //записуємо згенероване випадкове число }
            }
        /* Виводимо результат */
        for (int i = 0; i < m; i++) { for (int j = 0; j < n; j++) // зверніть увагу на відсутність фігурної дужки
            System.out.print(Array[i][j] + " "); //даний рядок відноситься до масиву по j
            System.out.println(); //виводимо символи переводу каретки і нового рядка //після кожного проходження стовпцевих елементів
        }
    }
}
```