

# Chương 5: Mảng (tt)

## Mảng nhiều chiều



# Khai báo / tạo mảng 2 chiều

// Khai báo mảng

```
dataType[][] refVar;
```

// Tạo mảng

```
refVar = new dataType[10][10];
```

// Kết hợp khai báo và tạo mảng

```
dataType[][] refVar = new dataType[10][10];
```

// Hoặc

```
dataType refVar[][] = new dataType[10][10];
```



# Ví dụ

```
int[][] matrix = new int[10][10];
```

```
// hoặc
```

```
int matrix[][] = new int[10][10];
```

```
matrix[0][0] = 3;
```

```
for (int i = 0; i < matrix.length; i++)  
    for (int j = 0; j < matrix[i].length; j++)  
        matrix[i][j] = (int) (Math.random() * 1000);
```

```
double[][] x;
```

# Minh họa mảng 2 chiều

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

```
matrix = new int[5][5];
```

(a)

matrix.length? 5

matrix[0].length? 5

	[0]	[1]	[2]	[3]	[4]
[0]	0	0	0	0	0
[1]	0	0	0	0	0
[2]	0	7	0	0	0
[3]	0	0	0	0	0
[4]	0	0	0	0	0

```
matrix[2][1] = 7;
```

(b)

	[0]	[1]	[2]
[0]	1	2	3
[1]	4	5	6
[2]	7	8	9
[3]	10	11	12

```
int[][] array = {  
    {1, 2, 3},  
    {4, 5, 6},  
    {7, 8, 9},  
    {10, 11, 12}  
};
```

(c)

array.length? 4

array[0].length? 3

# Kết hợp khai báo – tạo – khởi tạo giá trị

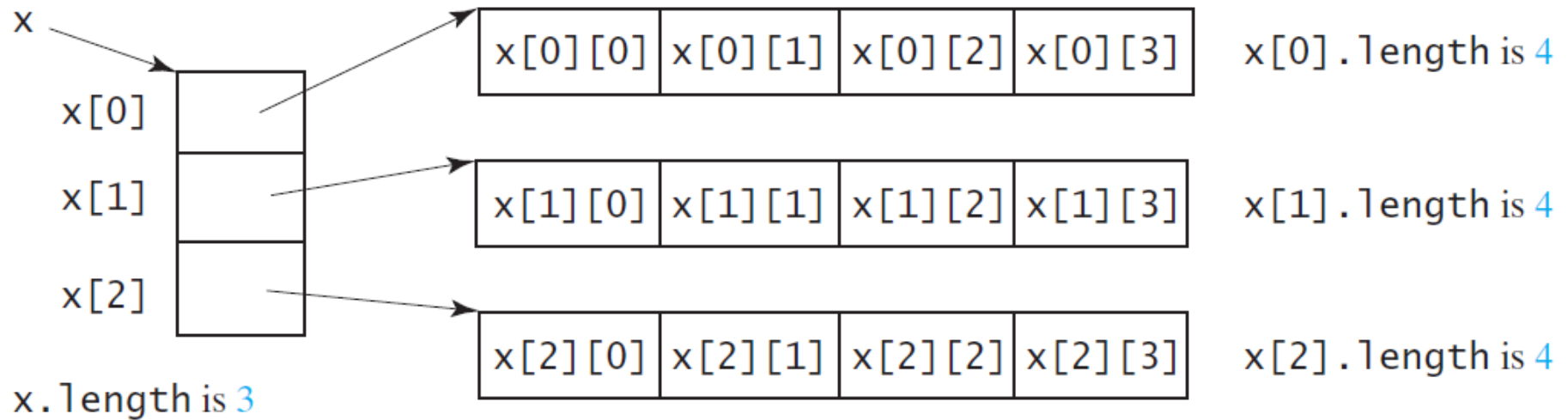
```
int[][] array = {  
    {1, 2, 3},  
    {4, 5, 6},  
    {7, 8, 9},  
    {10, 11, 12}  
};
```

Tương đương

```
int[][] array = new int[4][3];  
array[0][0] = 1; array[0][1] = 2; array[0][2] = 3;  
array[1][0] = 4; array[1][1] = 5; array[1][2] = 6;  
array[2][0] = 7; array[2][1] = 8; array[2][2] = 9;  
array[3][0] = 10; array[3][1] = 11; array[3][2] = 12;
```

# Kích thước của mảng 2 chiều

```
int[][] x = new int[3][4];
```



# Kích thước của mảng 2 chiều

```
int[][] array = {  
    {1, 2, 3},  
    {4, 5, 6},  
    {7, 8, 9},  
    {10, 11, 12}  
};
```

array.length

array[0].length

array[1].length

array[2].length

array[3].length

array[4].length

**ArrayIndexOutOfBoundsException**



# Mảng ghô ghê

Mỗi dòng trong một mảng 2 chiều là *một mảng 1 chiều*.  
Các dòng có thể có kích thước khác nhau, đó là một *mảng ghô ghê*.

Ví dụ:

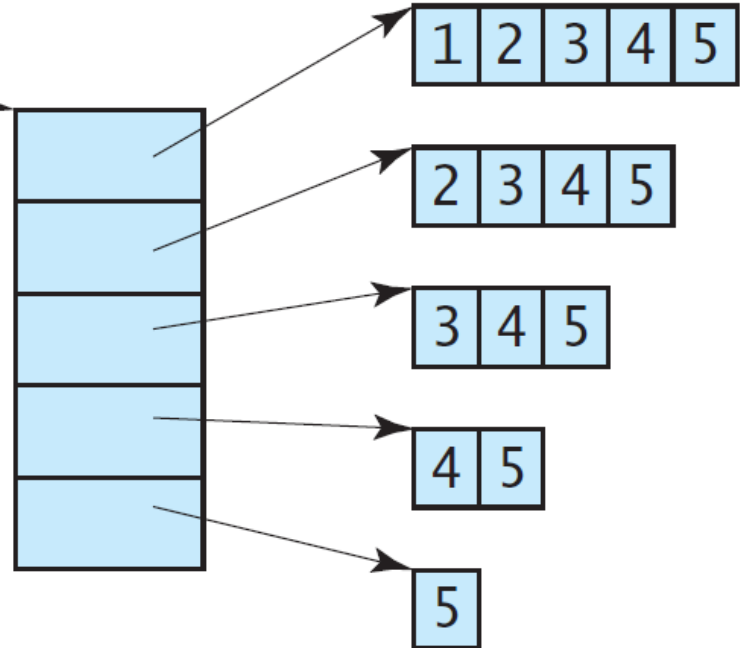
```
int[][] matrix = {  
    {1, 2, 3, 4, 5},  
    {2, 3, 4, 5},  
    {3, 4, 5},  
    {4, 5},  
    {5}  
};
```

```
matrix.length is 5  
matrix[0].length is 5  
matrix[1].length is 4  
matrix[2].length is 3  
matrix[3].length is 2  
matrix[4].length is 1
```



# Mảng ghồ ghề

```
int[][] triangleArray = {  
    {1, 2, 3, 4, 5},  
    {2, 3, 4, 5},  
    {3, 4, 5},  
    {4, 5},  
    {5}  
};
```



# Một số thao tác trên mảng 2 chiều

1. Khởi tạo mảng với các giá trị được nhập vào / ngẫu nhiên
2. In mảng
3. Tính tổng các phần tử
4. Tính tổng các phần tử theo từng cột
5. Tìm dòng có giá trị lớn nhất
6. Tìm vị trí nhỏ nhất của phần tử lớn nhất
7. *Trộn ngẫu nhiên*



# Khởi tạo mảng với các giá trị được nhập vào

```
java.util.Scanner input = new Scanner(System.in);
System.out.println("Enter " + matrix.length + " rows and " +
    matrix[0].length + " columns: ");
for (int row = 0; row < matrix.length; row++) {
    for (int column = 0; column < matrix[row].length; column++) {
        matrix[row][column] = input.nextInt();
    }
}
```



# Khởi tạo mảng với các giá trị ngẫu nhiên

```
for (int row = 0; row < matrix.length; row++) {  
    for (int column = 0; column < matrix[row].length; column++) {  
        matrix[row][column] = (int)(Math.random() * 100);  
    }  
}
```



# In mảng

```
for (int row = 0; row < matrix.length; row++) {  
    for (int column = 0; column < matrix[row].length; column++) {  
        System.out.print(matrix[row][column] + " ");  
    }  
  
    System.out.println();  
}
```



# Tính tổng các phần tử

```
int total = 0;
for (int row = 0; row < matrix.length; row++) {
    for (int column = 0; column < matrix[row].length; column++) {
        total += matrix[row][column];
    }
}
```



# Tính tổng các phần tử theo từng cột

```
for (int column = 0; column < matrix[0].length; column++) {  
    int total = 0;  
    for (int row = 0; row < matrix.length; row++)  
        total += matrix[row][column];  
    System.out.println("Sum for column " + column + " is "  
        + total);  
}
```



# Trộn ngẫu nhiên

```
for (int i = 0; i < matrix.length; i++) {  
    for (int j = 0; j < matrix[i].length; j++) {  
        int i1 = (int)(Math.random() * matrix.length);  
        int j1 = (int)(Math.random() * matrix[i].length);  
        // Swap matrix[i][j] with matrix[i1][j1]  
        int temp = matrix[i][j];  
        matrix[i][j] = matrix[i1][j1];  
        matrix[i1][j1] = temp;  
    }  
}
```





# Mảng nhiều chiều

Trong Java, chúng ta có thể tạo mảng *n-chiều*, với **n** là số nguyên dương bất kỳ.

Tương tự như cách tạo mảng 2 chiều, với  $n \geq 3$



# Mảng nhiều chiều

```
double[][][] scores = {  
    {{7.5, 20.5}, {9.0, 22.5}, {15, 33.5}, {13, 21.5}, {15, 2.5}},  
    {{4.5, 21.5}, {9.0, 22.5}, {15, 34.5}, {12, 20.5}, {14, 9.5}},  
    {{6.5, 30.5}, {9.4, 10.5}, {11, 33.5}, {11, 23.5}, {10, 2.5}},  
    {{6.5, 23.5}, {9.4, 32.5}, {13, 34.5}, {11, 20.5}, {16, 7.5}},  
    {{8.5, 26.5}, {9.4, 52.5}, {13, 36.5}, {13, 24.5}, {16, 2.5}},  
    {{9.5, 20.5}, {9.4, 42.5}, {13, 31.5}, {12, 20.5}, {16, 6.5}}};
```

Which student

Which exam

Multiple-choice or essay

