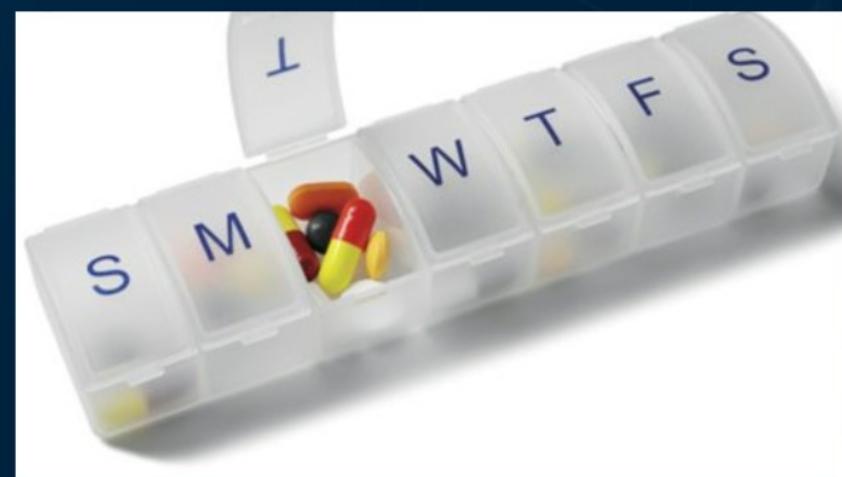


ARRAY

WHAT IS AN ARRAY?

0. Data structure / Collection of elements
1. Elements of same data type
2. Fixed number of size



First index - Array indexes starts from 0



0

1

2

3

4

5

Civil	CSE	ECE	EEE	IT	Mech
-------	-----	-----	-----	----	------

String

ARRAY DECLARATION

0

1

2

3

4

5

Civil

CSE

ECE

EEE

IT

Mech

```
String[] departments;
```

```
String departments[];
```

ARRAY INITIALIZATION

0	1	2	3	4	5
Civil	CSE	ECE	EEE	IT	Mech

```
String departments[] = {"Civil", "CSE", "ECE", "EEE", "IT", "Mech" };
```

```
String departments[] = new String[6];
```

```
departments[0] = "Civil" ;
```

```
departments[1] = "CSE";
```

```
...
```

ACCESS THE ELEMENTS OF ARRAY

```
String departments[] = {"Civil", "CSE", "ECE", "EEE", "IT", "Mech" };  
for(int i = 0; i < departments.length; i++) {  
    System.out.println(departments[i]);  
}
```

Output :

Civil
CSE
ECE
EEE
IT
Mech

CHANGE THE ARRAY ELEMENT

```
String departments[] = {"Civil", "CSE", "ECE", "EEE", "IT", "MECH" };  
departments[0] = "Aeronautical";  
  
for(int i = 0; i < departments.length; i++) {  
    System.out.println(departments[i]);  
}
```

Output :

Aeronautical
CSE
ECE
EEE
IT
Mech

LOOP THROUGH FOR-EACH

```
String departments[] = {"Civil", "CSE", "ECE", "EEE", "IT", "MECH" } ;  
for(String i : departments) {  
    System.out.println(i);  
}
```

Output :

Civil
CSE
ECE
EEE
IT
Mech

CAN PRINT AN ARRAY WITHOUT LOOP ?

```
String departments[] = {"Civil", "CSE", "ECE", "EEE", "IT", "MECH"};
```

```
Arrays.toString(departments)
```

First index - Array indexes starts from 0



0

1

2

3

4

5

CIVIL

CSE

ECE

EEE

IT

MECH

String

WHY DO ARRAY INDEXES START WITH 0?

- 0. Influenced by C Language
- 1. Address of first element

0	1	2	3	4	5
CIVIL	CSE	ECE	EEE	IT	MECH

address 0

address 1

address 2

address 3

address 4

address 5

String departments[6];

departments = &departments



Address of departments array

*departments = *(departments + 0)

*departments = "CIVIL"



Value of department[0]

*departments[1] = *(departments + 1)



Value of department[1]

*departments[n] = *(departments + n)



Value of department[n]

0

1

2

3

4

5

CIVIL

CSE

ECE

EEE

IT

MECH

String department[6];

Index 0 :

value of first element of array *(department) aka *(department + 0)

value of 2nd element of array *(department + 1)

department[0] = *(department + 0)

Index 1 :

department[1] = *(department + 1 - 1)

MULTI DIMENSIONAL ARRAY

MULTI DIMENSIONAL ARRAY

0. Arrays of array

1. Appends a [] per dimension

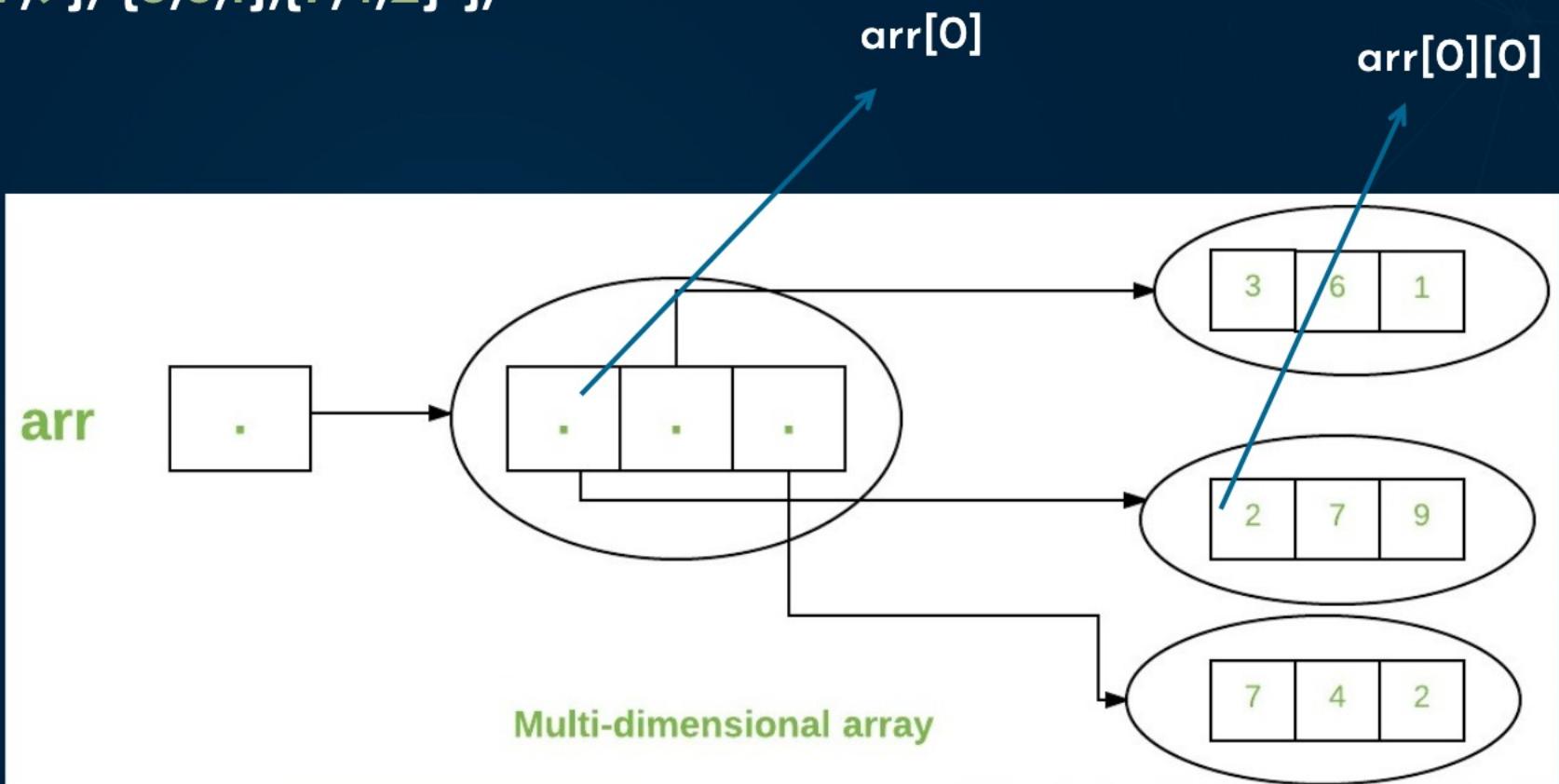
```
int intArray[][] = new int[2][4]; → a 2D array
```

```
int intArray[][][] = new int[2][4][3]; → a 3D array
```

INITIALIZATION OF A 2D ARRAY

```
int arr[][] = { { 2,7,9}, {3,6,1},{7,4,2} };
```

2	7	9
3	6	1
7	4	2



PRINTING 2D ARRAY

```
int arr[][] = { { 2,7,9},{3,6,1},{7,4,2} };
```

```
for (int i=0; i<3 ; i++)
{
    for (int j=0; j <3 ; j++) {
        System.out.print(arr[i][j] + " ");
    }
    System.out.println();
}
```

Output:

2	7	9
3	6	1
7	4	2

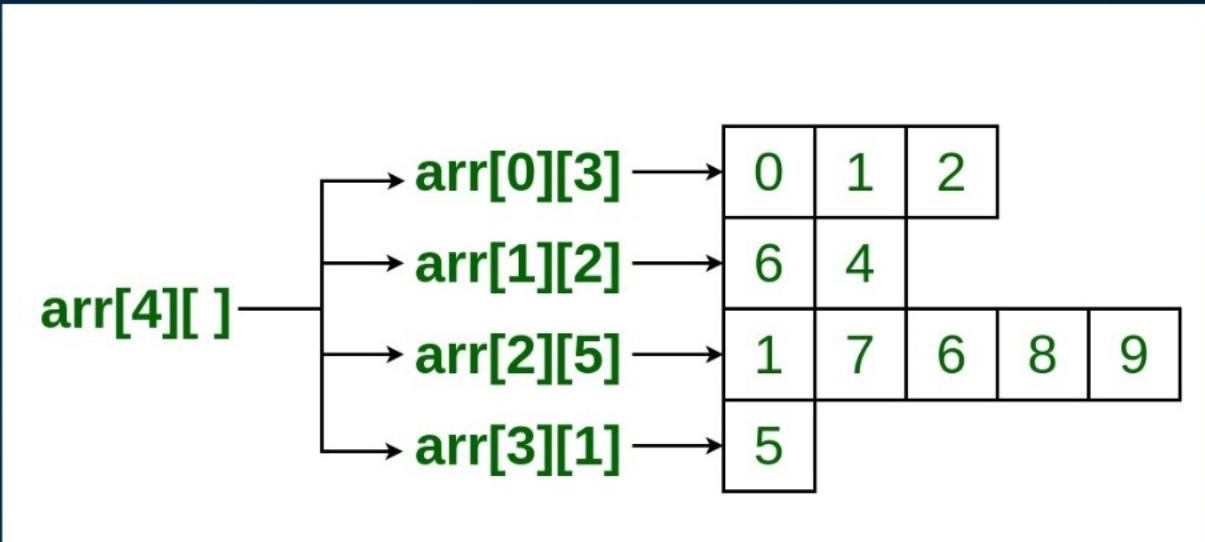
JAGGED ARRAY



JAGGED ARRAY

- Member array can be different sizes

```
int jaggedArray[][] = new int[4][];  
jaggedArray[0] = {0,1,3};  
jaggedArray[1] = {6,4};  
jaggedArray[2] = {1,7,6,8,9};  
jaggedArray[3] = {5};
```



ANONYMOUS ARRAY



ANONYMOUS ARRAY

O. Array without name

1. We can pass an array with user values without the referenced variable.

```
new int[] {0,1,3}; //Anonymous integer array
```

```
new char[] {'a','b','c'}; //Anonymous char array
```

```
new String[] {"ECE", "CSE", "IT"}; //Anonymous String array
```

```
new int[][] {{0,1,3}, {2,3}}; //Anonymous multi dimensional array
```

```
class Test {
    public static void main(String[] args)
    {
        // anonymous array
        sum(new int[]{ 1, 2, 3 });
    }

    public static void sum(int[] a)
    {
        int total = 0;

        // using for-each loop
        for (int i : a)
            total = total + i;

        System.out.println("The sum is: " + total);
    }
}
```

The sum is : 6

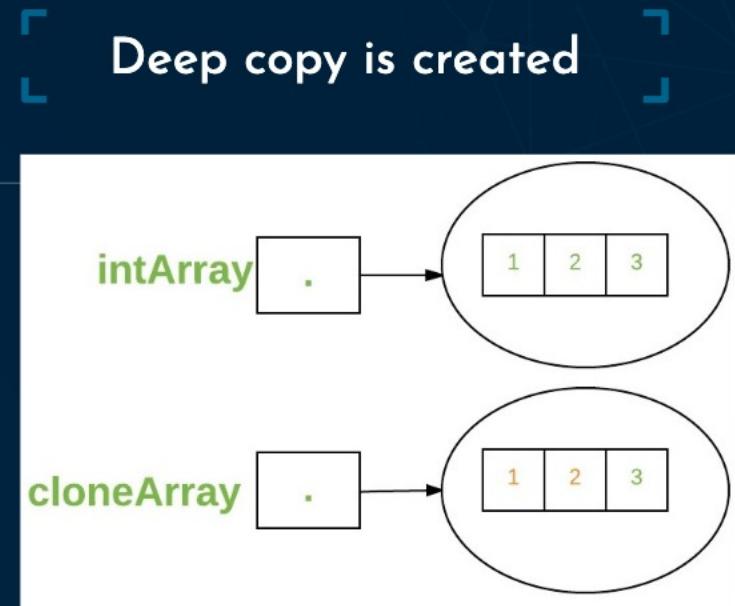
CLONING OF AN ARRAY

```
class Test
{
    public static void main(String args[])
    {
        int intArray[] = {1,2,3};

        int cloneArray[] = intArray.clone();

        System.out.println(intArray == cloneArray);

        for (int i = 0; i < cloneArray.length; i++) {
            System.out.print(cloneArray[i]+ " ");
        }
    }
}
```



Output :

False

1 2 3

CLONING OF A 2D ARRAY

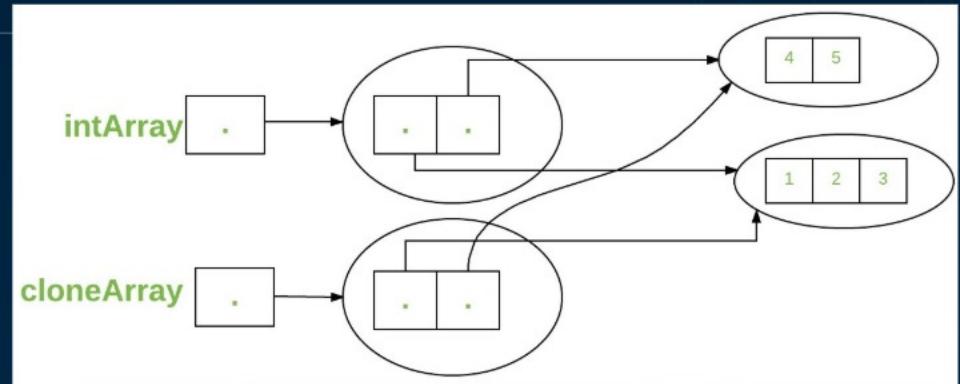
```
class Test
{
    public static void main(String args[])
    {
        int intArray[][] = {{1,2,3},{4,5}};

        int cloneArray[][] = intArray.clone();

        System.out.println(intArray == cloneArray);

        System.out.println(intArray[0] == cloneArray[0]);
        System.out.println(intArray[1] == cloneArray[1]);
    }
}
```

Shallow copy is created



Output :



What will be the output?

```
class TestReturnArray{  
  
    static int[] get(){  
        return new int[]{10,30,50,90,60};  
    }  
  
    public static void main(String args[]){  
  
        int arr[] = get();  
  
        for(int i=0;i<arr.length;i++){  
  
            System.out.println(arr[i]);  
        }  
    }  
}
```

10
30
50
90
60

What will be the output?

```
class GFG
{
    public static void main (String[] args)
    {
        int[] arr = new int[2];
        arr[0] = 10;
        arr[1] = 20;

        for (int i = 0; i <= arr.length; i++)
            System.out.println(arr[i]);
    }
}
```

Runtime error:

10

20

Exception in thread "main"
java.lang.ArrayIndexOutOfBoundsException:
at GFG.main(File.java:12)

WHAT IS ARRAYS CLASS?

0. The Arrays class in `java.util` package is a part of the Java Collection Framework.
1. It consists of only static methods and the methods of Object class to access Java Arrays.

ARRAY ROTATION :

Rotate by +90:

1. Transpose
2. Reverse each row

Rotate by -90:

Method 1 :

1. Transpose
2. Reverse each column

Method 2 :

1. Reverse each row
2. Transpose

Rotate by +180:

Method 1: Rotate by +90 twice

Method 2: Reverse each row and then reverse each column (Transpose)

Rotate by -180:

Method 1: Rotate by -90 twice

Method 2: Reverse each column and then reverse each row

Method 3: Rotate by +180 as they are same

JAVA GAMING PROJECTS WITH ARRAY

1. Tic Tac Toe Java Game
2. Snake and ladder Game
3. Mine Sweeper Game
4. Chess
5. Sudoku

& Many more

REFERENCE

- <https://www.geeksforgeeks.org/top-50-array-coding-problems-for-interviews/>
- <https://www.geeksforgeeks.org/category/data-structures/matrix/>
- <https://algorithms.tutorialhorizon.com/top-40-problems-on-arrays/>
- <https://leetcode.com/tag/array/>
- <https://www.hackerrank.com/domains/data-structures?filters%5Bsubdomains%5D%5B%5D=arrays>