

# Array

---

Ratna Mufidah, S.Kom., M.Kom.

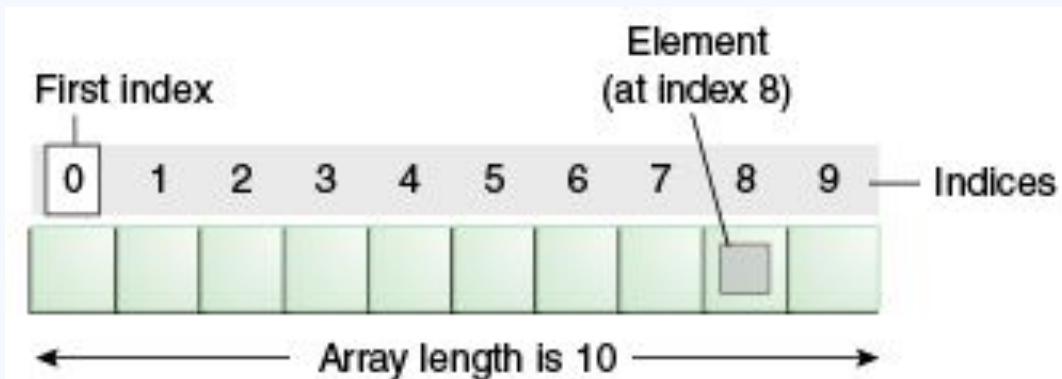
# Array

An array is a **collection of similar type of elements** which has **contiguous memory location**

Arrays are used to **store multiple values in a single variable.**

# Array in Java

**Array in Java** is **index-based**, the first element of the array is stored at the 0th index, 2nd element is stored on 1st index and so on.





# Array in Java

- Fixed length
  - Fast access
  - An array can hold primitives or objects
  - An array of primitives stores values of the primitives
  - An array of objects stores only the references to the objects
  - An array itself is actually an object
  - Arrays elements have default values
- 

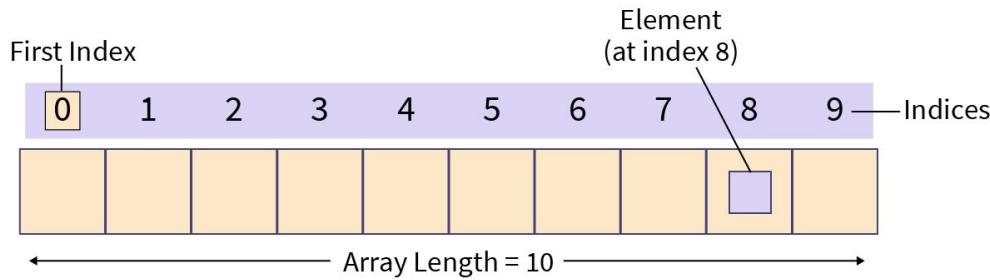
# Array types in Java

Types of Array in Java

Single Dimensional Array

Multi-Dimensional Array

# Single Dimensional Array in Java



SCALER  
*Topics*

# Single Dimensional Array in Java

## Syntax

```
dataType[] arr; (or)  
dataType []arr; (or)  
dataType arr[];
```

## Instantiation of array

```
arrayRefVar=new datatype[size];
```

# Single Dimensional Array in Java

`Array1DTest.java`

```
1 public class Array1DTest {  
2  
3     public static void main(String[] args) {  
4         // TODO Auto-generated method stub  
5         int a[]={};//declaration and instantiation  
6         a[0]=10;//initialization  
7         a[1]=20;  
8         a[2]=70;  
9         a[3]=30;  
10        a[4]=50;  
11  
12        //traversing array  
13        //length is the property of array  
14        for(int i=0;i<a.length;i++)  
15            System.out.print(a[i]+ " ");  
16    }  
17 }
```

# Single Dimensional Array in Java

Array1DTest1.java

```
1 public class Array1DTest1 {  
2  
3     public static void main(String[] args) {  
4         // TODO Auto-generated method stub  
5         //declaration, instantiation and initialization  
6         int a[]={10,20,70,30,50};  
7         //printing array  
8         //length is the property of array  
9         for(int i=0;i<a.length;i++)  
10            System.out.print(a[i]+ " ");  
11  
12    }  
13 }
```

# **Single Dimensional Array in Java**

## **\_For-each loop\_**

The Java for-each loop prints the array elements one by one.

### **Syntax**

```
for(data_type variable:array) {  
    //body of the loop  
}
```

# Single Dimensional Array in Java

## \_For-each loop\_

**Array1DFor.java**

```
1 public class Array1DFor {  
2  
3     public static void main(String[] args) {  
4         // TODO Auto-generated method stub  
5         //declaration, instantiation and initialization  
6         int a[]={10,20,70,30,50};  
7  
8         //printing array using for-each loop  
9         for(int i:a)  
10             System.out.print(i+" ");  
11     }  
12  
13 }
```

# Single Dimensional Array in Java

## PassingArray1D.java

```
1 public class PassingArray1D {  
2     //creating a method which receives an array as a parameter  
3     static void min(int arr[]){  
4         int min=arr[0];  
5         for(int i=1;i<arr.length;i++)  
6             if(min>arr[i])  
7                 min=arr[i];  
8  
9         System.out.println(min);  
10    }  
11  
12    public static void main(String[] args) {  
13        // TODO Auto-generated method stub  
14        //declaration, instantiation and initialization  
15        int a[]={10,20,70,30,50};  
16  
17        //passing array to method  
18        min(a);  
19    }  
20  
21 }
```

# Single Dimensional Array in Java

AnonymousArray1D.java

```
1 public class AnonymousArray1D {  
2     //creating a method which receives an array as a parameter  
3     static void max(int arr[]){  
4         int max=arr[0];  
5         for(int i=1;i<arr.length;i++)  
6             if(max<arr[i])  
7                 max=arr[i];  
8  
9         System.out.println(max);  
10    }  
11  
12    public static void main(String[] args) {  
13        // TODO Auto-generated method stub  
14  
15        //passing anonymous array to method  
16        max(new int[]{10,20,70,30,50});  
17    }  
18}
```

# Single Dimensional Array in Java

ReturnArray1D.java

```
1 public class ReturnArray1D {  
2  
3     //creating method which returns an array  
4     static int[] get(){  
5         return new int[]{10,20,70,30,50};  
6     }  
7  
8     public static void main(String args[]){  
9         //calling method which returns an array  
10        int arr[]=get();  
11        //printing the values of an array  
12        for(int i:arr)  
13            System.out.print(i+" ");  
14    }  
15 }
```

# Single Dimensional Array in Java

TestArray.java

```
1 public class TestArray {  
2  
3     public static void main(String[] args) {  
4         // TODO Auto-generated method stub  
5         //declaration, instantiation and initialization  
6         int a[]={10,20,70,30,50};  
7  
8         //printing array  
9         for(int i=0;i<=a.length;i++)  
10            System.out.print(a[i]+" ");  
11    }  
12  
13 }
```

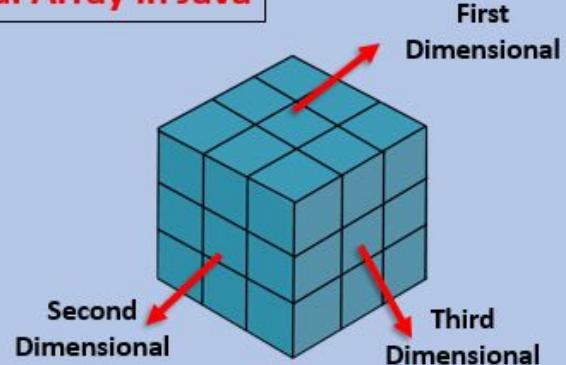
# Multidimensional Array in Java

## Multidimensional Array in Java

### Types of Multidimensional Array in Java

	Column 0	Column 1	Column 2
Row 0	X[0][0]	X[0][1]	X[0][2]
Row 1	X[1][0]	X[1][1]	X[1][2]
Row 2	X[2][0]	X[2][1]	X[2][2]

**2D-Array**



**3D-Array**

# Multidimensional Array in Java

## 2D Array Syntax

```
//declaration syntax  
  
data_type[][] array_name = new data_type[x][y];  
  
//initialization syntax  
  
array_name[row_index][column_index] = value;
```

## 3D Array Syntax

```
//declaration syntax  
  
data_type[][][] array_name = new data_type[x][y][z];  
  
//initialization syntax  
  
array_name[array_index][row_index][column_index] = value;
```

# Multidimensional Array in Java

## \_2D Array\_

	Column 0	Column 1	Column 2
Row 0	x[0][0]	x[0][1]	x[0][2]
Row 1	x[1][0]	x[1][1]	x[1][2]
Row 2	x[2][0]	x[2][1]	x[2][2]

<https://www.geeksforgeeks.org/multidimensional-arrays-in-java/>

# Multidimensional Array in Java

## \_2D Array\_

Array2DTest1.java

```
1 public class Array2DTest1 {  
2  
3     public static void main(String[] args) {  
4         // TODO Auto-generated method stub  
5         //declaring and initializing 2D array  
6         int arr[][]={{1,2,3},{4,5,6},{7,8,9}};  
7  
8         //printing 2D array  
9         for(int i=0;i<3;i++){  
10             for(int j=0;j<3;j++){  
11                 System.out.print(arr[i][j]+" ");  
12             }  
13             System.out.println();  
14         }  
15     }  
16 }
```

# Multidimensional Array in Java

## \_2D Array\_

### JaggedArrayTest.java

```
1 public class JaggedArrayTest {  
2  
3     public static void main(String[] args) {  
4         // TODO Auto-generated method stub  
5         //declaring a 2D array with odd columns  
6         int arr[][] = new int[4][];  
7         arr[0] = new int[3];  
8         arr[1] = new int[5];  
9         arr[2] = new int[2];  
10        arr[3] = new int[4];  
11        //initializing a jagged array  
12        int count = 1;  
13        for (int i=0; i<arr.length; i++)  
14            for(int j=0; j<arr[i].length; j++)  
15                arr[i][j] = count++;  
16  
17        //printing the data of a jagged array  
18        for (int i=0; i<arr.length; i++){  
19            for (int j=0; j<arr[i].length; j++){  
20                System.out.print(arr[i][j]+" ");  
21            }  
22            System.out.println(); //new line  
23        }  
24    }  
25 }
```

# Multidimensional Array in Java

## \_2D Array\_

`CopyArrayTest.java`

```
1 public class CopyArrayTest {  
2  
3     public static void main(String[] args) {  
4         // TODO Auto-generated method stub  
5         //declaring a source array  
6         char[] copyFrom = { 'd', 'e', 'c', 'a', 'f', 'f', 'e',  
7             'i', 'n', 'a', 't', 'e', 'd' };  
8         //declaring a destination array  
9         char[] copyTo = new char[7];  
10        //copying array using System.arraycopy() method  
11        System.arraycopy(copyFrom, 2, copyTo, 0, 7);  
12        //printing the destination array  
13        System.out.println(String.valueOf(copyTo));  
14    }  
15}  
16 }
```

# Multidimensional Array in Java

## \_2D Array\_

AddArrayTest.java

```
1 public class AddArrayTest {  
2  
3     public static void main(String[] args) {  
4         // TODO Auto-generated method stub  
5         //creating two matrices  
6         int a[][]={{1,5},{4,2}};  
7         int b[][]={{3,5},{2,6}};  
8  
9         //creating another matrix to store the sum of two matrices  
10        int c[][]=new int[2][2];  
11  
12        //adding and printing addition of 2 matrices  
13        for(int i=0;i<2;i++){  
14            for(int j=0;j<2;j++){  
15                c[i][j]=a[i][j]+b[i][j];  
16                System.out.print(c[i][j]+" ");  
17            }  
18            System.out.println(); //new line  
19        }  
20    }  
21 }
```

$$X = \begin{bmatrix} 1 & 5 \\ 4 & 2 \end{bmatrix} \quad Y = \begin{bmatrix} 3 & 5 \\ 2 & 6 \end{bmatrix}$$
$$X - Y = \begin{bmatrix} 1+3 & 5+5 \\ 4+2 & 2+6 \end{bmatrix}$$
$$Z = \begin{bmatrix} 4 & 10 \\ 6 & 8 \end{bmatrix}$$

# Multidimensional Array in Java

## \_2D Array\_

Multiplication of 2 Matrices in Java ???

$$AB = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$$

$$= \begin{bmatrix} 1(5) + 2(7) & 1(6) + 2(8) \\ 3(5) + 4(7) & 3(6) + 4(8) \end{bmatrix}$$

$$= \begin{bmatrix} 19 & 22 \\ 43 & 50 \end{bmatrix}$$

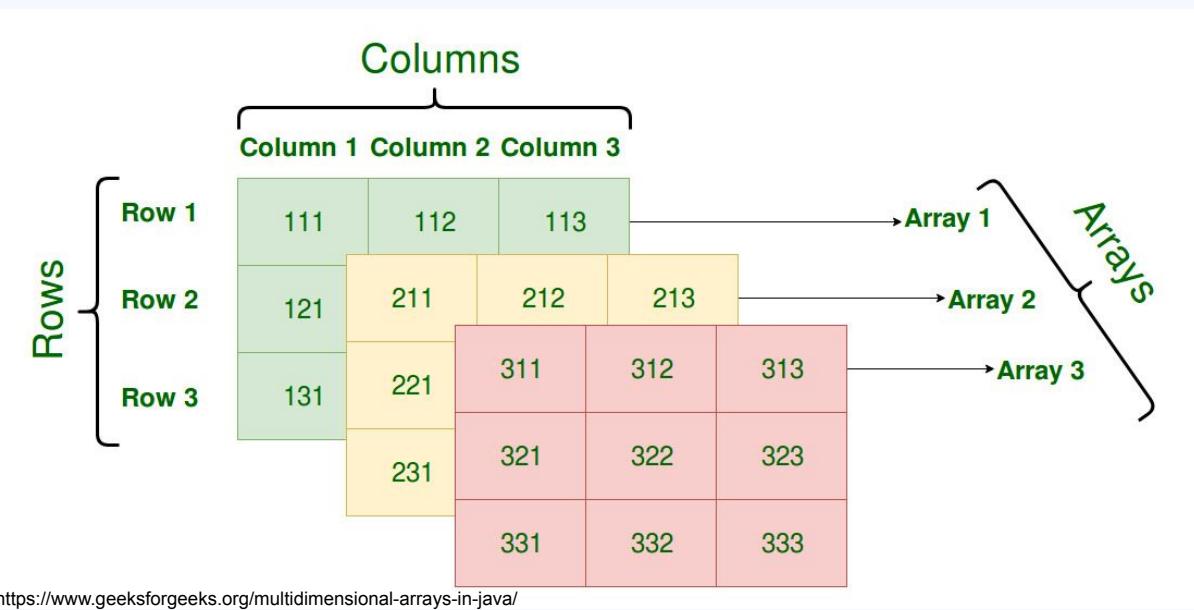
$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ -1 & 2 \end{bmatrix} =$$

$$\begin{bmatrix} 1.1 + 2.3 + 3.-1 & 1.2 + 2.1 + 3.2 \\ 4.1 + 0.3 + 1.-1 & 4.2 + 0.1 + 1.2 \end{bmatrix} =$$

$$\begin{bmatrix} 4 & 10 \\ 3 & 10 \end{bmatrix}$$

# Multidimensional Array in Java

## \_3D Array\_



# Multidimensional Array in Java

## \_2D Array\_

`Array3DTest.java`

```
1 public class Array3DTest {  
2  
3     public static void main(String[] args) {  
4         // TODO Auto-generated method stub  
5         int[][][] arr = { { { 1, 2 }, { 3, 4 } },  
6                            { { 5, 6 }, { 7, 8 } } };  
7  
8         for (int i = 0; i < 2; i++) {  
9             for (int j = 0; j < 2; j++) {  
10                 for (int k = 0; k < 2; k++) {  
11                     System.out.print(arr[i][j][k] + " ");  
12                 }  
13                 System.out.println();  
14             }  
15             System.out.println();  
16         }  
17     }  
18  
19 }
```

# Thanks !

Ada pertanyaan?

**CREDITS:** This presentation template was created by [Slidesgo](#), and includes icons by [Flaticon](#), and infographics & images by [Freepik](#)

# References

<https://www.geeksforgeeks.org/>

<https://www.theknowledgeacademy.com/>

<https://www.w3schools.com/>

<https://www.javatpoint.com/>

<https://www.codejava.net/>

