**Arrays**

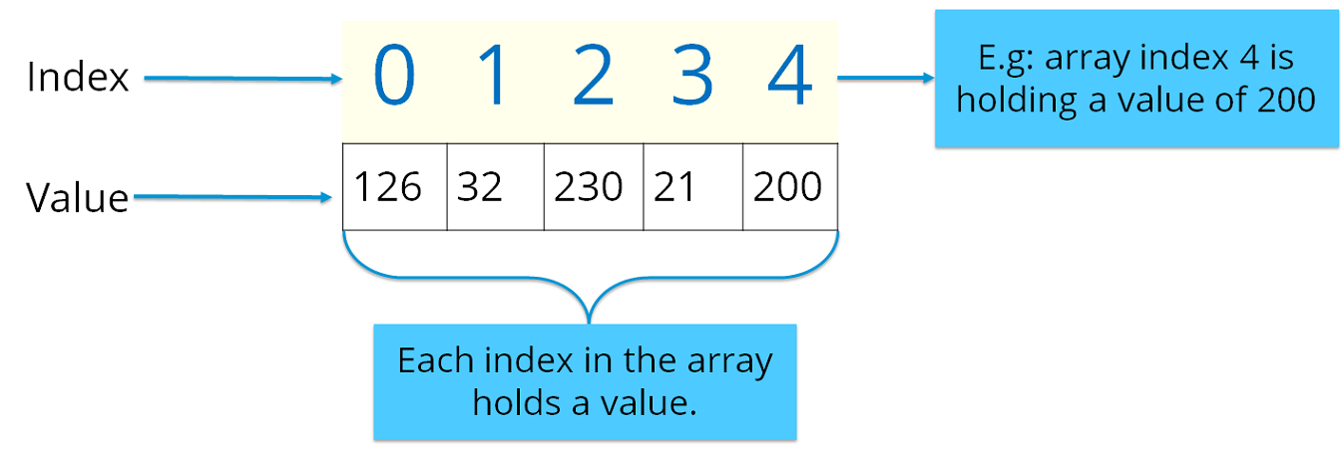
Def : - **Array** is Continues memory allocation which can store some **Homogenous** type of **data**.

Arrays in Java is similar to that of C++ or any other programming language.

An array is a data structure that holds **sequential elements of the same type.**

Let’s say you want to store 50 numbers. Instead of declaring individual variables, such as number0, number1, … and so on. You can declare one array variable – “numbers” and use number[0], number[1] to represent individual variables. This will ease your task and minimizes the redundancy.

Each array has two components: index and value. Refer to the below image for better understanding:



Here the indexing starts from zero and goes till (n-1) where n= size of the array. Let’s say you want to store 10 numbers, then the indexing starts from zero and goes till 9.

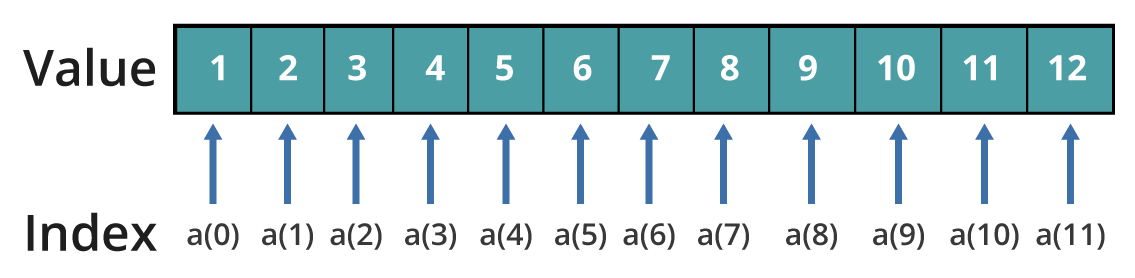
There are two types of arrays in Java:

* Single-dimension Array
* Multi-dimension Array

**Single-dimension Array:** In a single-dimension array, a list of variables of the same type can be accessed by a common name. You can initialize the array using the following syntax:

int a[] = new int[12];

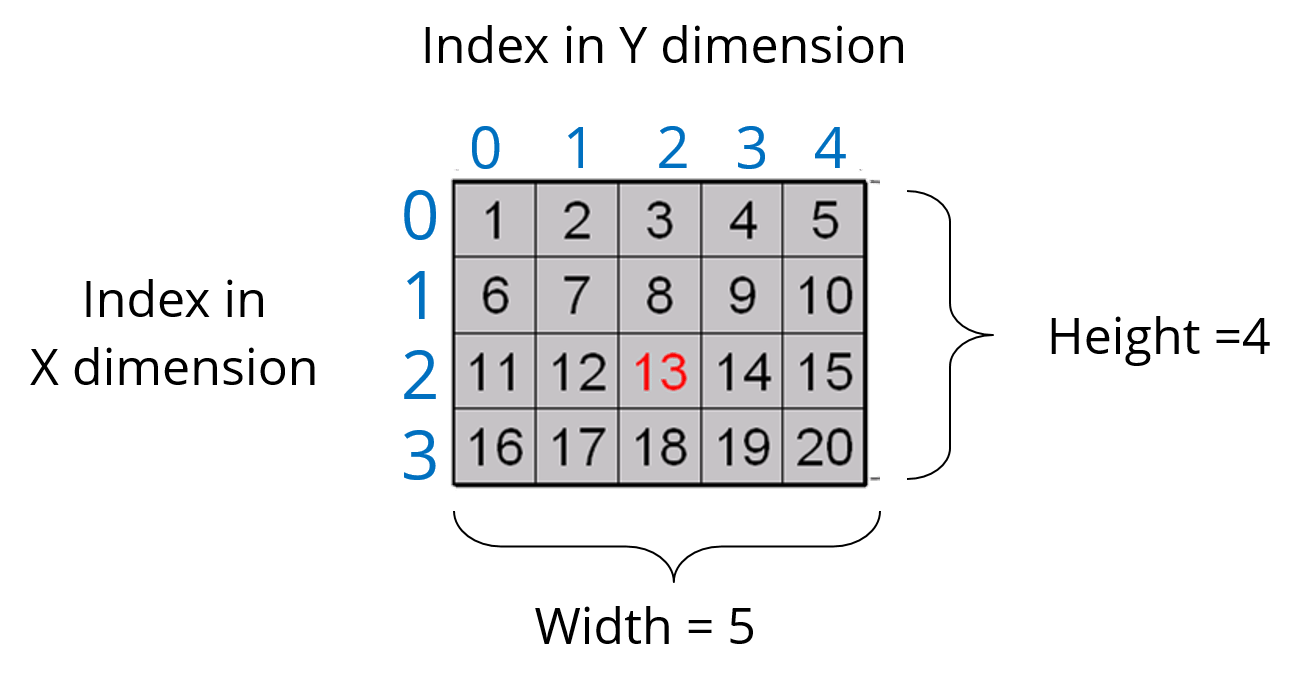
You can refer to the below image where I have stored data with respect to the given index.

****

**Multi–dimension Array:**In a multi-dimension array, your data is stored in a matrix form. Here, you can initialize the array using the following syntax:

int table[][]= new int[4][5];

It is quite similar to the matrix that we use in mathematics. Refer to the below image where I have stored data with respect to different dimensions.

Thus, arrays help you in optimizing the code where you can insert the data at any location.

Let’s see the below code to understand the concept of array in Java.

|  | **public** **class** Multi\_dimension\_Array  {  **public** **static** **void** main(String[] args)  {    **int**[][] arr ={  { 1, 2, 3 , 4 , 5 },  { 6, 7, 8, 9, 10 },  { 11, 12, 13, 14, 15 }    };    **for**(**int** i=0;i<5;i++)  {  **for**(**int** j=0;j<5;j++)  {  System.***out***.println( arr[0][0]);  }    }      }    } |
| --- | --- |

In the above code, I have explained how you can take input for the array and print the same.  
I hope you guys are clear with how an array looks like and how do you initialize one. Now, let’s summarize the above topics and see the entire structure of a Java program.

**Java Tutorial: Structure of a Program**

Till now, we have learned about member variables, data types, control statements, classes and objects. Let’s see how all of them are structured together in a class in Java.

|  | **public** **class** Array1 {  **public** **static** **void** main(String[] args)  {  **char** array\_Name[];//Array Decalration    array\_Name=**new** **char**[5]; // Storage declaration    array\_Name[0]='A';  array\_Name[1]='B';  array\_Name[2]='c';  array\_Name[3]='d';  array\_Name[4]='e';    **for**(**int** i=4;i>=2;i--)  {    System.***out***.println(array\_Name[i]);    }  System.***out***.println(" The Length is : " + array\_Name.length);    }  } |
| --- | --- |