1D Array

An array is a sequential collection of elements of same data type and stores data elements in a continuous memory location. The elements of an array are accessed by using an index. The index of an array of size N can range from 0 to N-1 . For example, if your array size is , then your index will range from 0 to 4 (5-1). Each element of an array can be accessed by using Index.

Consider following array. The size of this array is 5 . If you want to access 12, then you can access it by using arr[1]=12.

arr	4	12	7	15	9
index	0	1	2	3	4

Array declaration:

Declaring an array is language-specific.

ex:JAVA

int arr[];   //declaring array

arr = new int[20]; // allocating memory to array

int[] arr = new int[20]; // combining both statements in one

Notes:

1.The elements in the array allocated by new will automatically be initialized to zero (for numeric types), false (for boolean), or null (for reference types).Refer Default array values in Java

2.Obtaining an array is a two-step process. First, you must declare a variable of the desired array type. Second, you must allocate the memory that will hold the array, using new, and assign it to the array variable. Thus, in Java all arrays are dynamically allocated.

 int[] intArray = new int[]{ 1,2,3,4,5,6,7,8,9,10 };

 // Declaring array literal

- Accessing Java Array Elements using for Loop:

Each element in the array is accessed via its index. The index begins with 0 and ends at (total array size)-1. All the elements of array can be accessed using Java for Loop.

```
// accessing the elements of the specified array

for (int i = 0; i < arr.length; i++)

  System.out.println("Element at index " + i +

                " : "+ arr[i]);
```

What happens if we try to access element outside the array size?

Compiler throws ArrayIndexOutOfBoundsException to indicate that array has been accessed with an illegal index. The index is either negative or greater than or equal to size of array.