**One Dimensional Array in Java with Example**

An array with one dimension is called one-dimensional array or single dimensional array in java.

It is a list of variables (called elements or components) containing values that all have the same type.

One dimensional array represents one row or one column of array elements that share a common name and is distinguishable by index values.

For example, marks obtained by a student in five subjects can be represented by single-dimensional array because these marks can be written as one row or one column.

## Creating One dimensional Array in Java

There are basically two ways to create a one dimensional array in java that are as follows:

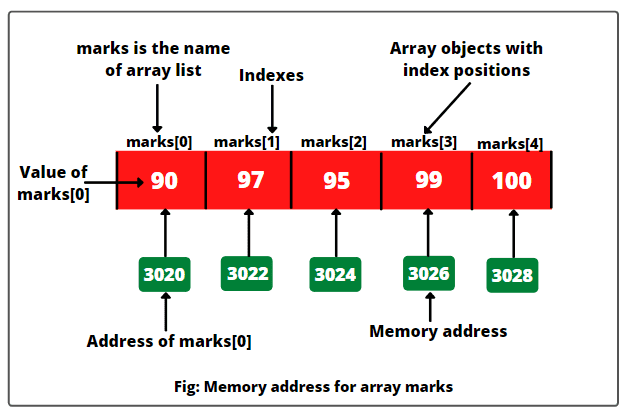
1. We can declare one-dimensional array and store values (or elements) directly at the time of its declaration, like this:

int marks[ ] = { 90, 97, 95, 99, 100 }; // declare marks[ ] and initialize with five values.

In this single dimensional array declaration, int represents integer type values that can be stored into the array. marks[ ] defines the name of array with one dimension.

A pair of square braces [ ] after array name represents one dimension. Then, we have mentioned marks obtained by five subjects (as elements) inside the curly { and } braces.

Now JVM will create five blocks represented by marks[0], marks[1], marks[2], and so on inside the memory with different addresses.

[](https://www.scientecheasy.com/2021/08/one-dimensional-array-in-java.html/)

Here, the values 0, 1, 2, 3, . . . . is called index value of array that refers to the element position in an array. In these blocks of memory, JVM stores five elements into an array, shown below:

This is the first way to declare single dimensional array with storing elements at the time of its declaration. Let’s move on to the second way to declare a one-dimensional array.

2. The second way of creating a one-dimensional array is by declaring array first and then allocating memory for it using the new keyword.  
The syntax is as follows:

int marks[ ]; // declare marks arrays.

marks = new int[5]; // allocating memory for storing 5 integer elements into an array.

The above two statements can also be written by combining them into a single statement, as:

int marks[ ] = new int[5];

JVM has allocated memory locations for storing five integer elements into the array. But, we have not stored actual elements into the array so far. To store elements into the array, we can write statements like these in the program:

marks[0] = 90;

marks[1] = 97;

marks[2] = 95;

marks[3] = 99;

marks[4] = 100;

Or, the programmer can pass the values from the keyboard to the array by using for loop, as follows:

for(int i = 0; i < 5; i++)

{

// Read the integer values from the keyboard and store them into marks[i].

marks[i] = Integer.parseInt(br.readLine));

}

Let us see some more examples for single dimensional array:

1. float salary[ ] = {99859.55f, 15000f, 14500.75f, 9050f};

2. float salary[ ] = new float[10];

3. char ch [ ] = {'s', 'c', 'i', 'e', 'n', 't', 'e', 'c', 'h', 'e', 'a', 's', 'y'};

4. char ch [ ] = new char [10] ;

5. String colors [ ] = {"Orange", "Red", "Sky blue", "Green"};

6. String names = new String[5];