**Array**

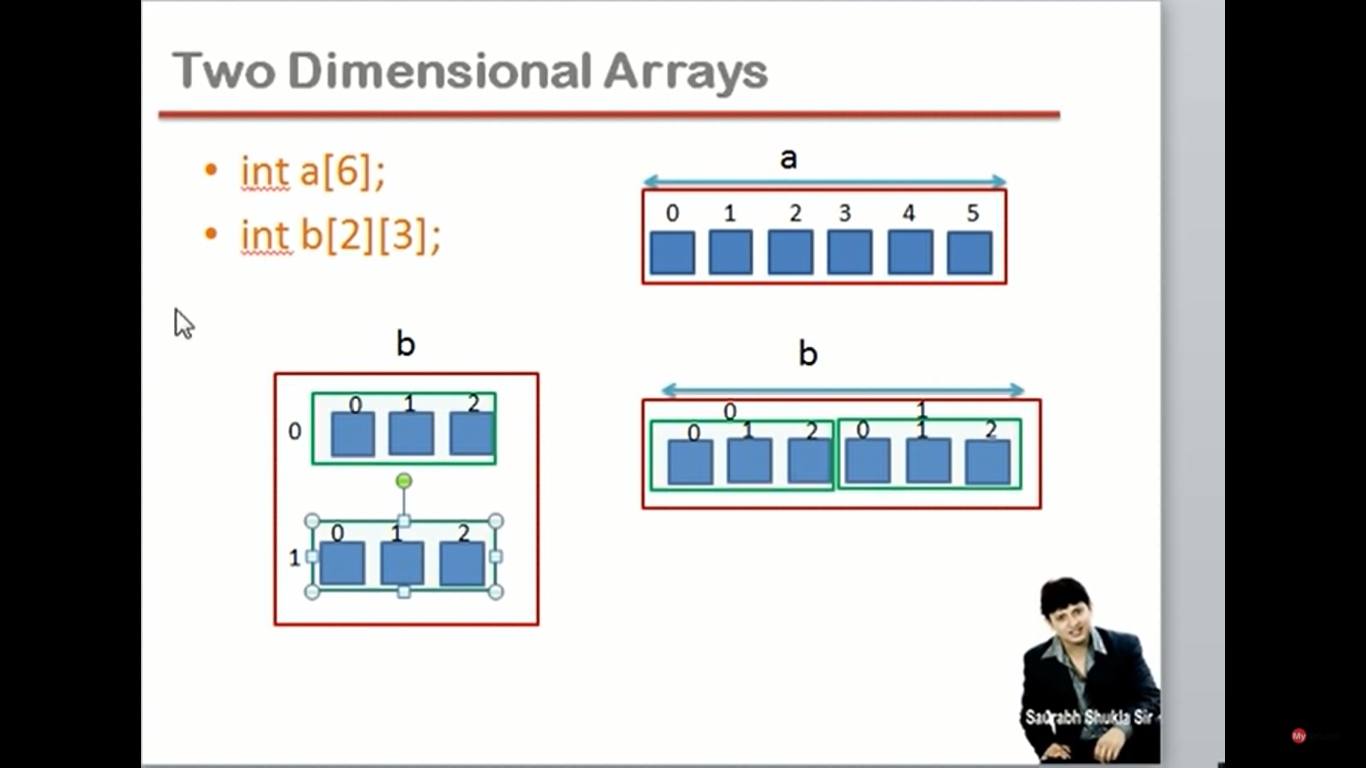
Array is a linear collection of similar elements or group of variable containing similar elements (like Element of only int datatype). It is also known as Subscript variable.

int a[10]; **//While declaring an array, means we are defining array size.**

a[0] =12; **//While using array, means we using its particular variable.**

**Note:**

* int a[]; **//Declaring an Empty array is an error.**We have to mention array size while declaring it. Ex: int a[10];
* int a[10]; While declaring an array, technically ‘**a**’ is the address of the first block/variable representation. For the sake of perception, we can also say ‘a’ is the name of the group of variable i.e. array.
* Int a[5] = {1, 2, 3, 4, 5}; In this way we can initialize the value in the array while declaring it. All value will store correspondingly in linear way.  
  But before initializing value in the array, All variable containing ‘**Garbage Value’** or we can say that an empty array contains Garbage Value.
* It is totally optional to mention the size of array only when initializing the values while declaring an array.  
  For Example: int a[] = {1, 2, 3, 4, 5};
* An array always consume memory continuously in linear way irrespective of its size.
* Initializing values more than the size of the array, will return an error i.e. too many initializes’.   
  For Ex: int a[5] = {1,2,3,4,5,6,7};  
  But we are free initialize values lower than the size of the array, Where the address of the remaining index variable of array will automatically contain ‘**0**’ value, *only when we are initializing values while declaring the array*.  
  For Ex: int a[5] = {1,2};



**One-Dimensional Array:**  
int a[6];   
In One-Dimensional array, space consume by array in the memory continuously linear in direction. That’s why it’s also known as Linear array.   
In one-dimensional array we create a group of variables which contains value/element. For ex: int a[6]; This array containing total 6 no. Of variables in it.

**Two Dimensional Arrays:**int b[2][3];  
Here, a ‘b’ array containing ‘2 blocks’ where each block having 3 number of variables/blocks. Two-dimensional array also know as ‘**Array of Arrays**.’ So total number of variable in the array = 2\*3 = 6 variables.

Two-dimensional array will also consume values continuously linear in direction technically but for the sake of perception, In two-dimensional array we can show the array in the form of 2D-matrix but not using One-dimensional array. So we can also consider the above 2-D array in row-column i.e. ( 2rows & 3 columns).

**Note:**Here, Both dimension of array will consume the same memory size.

**\*Why and when we need One-Dimensional, Two-Dimensional & 3-Dimensional array?**

**Ans:🡪** For the sake of ease to representing real-world data, we use various dimensions of array.   
For Example:-  
Let say, we need to represent the marks of 10 students of a class in a school. In this case, we will be using simple **One-Dimensional array** to represent the marks of all 10 students.  
Like: int C[10];

Let say, we need to represent the marks of every 10 students of each 5 classes in a school. In this case, we will be using **Two-Dimensional array** to represent the marks of all the students of 5 classes i.e. 50 students.  
Here either we can create 5 array of class containing 10 students to represent their marks.  
Like: int C1[10], C2[10], C3[10], C4[10], C5[10];  
OR, we can combine all the above array in the form of 2-Dimensional array, such that , an ‘S’ array representing a School containing 5 number of classes having 10 student in each class.  
Like: int S[5][10]; //Total 5\*10 = 50 students

Let say, we need to represent the marks of every 10 students of each 5 classes of 10 different schools. In this case, we will be using **Three-Dimensional array** to represent the marks of all the students of 5 classes of 10 different school i.e. 500 students.  
Here either we can create 10 2D- array of different schools containing 5 classes having 10 students in every 5 classes to represent their marks.  
Like: int S1[5][10], S2[5][10], S3[5][10], S4[5][10],.... S10[5][10];  
OR, we can combine all the above array in the form of 3-Dimensional array, such that , an ‘S’ array representing its name containing 10 different schools with 5 number of classes having 10 students in each class.  
Like: int S[10][5][10]; //Total 10\*5\*10 = 500 students