Arrays

An array is a derived data type, which can be used for storing multiple values.

If an application requires multiple values, then we can store those multiple values by declaring multiple variables. If we declare multiple variables in a program then the code size will increase, and readability will be reduced.

In order to reduce the code size, and improve the readability of the code, we take the help of arrays. Arrays in java language are classified into two types. They are

1) Single dimension array
2) Multi dimension array

Single dimension array: single dimension array is a collection of multiple values represented in the form of a single row or single column.

Syntax for declaring a single dimension array:
datatype arrayName[]; each pair of square bracket represents one dimension.

The name of the array can be any valid java identifier

Rule 1: At the time of array declaration we can specify the pair of [] either before the array name or after the array name.

Example:
```
int rollNo[];
double[] marks;
char []grade;
boolean result[];
```

Rule 2: At the time of array declaration we should not specify the size of the array.

Syntax for Creation of single dimension Array:
```
datatype arrayName[] = new datatype[size];
        Or
datatype arrayName[];
arrayName = new datatype[size];
```

Example:
```
int arr[] = new int[10];
        Or
int arr[];
arr=new int[10];
```

Rule: Specifying the size of the array at the time of array creation is mandatory and it should be of byte, short, int, char type only.

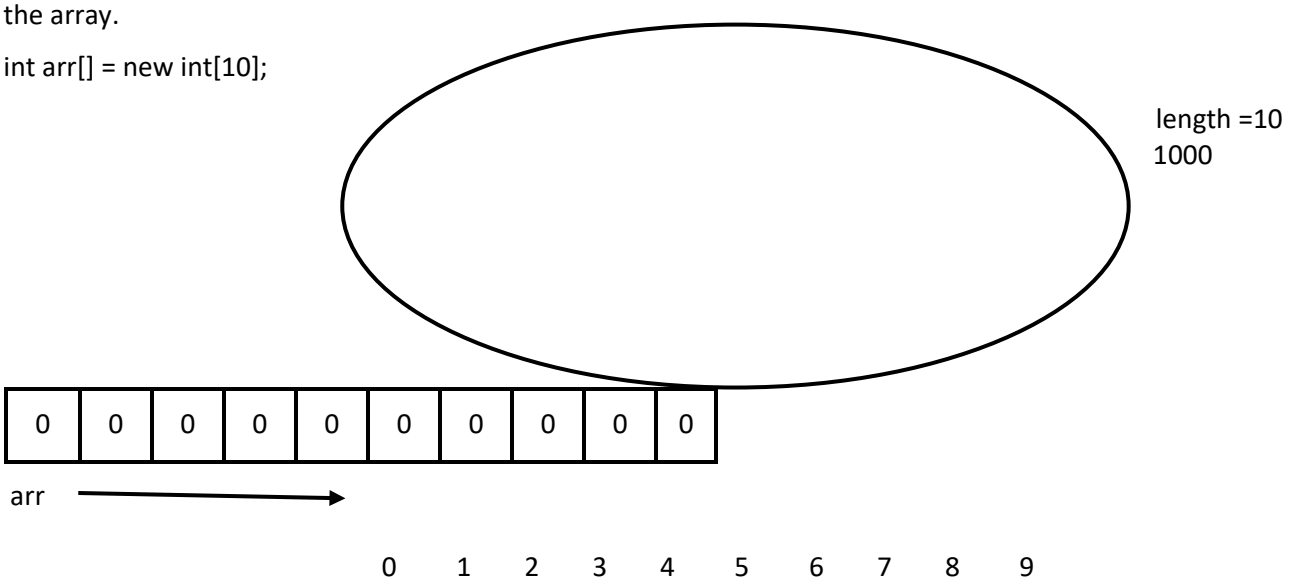
The size of the array must be a positive number. If a negative number is specified, then we get a runtime error called NegativeArraySizeException.

When an array is created, the memory for that array will be allocated in sequential memory locations. The amount of memory allocated to an array depends upon the size of the array and the size of the datatype of the array. Once the memory for the array is allocated, all the array elements will be initialized automatically with default values.

An array contains multiple values and for the entire array we have only one name. In order to access the array elements we take the help of index position. The index position will always begin with 0. The range of the index position will be 0 to size - 1.

Every array is internally an object and it contains a default variable called length, which represents the size of the array.

int arr[] = new int[10];



Syntax for accessing the array elements:
arrayName[index]

Example:
arr[0]
arr[1]

When we are accessing the array elements, we are supposed to specify the index position within the range otherwise we get runtime error called ArrayIndexOutOfBoundsException.

Syntax to assign a value to an array element:
arrayName[index] = value;

Example:
arr[0] = 10;
arr[1] = 20;

For each loop(enhanced for loop): This loop is introduced in java 1.5 version and it designed for accessing the elements from arrays(collection). It is also called as enhanced for loop.

Syntax:

```
for(declaration : arrayName)  {
   statements;
}
```

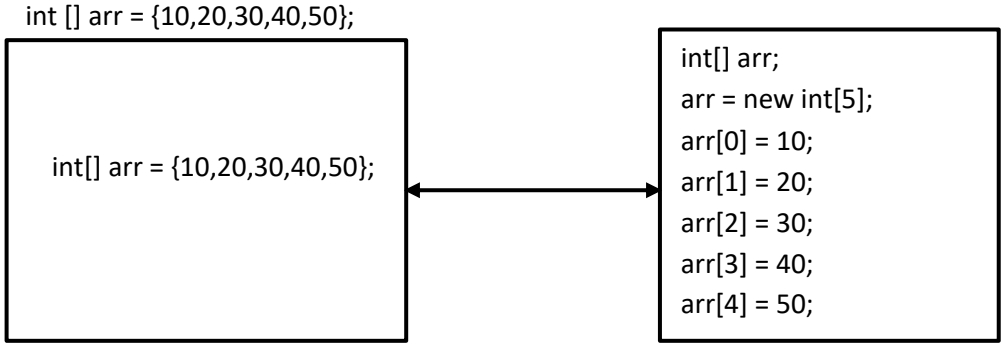
The declaration of a variable in for each loop must be same as that of the type of elements available in the array. In for each loop, the statements will be executed one time for each element available in the array and therefore called for each loop.

Declaring, creating, and assigning the array elements in a single line.
Syntax:
datatype arrayName[] = { list of values };

Within the { } we can specify any number of values separated by a comma.

- When we are declaring, creating and assigning the values in a single line, then we need not specify the size of the array.
- The creation of array will be done by JVM and the size of the array is also decided by JVM based on number o f values specified within the { }.
- Once the size of the array is decided the JVM itself will assign the values to the array element.

Example: int [] arr = {10,20,30,40,50};



Multi dimensional arrays: The multi dimension arrays in java will be represented in the form of array of arrays.

Syntax for declaring a two dimension array:
datatype arrayName[][];

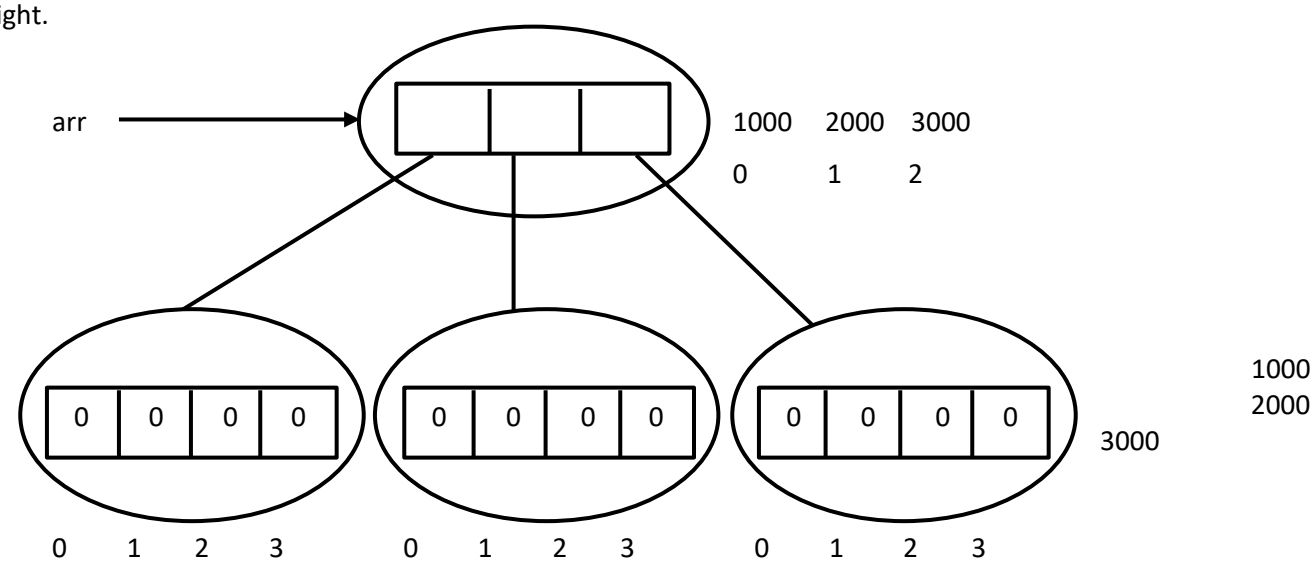
Rule 1: At the time of array declaration we can specify the pair of [][] either before the array name or after the array name.
int arr1[][];
int[][] arr2;
int [][]arr3;
int[] []arr4;
int []arr5[];
int[] arr6[];

Rule 2: At the time of array declaration we should not specify size of the array.

Syntax for creating 2 dimensional array :
datatype arrayName[][] = new datatype[size1][size2];
Or
datatype arrayName[][];
arrayName = new datatype[size1][size2];

Example:
int arr[][] = new int[3][4];
Or
int arr[][];
arr = new int[3][4];

Rule: In a multi dimension array specifying the first dimension of array is mandatory and the remaining are optional. If we are specifying the remaining dimensions then they must be specified in the order from left to right.



Jagged array: In multi dimension array of the arrays have unequal size then they are called Jagged array.

Example of Jagged array: