Course Name - Object Oriented Programming using Java

Array

Presented By
Dr. Sudipta Sahana
Asso. Prof.
Dept. of CSE
UEM – Kolkata

Topic of Interest

- Java Arrays
- Access the Elements of an Array
- Change an Array Element
- Array Length
- Loop Through an Array
- Multidimensional Arrays
- Anonymous & Reusable Arrays

Java Arrays

Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value. To declare an array, define the variable type with square brackets:

```
String[] cars = {"Volvo", "BMW", "Ford", "Mazda"};
```

Four Ways To define an Array

int x[]; // declare an int array
x=new int[10]; // allocate memory
	The first line declares an array of type int. **Not recommended by experts**
	The second line allocates memory for 10 elements with all initialized to 0

int x[]=new int[10];	This is nothing but a merged version of the above. **Not recommended by experts.**

int[] x;
x=new int[10];
	The first line declares an array of type int. **This is recommended.**
	The second line allocates memory for 10 elements with all initialized to 0

int [] x=new int[10];	**It's a merged version of the above and the this is recommended.**

Access the Elements of an Array

To access an array element by referring to the index number. This statement accesses the value of the first element in cars:

```
String[ ] cars = {"Volvo", "BMW", "Ford", "Mazda"};
System.out.println(cars[0]);
```

Change an Array Element

To change the value of a specific element, refer to the index number:

```
String[] cars = {"Volvo", "BMW", "Ford", "Mazda"};
cars[0] = "Opel";
System.out.println(cars[0]);
```

Array Length

To find out how many elements an array has, use the length property:

```
String[] cars = {"Volvo", "BMW", "Ford", "Mazda"};
System.out.println(cars.length);
```

Loop Through an Array

We can loop through the array elements with the for loop, and use the length property to specify how many times the loop should run. The following example outputs all elements in the **cars** array:

```
String[ ] cars = {"Volvo", "BMW", "Ford", "Mazda"};
for (int i = 0; i < cars.length; i++) { System.out.
println(cars[i]);
}
```

Multidimensional Arrays

- Java deals multi dimensional arrays as Array of Arrays.
- We can declare a 2-D int array as…

1. int[][] x=new int[4][3]; // 4 rows & 3 cols per row
2. int[][] x=new int[4][]; // 4 rows and cols to be dynamically alocated the size

   x[0]=new int[5]; // means 5 cols in 0-th row

   x[2]=new int[3]; // means 3 cols in 2-th row (3rd row)
3. int[][] x=new int[][5]; // Error; u can’t fix-up cols size without rows.
4. int[][] x={{1,2,3},{4,5,6,7},}; // first row has 3 cols where as second row has 4 cols
5. In general, for a 2-D array *myArray, myArray.length* is the number of rows and *myArray[i].length* is the number of cols in the i-th row

Anonymous & Reusable Arrays

- Look at the following code…

  System.out.println(new int[]{0,1,2,3,4}.length);

  This gives an output 5 as this is an array which has no name.

- *int[] x={1,2,3,4,5,6,7};*

  // codes go here

  x=new int[90]; // the previous elements are lost

Thus arrays can be reused

Thank You