JPL :: Multidimensional Arrays

TalentSprint

Licensed To Skill

Version 1.0.4

Learning Objectives

By the end of this Presentation, you will be able to:

- Learn how to find Array size.
- Understand enhanced for loop
- Construct, Initialize and Manipulate Multidimensional arrays

Multidimensional Arrays

Array length

We can refer to array length by using length attribute of array object i.e. array_name.length

```
int arr1 = new int[10]; arr1.length – 10
int arr2 = new int[15]; arr2.length – 15
int arr3 = new int[4]; arr3.length – 4
for (int i = 0; i < 10; i++) {   }
for (int i = 0; i < arr1.length; i++) {   }
```

Multidimensional Arrays

Accessing array elements one by one

```
class ArrayDemo {
    public static void main ( String[] args ) {
        int[]  array = { 20, 19, 12, 25} ;
        for (int indx = 0; indx < array.length; indx++ )
            System.out.println(array[indx]);
    }
}
```

Multidimensional Arrays

Another way to access array elements one by one

```
class Demo {
    public static void main ( String[] args ) {
        int[] array = { 20, 19, 12, 25} ;
        for ( int num : array )
            System.out.println(num);
    }
}
```

Multidimensional Arrays

1. Write a Java program to store a student marks of 5 subjects
2. Now write java program to store more than one student marks of 5 subjects

Multidimensional Arrays

Multi Dimensional arrays

2 - Dimensional arrays

3 - Dimensional arrays

Jagged arrays

Multidimensional Arrays

Declaring Two Dimensional array variable We can declare 2D array either of the following way. <type>[][] variable_name; <type>[] variable_name[]; <type> variable_name[][];

Example

```
int [][]  cse;
int[]  cse[];
int cse [][];
```

Array is not created at declaration and memory not allocated

Multidimensional Arrays

Defining Two Dimensional array

variable_name = **new** <type>[rows][columns];

Example:
cse = **new int**[3][4];

Declaring and Defining in the same statement:
int [][] cse = **new int**[3][4];
cse.length = 3
cse[0].length = 4
cse[1].length = 4
cse[2].length = 4

In the above example, 3X4 = 12 integer variables will be created and 48 bytes of memory will be allocated.

Multidimensional Arrays

Initializing 2D array

```
for (i = 0; i < cse.length; i++)
    for (j = 0; j < cse[i].length; j++)
        cse[i][j] = xxxxx;
```

Multidimensional Arrays

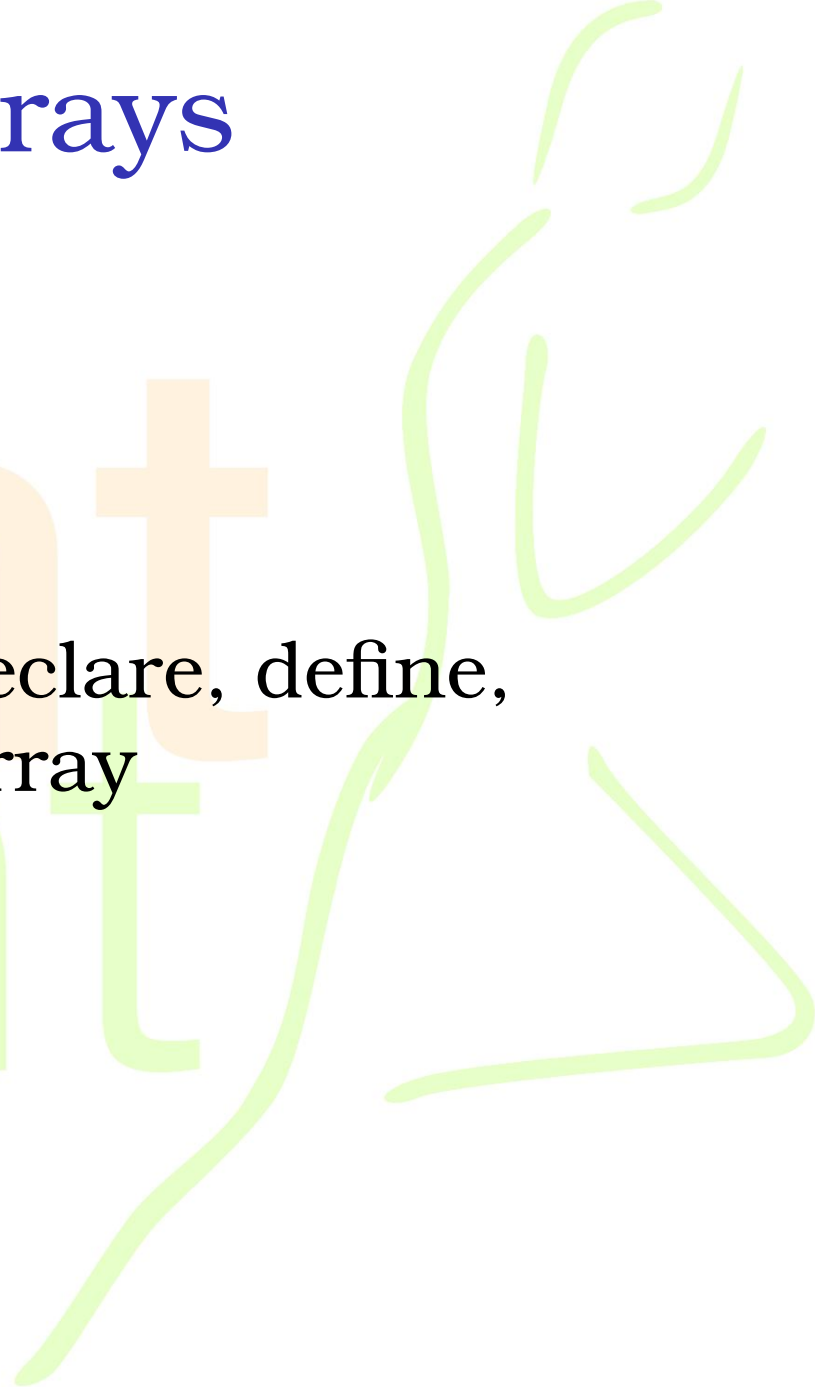
Declaring, Defining and Initializing 2D array at the same time

Initializing an Array

```
int [][]  cse = { {20, 19, 12, 25},
                  {23, 12, 18, 21},
                  {16, 18, 17, 20} };
```

Multidimensional Arrays

Write a Java program to declare, define, initialize and display 2D array



Multidimensional Arrays

Defining Three dimensional array

```
int [][][]   college = new int[2][2][4];
college.length = 2
college[0].length = 2
college[1].length = 2
college[0][0].length = 4
college[0][1].length = 4
college[1][0].length = 4
college[1][1].length = 4
```

Multidimensional Arrays

Defining Jagged array

```
int [][]  cse = new int[3][];

cse[0] = new int[4];
cse[1] = new int[2];
cse[2] = new int[3];

cse.length = 2
cse[0].length = 4
cse[1].length = 2
cse[2].length = 3
```

Multidimensional Arrays

Write a Java program to declare, define, initialize and display Jagged array

Multidimensional Arrays

1. Write a Java program to transpose a given 3 X 3 matrix.
2. Write a Java program to test a given matrix is lower triangle or not.
3. Write a Java program to test a given matrix is upper triangle or not.

Multidimensional Arrays