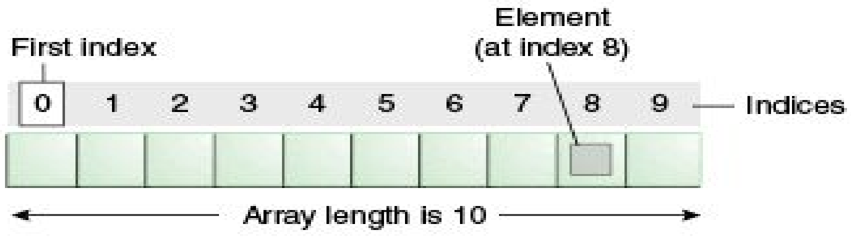
CSE 215L: Programming Language II Lab
Section: 7
Fall 2020

Arrays: a collection of a fixed number of data and those must be a single type of data.

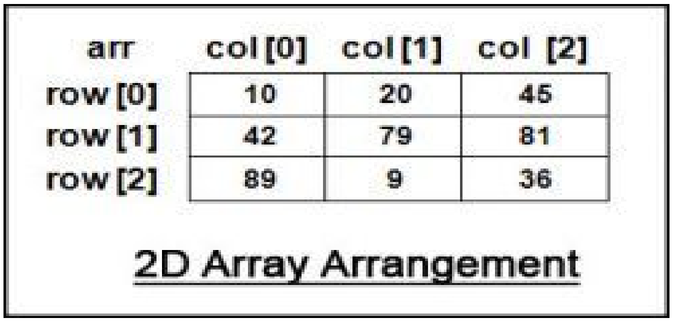
Single- dimensional Array:



Declaration	Declaration+Instantiation	Example

dataType[] arr;
dataType []arr;
dataType arr[];
	dataType[] arr = new dataType[size];	int arr[] = {1,2,3};
		int arr[] = new int [3];

Multidimensional Array:

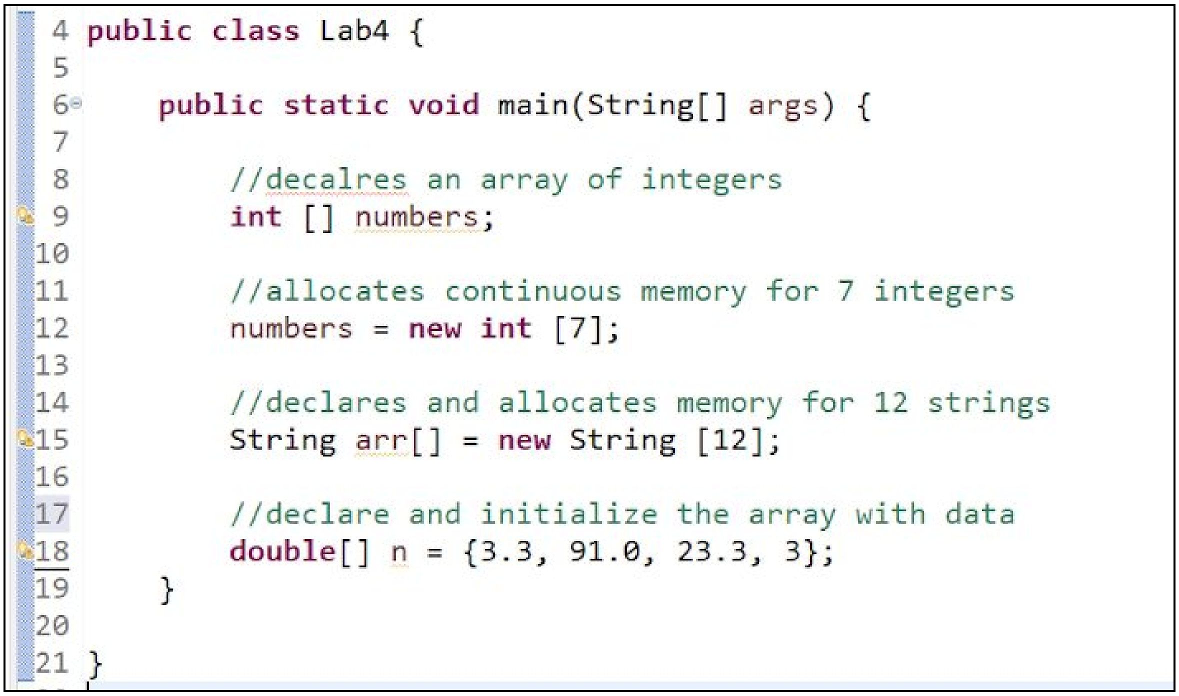


arr	col [0]	col [1]	col [2]
row [0]	10	20	45
row [1]	42	79	81
row [2]	89	9	36

2D Array Arrangement

Declaration	Declaration+Instantiation	Example

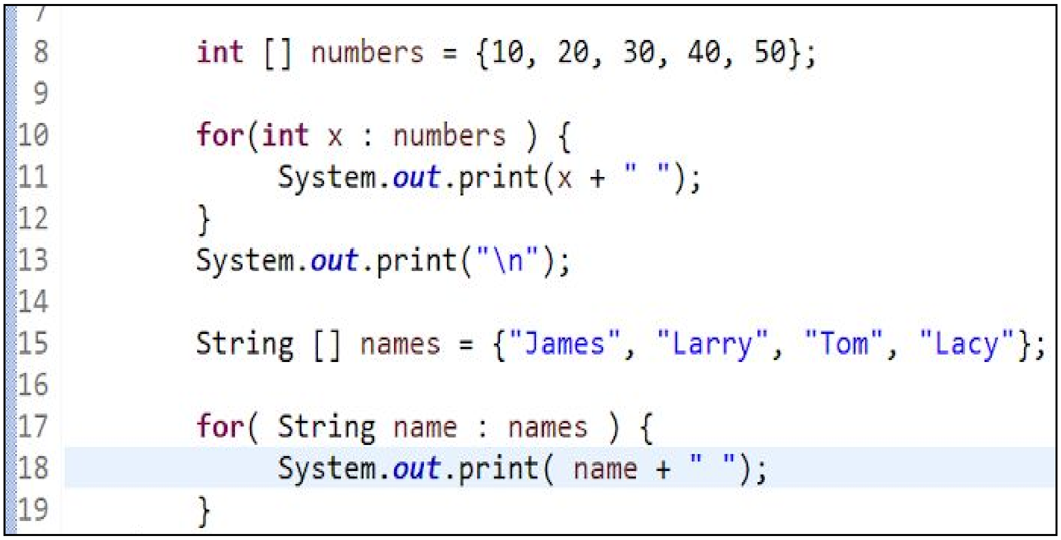
dataType[][] arr;
dataType [][]arr;
dataType arr[][];
dataType []arr[];
	dataType[][] arr = new dataType[size][size];	int[][] arr = new int [2][3];
		//2D array
		Int[][][] arr = new int[2][3][4];
		//3D array



```java
public class Lab4 {

    public static void main(String[] args) {

        //decalres an array of integers
        int [] numbers;

        //allocates continuous memory for 7 integers
        numbers = new int [7];

        //declares and allocates memory for 12 strings
        String arr[] = new String [12];

        //declare and initialize the array with data
        double[] n = {3.3, 91.0, 23.3, 3};
    }

}
```

for-each loop:

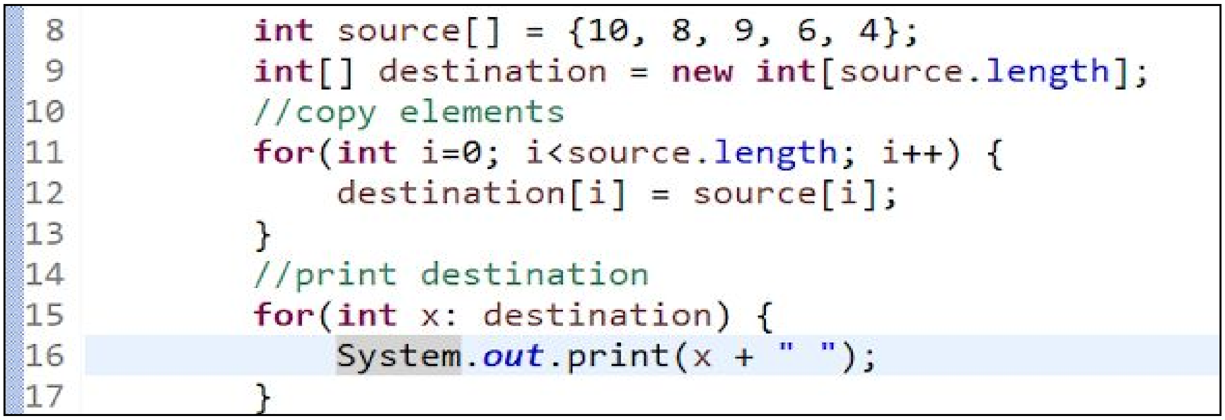
- for-each loop can be used to iterate over an array
- Use keyword 'for'
- Declare a variable of the same data type as that of the array
- Put a ':' followed by the array name



```java
        int [] numbers = {10, 20, 30, 40, 50};

        for(int x : numbers ) {
            System.out.print(x + " ");
        }
        System.out.print("\n");

        String [] names = {"James", "Larry", "Tom", "Lacy"};

        for( String name : names ) {
            System.out.print( name + " ");
        }
```

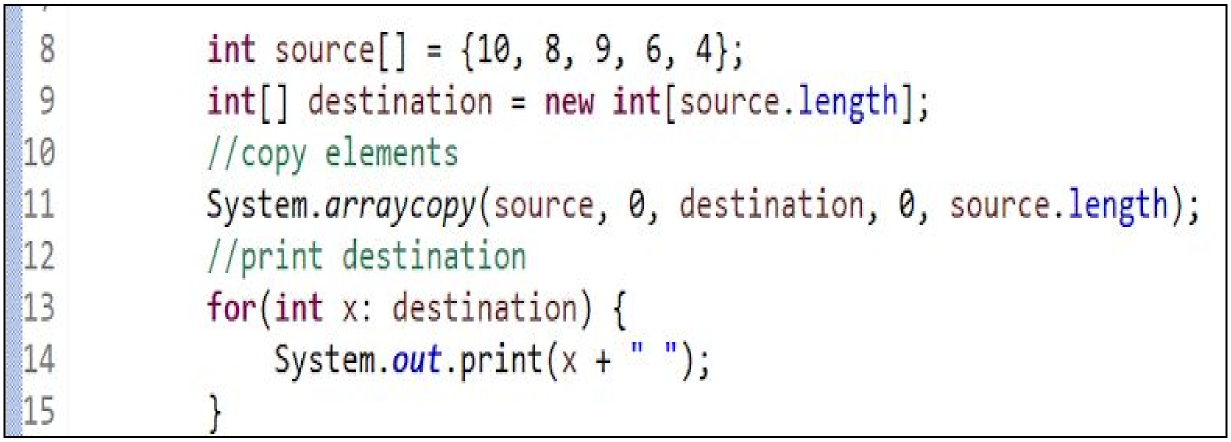
<u>Copy Array:</u>

Example1:



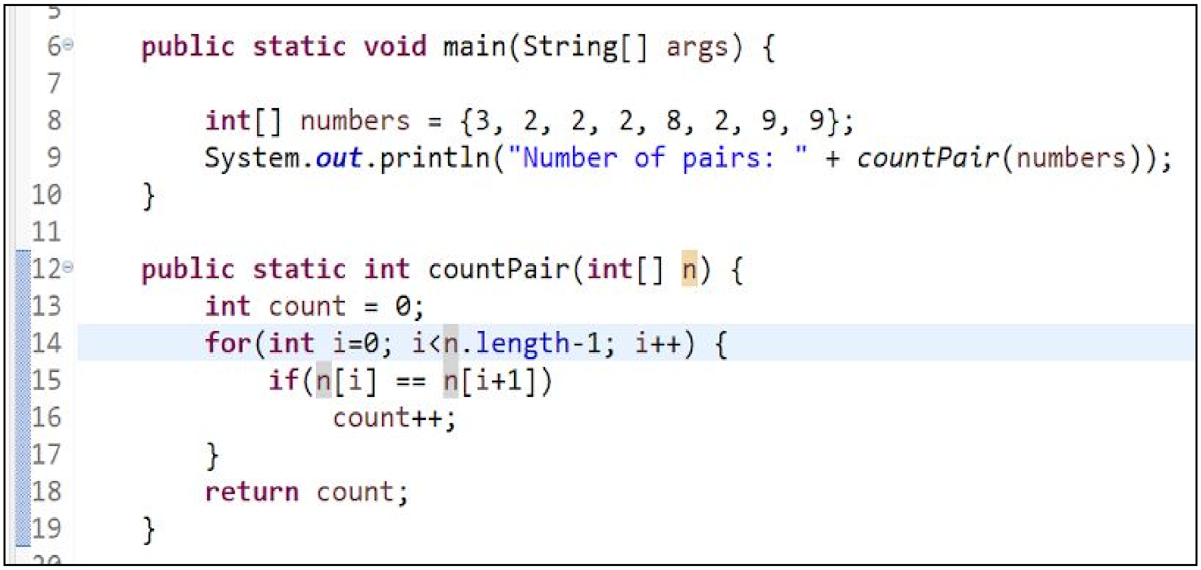
```
        int source[] = {10, 8, 9, 6, 4};
        int[] destination = new int[source.length];
        //copy elements
        for(int i=0; i<source.length; i++) {
            destination[i] = source[i];
        }
        //print destination
        for(int x: destination) {
            System.out.print(x + " ");
        }
```

Example2:



```
        int source[] = {10, 8, 9, 6, 4};
        int[] destination = new int[source.length];
        //copy elements
        System.arraycopy(source, 0, destination, 0, source.length);
        //print destination
        for(int x: destination) {
            System.out.print(x + " ");
        }
```

Arrays and Methods:



```
    public static void main(String[] args) {

        int[] numbers = {3, 2, 2, 2, 8, 2, 9, 9};
        System.out.println("Number of pairs: " + countPair(numbers));
    }

    public static int countPair(int[] n) {
        int count = 0;
        for(int i=0; i<n.length-1; i++) {
            if(n[i] == n[i+1])
                count++;
        }
        return count;
    }
```

Practice problems:

1. Write a program that will declare an array of 10 integers and assign 10 random (use Math.random()) integers (integers should be in the range: 0 - 50). Find the average of the array elements.
2. Take an integer array and print only the numbers that are in consecutive orders of 3. If there is no such number print “None”.

   For example:

Sample input	Sample output
Enter size: 12 Enter numbers: 1 2 3 2 2 2 11 4 4 4 3 3	The numbers: 2 4

3. Declare an integer array of size 6, initialize it with user input, calculate and print the average. Now calculate the percentage of numbers that are above that average.

   For example: if 3 of the array elements are greater than average, percentage is: 3 * 100 / 6 = 50%

Sample input	Sample output
Enter numbers: 88 32 6 12 4 59	Average: 33.5 Percentage of numbers greater than average: 33.33%

4. Take a 3X3 array and initialize it with these values: 3 4 9
   2 9 11
   4 5 0

   Calculate and print the sum for each row, column and both diagonals.

Sample output:
Sum of row 1: 16 Sum of row 2: 22 Sum of row 3: 9 Sum of column 1: 9 Sum of column 2: 18 Sum of column 3: 20 Sum of the diagonals: 12 and 22

5. Write a program to find the frequency of each element in the array.

Sample input	Sample output
Enter size: 6 Enter numbers: 11 0 93 0 11 11	11 occurs 3 times 0 occurs 2 times 93 occurs 1 time