**Guided Lab- 303.6.1 - Array**

**Lab Objective**

The goal of this lab is to provide you with hands-on experience working with Java arrays and iterating over an Array using loops.

By the end of this lab you will be able to create and use arrays in Java.

**Instructions**

# Example 1: Access Array Elements

Create a class named **arraydemoOne** and write the below code.

| pubic class arraydemoOne {  public static void main(String[] args) {    *// create an array*  int[] age = {12, 4, 5, 2, 5};  *// access each array elements*  System.out.println("Accessing Elements of an Array:");  System.out.println("First Element: " + age[0]);  System.out.println("Second Element: " + age[1]);  System.out.println("Third Element: " + age[2]);  System.out.println("Fourth Element: " + age[3]);  System.out.println("Fifth Element: " + age[4]);  } } |
| --- |

**Output**

Accessing Elements of an Array:

* First Element: 12
* Second Element: 4
* Third Element: 5
* Fourth Element: 2
* Fifth Element: 5

In the above example, notice that we are using the index number to access each element of the array.

We can use loops to access all the array elements at once.

# Example 2: Using for Loop

Create a class named **arraydemoTwo** and write the code below in it

| public class Main {  public static void main(String[] args) {    *// create an array*  int[] age = {12, 4, 5};   *// loop through the array*  *// using for loop*  System.out.println("Using for Loop:");  for(int i = 0; i < age.length; i++) {  System.out.println(age[i]);  }  } } |
| --- |

**Output**

Using for Loop:

12

4

5

In the above example, we are using the ***for Loop*** in Java to iterate through each element of the array. Notice the expression inside the loop,

age.length

# 

# 

# Example 3: Iterating Over an Array using EnhancedForLoop

Create a class named **EnhancedForLoop** and write the code below.

| public class EnhancedForLoop {  public static void main(String[] args) {  String[] names = { "New York", "Dallas", "Las Vegas", "Florida" };  for (String name : names) {  System.out.println(name);  }  }  } |
| --- |

**Output**

| New York  Dallas  Las Vegas  Florida |
| --- |

# 

# 

# Example 4: Compute the Sum and Average of Array Elements

Create a class named **arraydemothree** and write the code below.

| public class **arraydemothree** {  public static void main(String[] args) {   int[] numbers = {2, -9, 0, 5, 12, -25, 22, 9, 8, 12};  int sum = 0;  Double average;    *// access all elements using for each loop*  *// add each element in sum*  for (int number: numbers) {  sum += number;  }    *// get the total number of elements*  int arrayLength = numbers.length;   *// calculate the average*  *// convert the average from int to double*  average = ((double)sum / (double)arrayLength);   System.out.println("Sum = " + sum);  System.out.println("Average = " + average);  } } |
| --- |

**Output:**

Sum = 36

Average = 3.6

In the above example, we have created an array of named numbers. We have used them ***for...each loop*** to access each array element.

Inside the loop, we calculate the sum of each element. Notice the line:

int arrayLength = number.length;

Here, we are using the length attribute of the array to calculate the size of the array. We then calculate the average using:

average = ((double)sum / (double)arrayLength);

As you can see, we are converting the **int** value into a **double**. This is called "type casting” in Java.

# Example 5: Mean and Standard Deviation

Find the mean and standard deviation of the numbers kept in an array.

Create a class named **MeanSDArray** and write the code below.

| public class MeanSDArray {  public static void main(String[] args) { *// Declare variable*  int[] marks = {74, 43, 58, 60, 90, 64, 70};  int sum = 0;  int sumSq = 0;  double mean, stdDev;   *// Compute sum and square-sum using loop*  for (int i = 0; i < marks.length; ++i) {  sum += marks[i];  sumSq += marks[i] \* marks[i];  }  mean = (double)sum / marks.length;  stdDev = Math.sqrt((double)sumSq / marks.length - mean \* mean);   *// Print results*  System.out.printf("Mean is: %.2f%n", mean);  System.out.printf("Standard deviation is: %.2f%n", stdDev);  } } |
| --- |

**Output:**

Mean is: 65.57

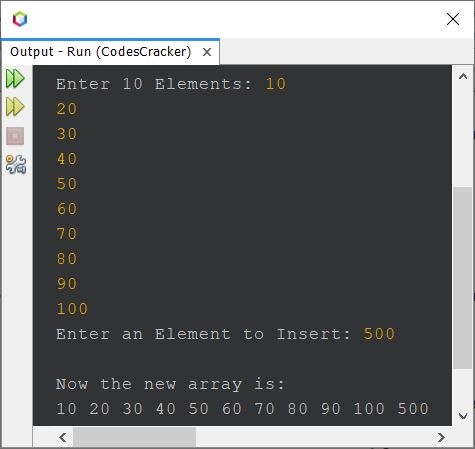
Standard deviation is: 13.56

# Example 6: Insert an Element at the end of an Array in Java

Create a class named **insertElements** and write the code below.

| import java.util.Scanner;  public class insertElements  {  public static void main(String[] args)  {  int i, element;  int[] arr = new int[11];  Scanner scan = new Scanner(System.in);    System.out.print("Enter 10 Elements: ");  for(i=0; i<10; i++)  arr[i] = scan.nextInt();    System.out.print("Enter an Element to Insert: ");  element = scan.nextInt();  arr[i] = element;    System.out.println("\nNow the new array is: ");  for(i=0; i<11; i++)  System.out.print(arr[i]+ " ");  }  } |
| --- |

The snapshot given below shows the sample run of the above program, with user input **10, 20, 30, 40, 50, 60, 70, 80, 90, 100** as ten elements and **500** as the new element to insert at the end of array:



**Submission Instructions:**

Include the following deliverables in your submission -

* + Submit your source code using the Start Assignment button in the top right corner of the assignment page in Canvas.

**CANVAS STAFF USE ONLY: Canvas Submission Guideline:**

| **Instructions for Canvas Assignment Creation** |
| --- |
| **Assignment Name: GLAB - 303.6.1 - Array**  **Points:** **100**  **Assignment Group: Module 303: Java SE Review (Not Graded)**  **Display Grade As: Complete/Incomplete**  **Do not count this assignment towards the final grade: Checked**  **Submission Types: File Uploads**  **Everything else is the default.** |

## 