

Java - Introduction to Programming

Lecture 10

Arrays In Java

Arrays in Java are like a list of elements of the same type i.e. a list of integers, a list of booleans etc.

- a. Creating an Array (method 1) - with **new** keyword

```
int[] marks = new int[3];  
marks[0] = 97;  
marks[1] = 98;  
marks[2] = 95;
```

- b. Creating an Array (method 2)

```
int[] marks = {98, 97, 95};
```

- c. Taking an array as an input and printing its elements.

```
import java.util.*;  
  
public class Arrays {  
    public static void main(String args[]) {  
        Scanner sc = new Scanner(System.in);  
        int size = sc.nextInt();  
        int numbers[] = new int[size];  
  
        for(int i=0; i<size; i++) {  
            numbers[i] = sc.nextInt();  
        }  
  
        //print the numbers in array  
        for(int i=0; i<arr.length; i++) {  
            System.out.print(numbers[i]+" ");  
        }  
    }  
}
```

Homework Problems

1. Take an array of names as input from the user and print them on the screen.

```
import java.util.*;

public class Arrays {

    public static void main(String args[]) {

        Scanner sc = new Scanner(System.in);

        int size = sc.nextInt();

        String names[] = new String[size];

        //input

        for(int i=0; i<size; i++) {

            names[i] = sc.next();

        }

        //output

        for(int i=0; i<names.length; i++) {

            System.out.println("name " + (i+1) + " is : " + names[i]);

        }

    }

}
```

2. Find the maximum & minimum number in an array of integers.

MIN_VALUE returns the minimum value the data Type can take, MAX_VALUE returns the maximum value the Data type can take

[HINT : Read about [Integer.MIN_VALUE](#) & [Integer.MAX_VALUE](#) in Java]

```
import java.util.*;

public class Arrays {

    public static void main(String args[]) {

        Scanner sc = new Scanner(System.in);

        int size = sc.nextInt();

        int numbers[] = new int[size];

        //input

        for(int i=0; i<size; i++) {

            numbers[i] = sc.nextInt();

        }

        int max = Integer.MIN_VALUE;

        int min = Integer.MAX_VALUE;

        for(int i=0; i<numbers.length; i++) {

            if(numbers[i] < min) {

                min = numbers[i];

            }

            if(numbers[i] > max) {

                max = numbers[i];

            }

        }

    }

}
```

```

        System.out.println("Largest number is : " + max);

        System.out.println("Smallest number is : " + min);

    }
}

```

3. Take an array of numbers as input and check if it is an array sorted in ascending order.

Eg : { 1, 2, 4, 7 } is sorted in ascending order.

{3, 4, 6, 2} is not sorted in ascending order.

```

import java.util.*;

public class Arrays {

    public static void main(String args[]) {

        Scanner sc = new Scanner(System.in);

        int size = sc.nextInt();

        int numbers[] = new int[size];

        //input

        for(int i=0; i<size; i++) {

            numbers[i] = sc.nextInt();

        }

        boolean isAscending = true;
    }
}

```

```
        for(int i=0; i<numbers.length-1; i++) { // NOTICE numbers.length - 1 as
termination condition

            if(numbers[i] > numbers[i+1]) { // This is the condition for
descending order

                isAscending = false;

            }

        }

        if(isAscending) {

            System.out.println("The array is sorted in ascending order");

        } else {

            System.out.println("The array is not sorted in ascending order");

        }

    }

}
```