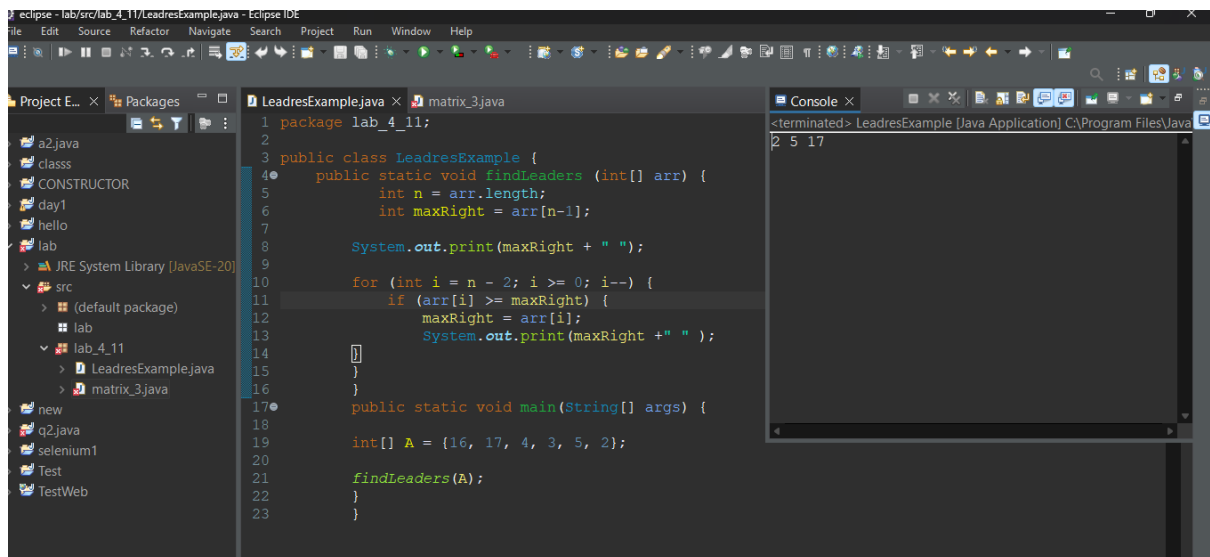


Q 1 Given an array A of positive integers. Your task is to find the leaders in the array. An element of array is leader if it is greater than or equal to all the elements to its right side. The rightmost element is always a leader.



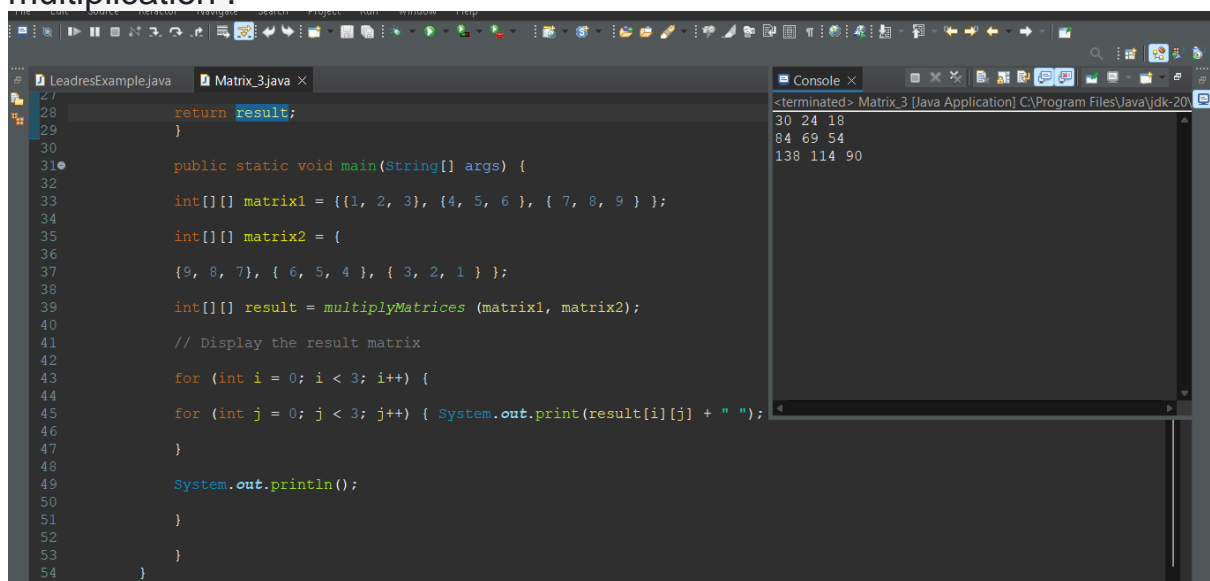
The screenshot shows the Eclipse IDE with a project named 'lab'. The package 'lab\_4\_11' contains two files: 'LeadresExample.java' and 'matrix\_3.java'. The 'LeadresExample.java' file contains the following code:

```
1 package lab_4_11;
2
3 public class LeadresExample {
4     public static void findLeaders (int[] arr) {
5         int n = arr.length;
6         int maxRight = arr[n-1];
7
8         System.out.print(maxRight + " ");
9
10        for (int i = n - 2; i >= 0; i--) {
11            if (arr[i] >= maxRight) {
12                maxRight = arr[i];
13                System.out.print(maxRight + " ");
14            }
15        }
16    }
17
18    public static void main(String[] args) {
19        int[] A = {16, 17, 4, 3, 5, 2};
20
21        findLeaders(A);
22    }
23 }
```

The console output shows the result of the program execution:

```
<terminated> LeadresExample [Java Application] C:\Program Files\Java
2 5 17
```

Q 2 Write a program to define two 3 \* 3 matrix and perform matrix multiplication .



The screenshot shows the Eclipse IDE with a project named 'lab'. The package 'lab\_4\_11' contains two files: 'LeadresExample.java' and 'Matrix\_3.java'. The 'Matrix\_3.java' file contains the following code:

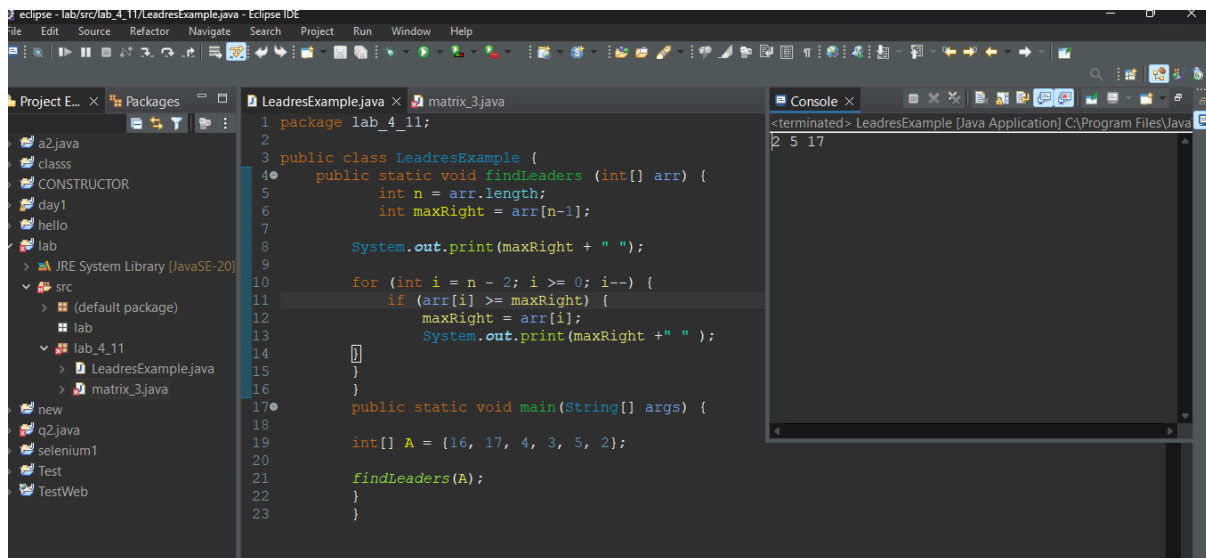
```
27
28     return result;
29 }
30
31 public static void main(String[] args) {
32
33     int[][] matrix1 = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};
34
35     int[][] matrix2 = {
36
37     {9, 8, 7}, {6, 5, 4}, {3, 2, 1}};
38
39     int[][] result = multiplyMatrices (matrix1, matrix2);
40
41     // Display the result matrix
42
43     for (int i = 0; i < 3; i++) {
44         for (int j = 0; j < 3; j++) { System.out.print(result[i][j] + " ");
45         }
46     }
47
48     System.out.println();
49
50 }
51
52 }
53
54 }
```

The console output shows the result of the program execution:

```
<terminated> Matrix_3 [Java Application] C:\Program Files\Java\jdk-20
30 24 18
84 69 54
138 114 90
```

Q 3 write a program to define a 3\*3 array and assign data within program . Ask a number from user and search if that number exists in array or not . If particular number exist with array then print “ **found** Message “ and also print row number and column of number if found .

Print “**data does not exists**” if number not found.

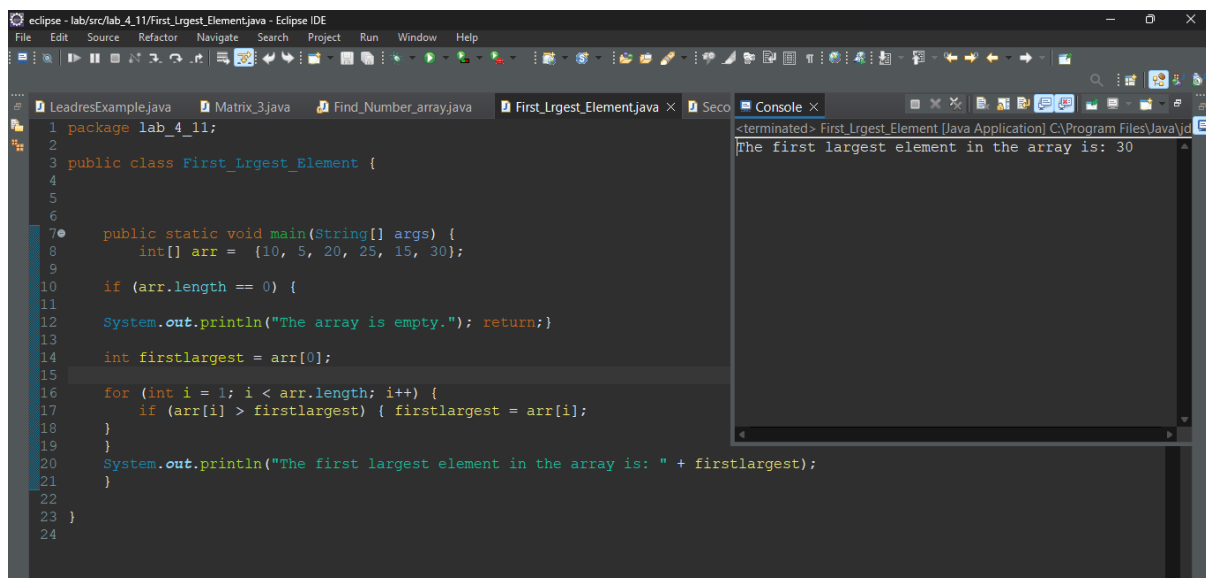


The screenshot shows the Eclipse IDE with a project named 'lab\_4\_11'. The package 'lab\_4\_11' contains two files: 'LeadresExample.java' and 'matrix\_3.java'. The 'LeadresExample.java' file contains the following code:

```
1 package lab_4_11;
2
3 public class LeadresExample {
4     public static void findLeaders (int[] arr) {
5         int n = arr.length;
6         int maxRight = arr[n-1];
7
8         System.out.print(maxRight + " ");
9
10        for (int i = n - 2; i >= 0; i--) {
11            if (arr[i] >= maxRight) {
12                maxRight = arr[i];
13                System.out.print(maxRight + " ");
14            }
15        }
16    }
17
18    public static void main(String[] args) {
19        int[] A = {16, 17, 4, 3, 5, 2};
20
21        findLeaders(A);
22    }
23 }
```

The console output shows the result of the program execution: "2 5 17".

4. Write a Java program to find the first largest element in an array.

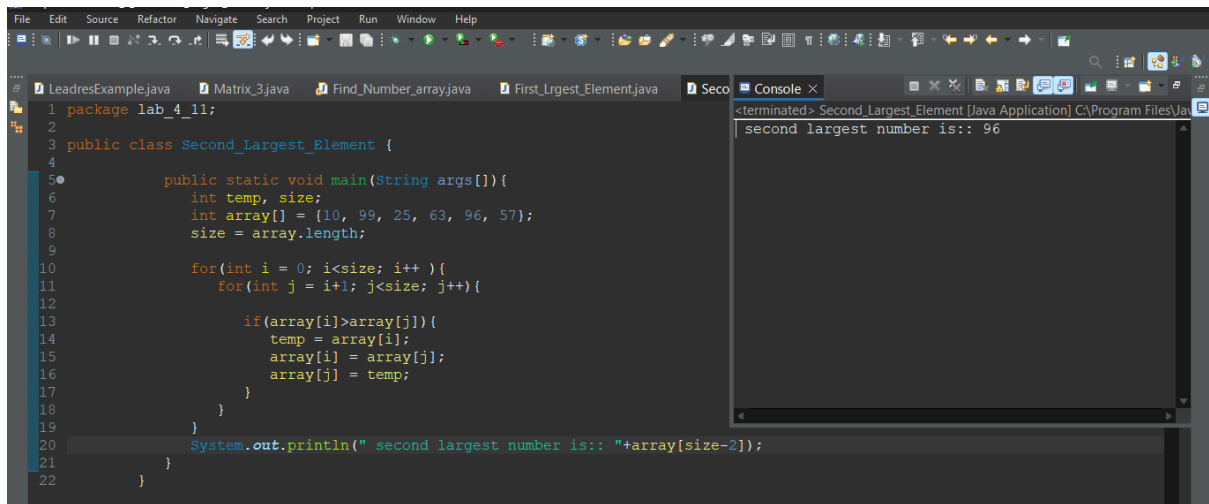


The screenshot shows the Eclipse IDE with a project named 'lab\_4\_11'. The package 'lab\_4\_11' contains four files: 'LeadresExample.java', 'Matrix\_3.java', 'Find\_Number\_array.java', and 'First\_Lrgest\_Element.java'. The 'First\_Lrgest\_Element.java' file contains the following code:

```
1 package lab_4_11;
2
3 public class First_Lrgest_Element {
4
5
6
7     public static void main(String[] args) {
8         int[] arr = {10, 5, 20, 25, 15, 30};
9
10        if (arr.length == 0) {
11            System.out.println("The array is empty."); return;}
12
13        int firstlargest = arr[0];
14
15        for (int i = 1; i < arr.length; i++) {
16            if (arr[i] > firstlargest) { firstlargest = arr[i];
17            }
18        }
19
20        System.out.println("The first largest element in the array is: " + firstlargest);
21    }
22 }
23
24 }
```

The console output shows the result of the program execution: "The first largest element in the array is: 30".

5. Write a Java program to find the second largest element in an array.



The image shows a screenshot of an IDE with a Java file named `Second_Largest_Element.java` and a console window. The code implements a bubble sort algorithm to find the second largest element in an array.

```
1 package lab_4_11;
2
3 public class Second_Largest_Element {
4
5     public static void main(String args[]){
6         int temp, size;
7         int array[] = {10, 99, 25, 63, 96, 57};
8         size = array.length;
9
10        for(int i = 0; i<size; i++){
11            for(int j = i+1; j<size; j++){
12
13                if(array[i]>array[j]){
14                    temp = array[i];
15                    array[i] = array[j];
16                    array[j] = temp;
17                }
18            }
19        }
20        System.out.println(" second largest number is:: "+array[size-2]);
21    }
22 }
```

The console window shows the output of the program:

```
<terminated> Second_Largest_Element [Java Application] C:\Program Files\Java\
second largest number is:: 96
```