

A dark blue vertical bar on the left side of the page. A blue arrow points to the right from the bar, containing the date.

10/2/2024

DSA Lab 06

FAHEEM AKBAR

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and sweep upwards and to the right.

CMS ID: 023-23-0365

DSA LAB 06 SOLUTION || SUBMITTED TO MA'AM MARINA

1. Implement selection sort and insertion sort.

```
1 import java.util.Arrays;
2
3 class InsertionSort{
4     void insertionSort(int array[]){
5
6         int size =array.length;
7
8         for(int i=1;i<array.length;i++){
9             int temp=array[i];
10            int j=i-1;
11
12            while (j>=0 && temp<array[j]) {
13                array[j+1]=array[j];
14                j--;
15            }
16            array[j+1]=temp;
17        }
18    }
19
20    public static void main(String args[]){
21
22        int [] data={9,5,4,1,3};
23
24        InsertionSort sort=new InsertionSort();
25        sort.insertionSort(data);
26
27        System.out.println("Sorting array in assecending a
28        System.out.println(Arrays.toString(data));
29    }
30 }
```

Opening Java Projects: check details

Couldn't start client Java Language Server

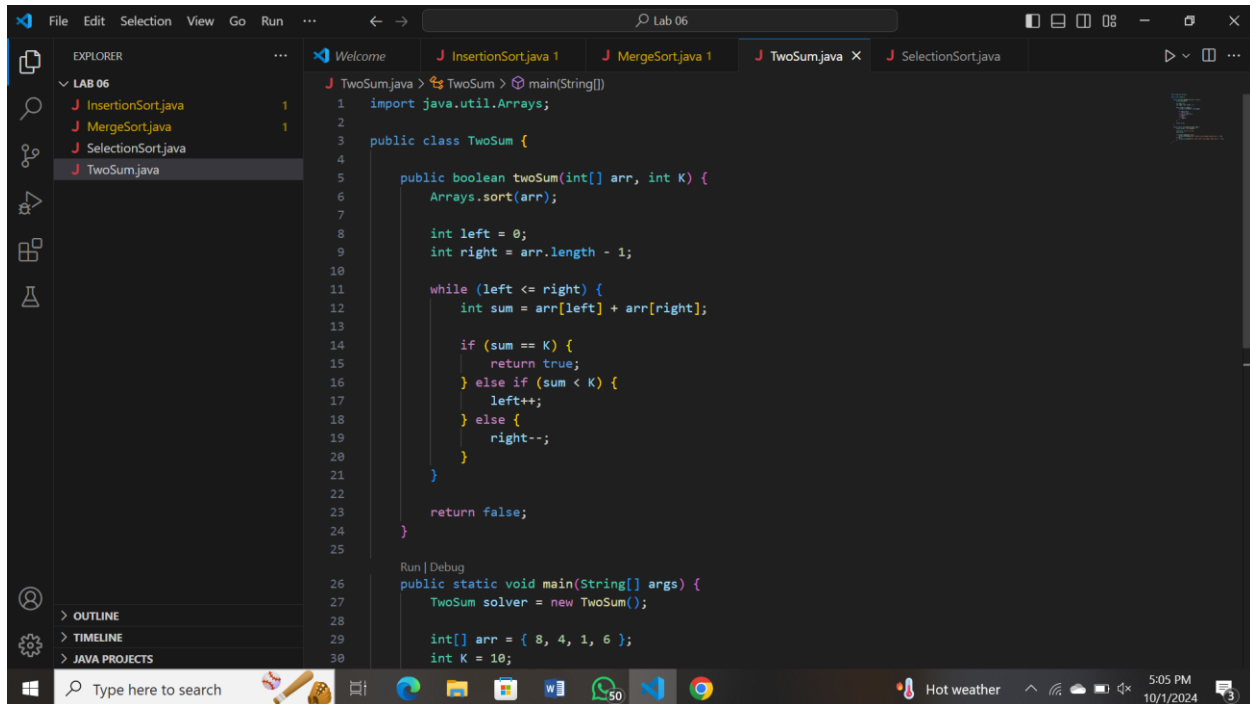
```
3 class InsertionSort{
4     void insertionSort(int array[]){
5
6         int size =array.length;
7
8         for(int i=1;i<array.length;i++){
9             int temp=array[i];
10            int j=i-1;
11
12            while (j>=0 && temp<array[j]) {
13                array[j+1]=array[j];
14                j--;
15            }
16            array[j+1]=temp;
17        }
18    }
19
20    public static void main(String args[]){
21
22        int [] data={9,5,4,1,3};
23
24        InsertionSort sort=new InsertionSort();
25        sort.insertionSort(data);
26
27        System.out.println(x:"Sorting array in assecending array :");
28        System.out.println(Arrays.toString(data));
29    }
30 }
```

Run | Debug

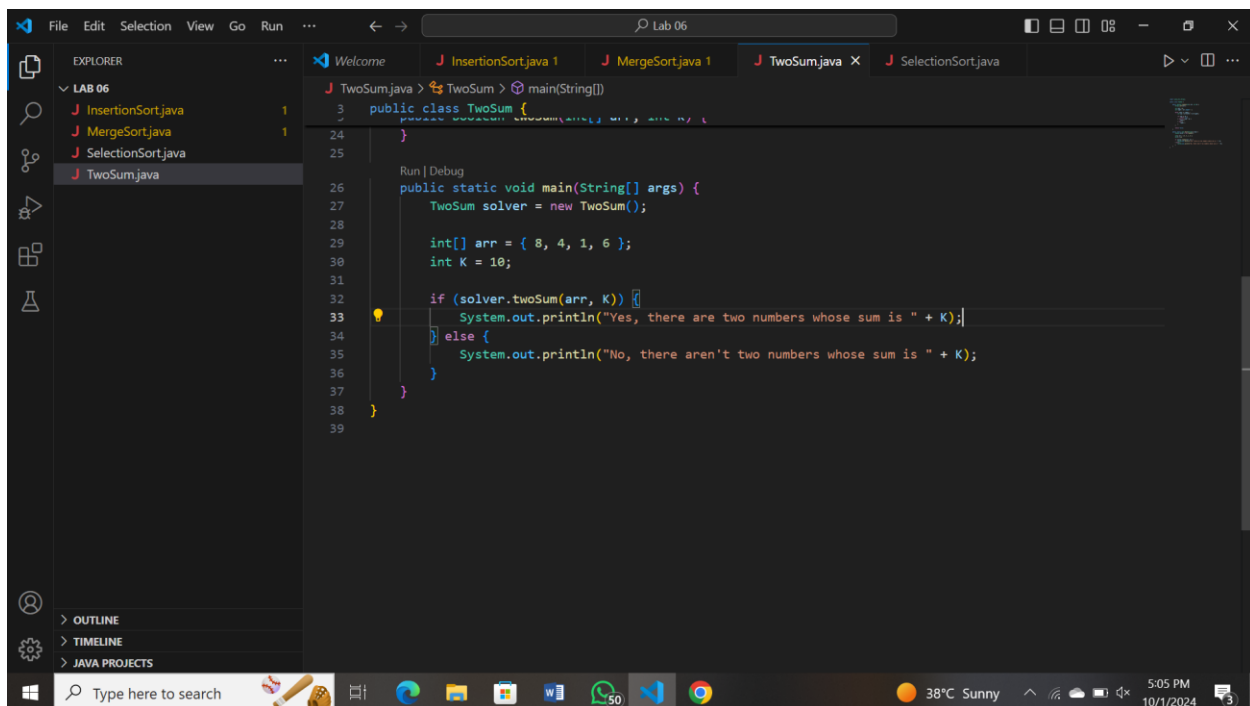
Sorting array in assecending array :
[1, 3, 4, 5, 9]

PS C:\Faheem Study\BSCS Semester 3\DSA\DSA LAB\Lab 06> & 'C:\Program Files\Java\jdk-22\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\PHLS\AppData\Roaming\Code\User\workspaceStorage\6bc14cf7e79ad27e39bb90c2a02b3b66\redhat_java\jdt_ws\Lab_06_67aaca79\bin' 'InsertionSort'

4. (Solve in $N \log N$): We are given an array that contains N numbers. We want to determine if there are two numbers whose sum equals a given number K . For instance, if the input is 8, 4, 1, and 6, and K is 10, then the answer is yes (4 plus 6 is 10). A number n may appear more than once in the input array; in that case and only in that case the sum may have the form $n + n$.

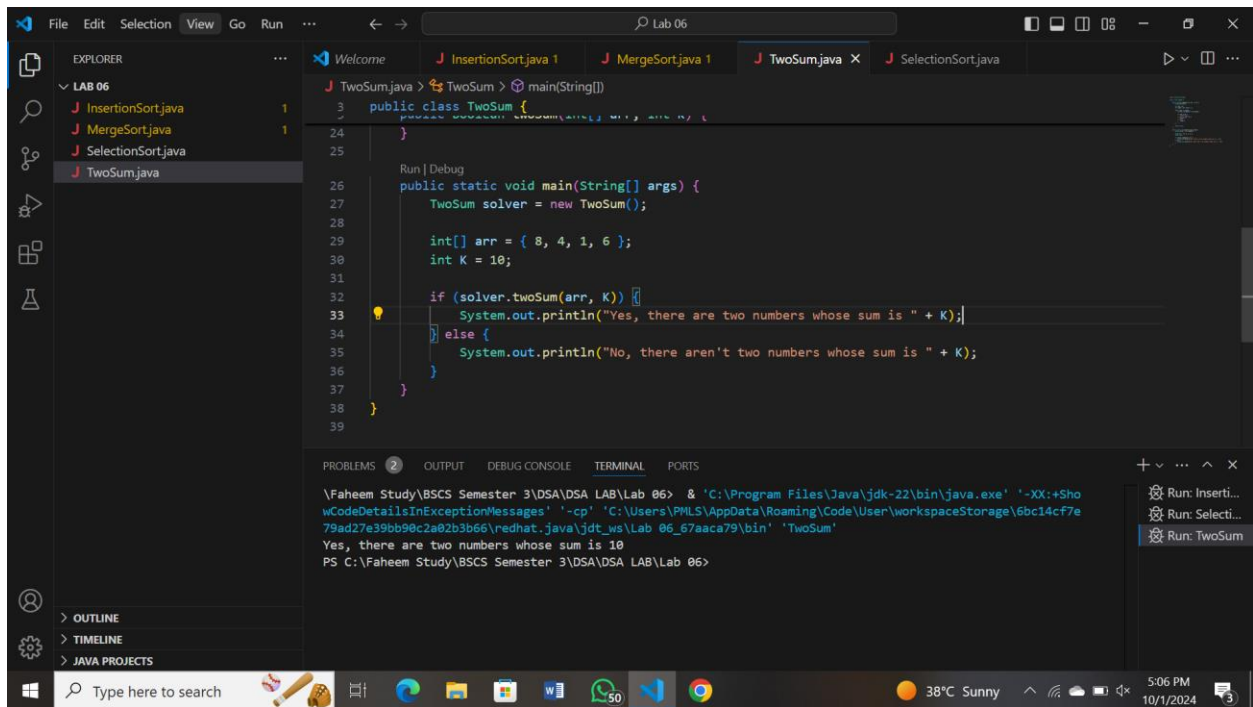


```
1 import java.util.Arrays;
2
3 public class TwoSum {
4
5     public boolean twoSum(int[] arr, int K) {
6         Arrays.sort(arr);
7
8         int left = 0;
9         int right = arr.length - 1;
10
11         while (left <= right) {
12             int sum = arr[left] + arr[right];
13
14             if (sum == K) {
15                 return true;
16             } else if (sum < K) {
17                 left++;
18             } else {
19                 right--;
20             }
21         }
22
23         return false;
24     }
25
26     Run | Debug
27     public static void main(String[] args) {
28         TwoSum solver = new TwoSum();
29
30         int[] arr = { 8, 4, 1, 6 };
31         int K = 10;
```



```
24     }
25
26     Run | Debug
27     public static void main(String[] args) {
28         TwoSum solver = new TwoSum();
29
30         int[] arr = { 8, 4, 1, 6 };
31         int K = 10;
32
33         if (solver.twoSum(arr, K)) {
34             System.out.println("Yes, there are two numbers whose sum is " + K);
35         } else {
36             System.out.println("No, there aren't two numbers whose sum is " + K);
37         }
38     }
39 }
```

Task 4 Output:



The screenshot displays an IDE window titled "Lab 06" with several tabs: "Welcome", "InsertionSort.java 1", "MergeSort.java 1", "TwoSum.java X", and "SelectionSort.java". The "TwoSum.java" tab is active, showing the following code:

```
1 public class TwoSum {
2     public static void main(String[] args) {
3         TwoSum solver = new TwoSum();
4
5         int[] arr = { 8, 4, 1, 6 };
6         int K = 10;
7
8         if (solver.twoSum(arr, K)) {
9             System.out.println("Yes, there are two numbers whose sum is " + K);
10        } else {
11            System.out.println("No, there aren't two numbers whose sum is " + K);
12        }
13    }
14 }
```

The code is executed, and the output is shown in the "TERMINAL" pane at the bottom. The output indicates that the program successfully found two numbers whose sum is 10.

```
\Faheem Study\BSCS Semester 3\DSA LAB\Lab 06> & 'C:\Program Files\Java\jdk-22\bin\java.exe' '-XX:+Sho
wCodeDetailsInExceptionMessages' '-cp' 'C:\Users\PMLS\AppData\Roaming\Code\User\workspaceStorage\6bc14cf7e
79ad27e39bb98c2a02b3b66\redhat.java\jdt_ws\Lab 06_67aaca79\bin' 'TwoSum'
Yes, there are two numbers whose sum is 10
PS C:\Faheem Study\BSCS Semester 3\DSA LAB\Lab 06>
```

The Windows taskbar at the bottom shows the system clock as 5:06 PM on 10/1/2024, with a weather forecast of 38°C Sunny.