



---

# DSA LAB NO 6

---

ASSIGNMENT

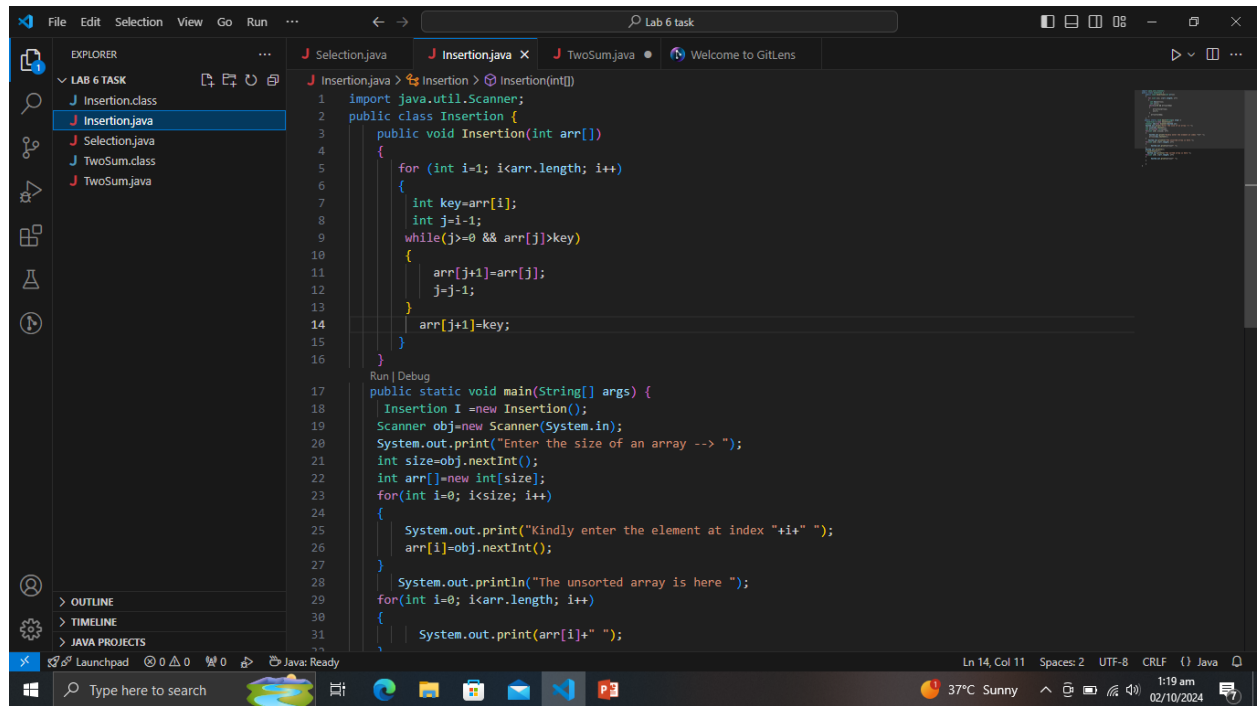


UZAIR ALI S/O YAR MUHAMMAD MEMON

BS-CS-III-SEC-A  
0023-23-0350

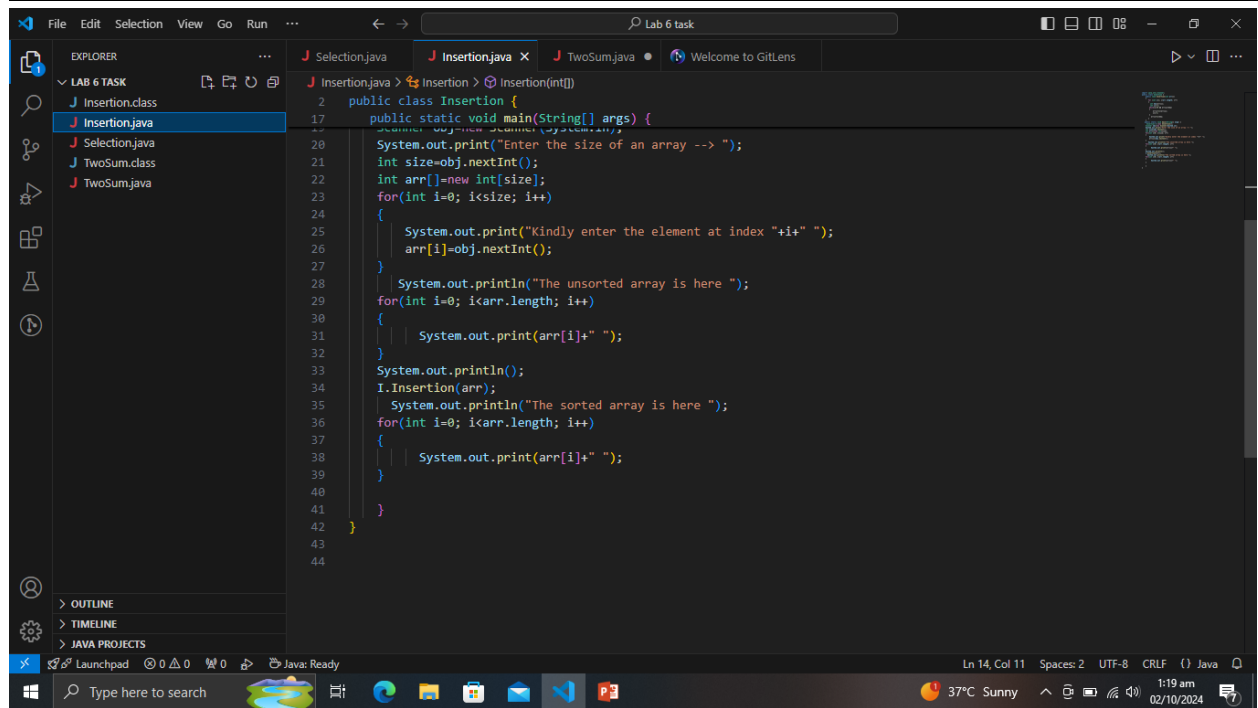
# 1. Implement insertion sort.

## Code



This screenshot shows the implementation of the `Insertion` method in `Insertion.java`. The method `Insertion(int[] arr)` takes an array of integers and sorts it using the insertion sort algorithm. It iterates through the array starting from the second element, and for each element, it finds its correct position in the sorted subarray by shifting elements to the right. The `main` method prompts the user to enter the size of the array and then the elements, displaying the unsorted array before sorting.

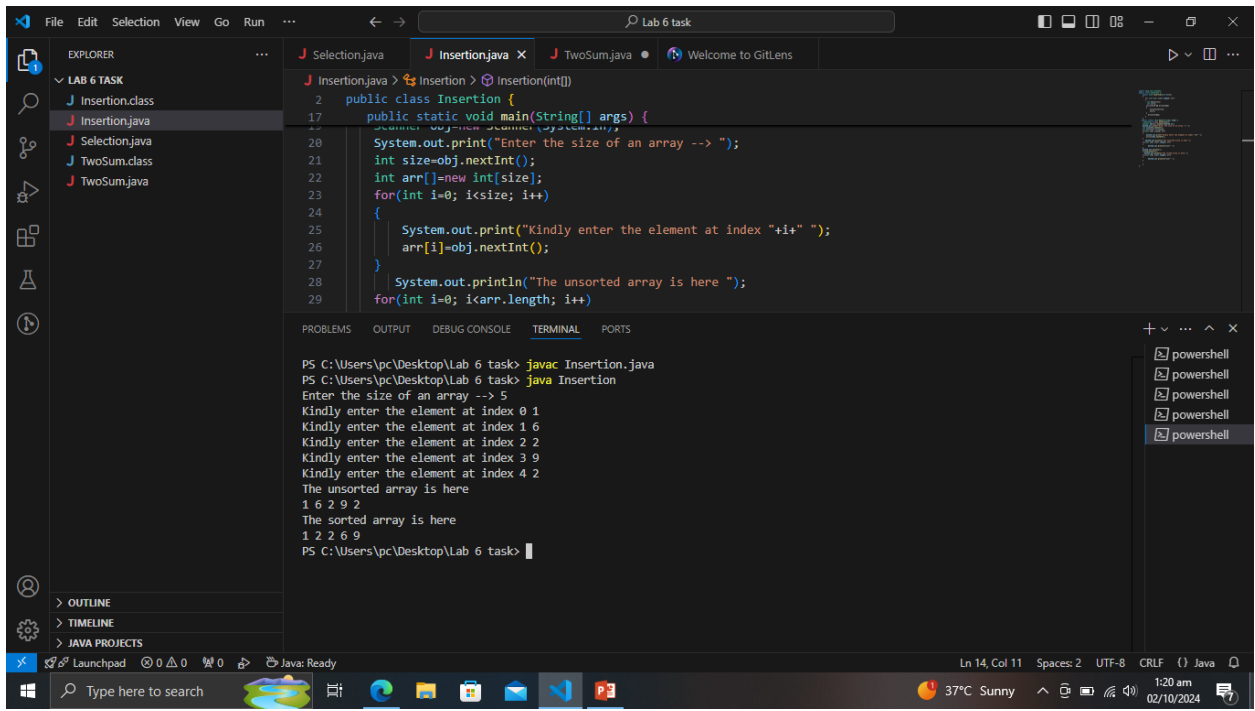
```
1 import java.util.Scanner;
2 public class Insertion {
3     public void Insertion(int arr[])
4     {
5         for (int i=1; i<arr.length; i++)
6         {
7             int key=arr[i];
8             int j=i-1;
9             while(j>=0 && arr[j]>key)
10            {
11                arr[j+1]=arr[j];
12                j=j-1;
13            }
14            arr[j+1]=key;
15        }
16    }
17    Run | Debug
18    public static void main(String[] args) {
19        Insertion I =new Insertion();
20        Scanner obj=new Scanner(System.in);
21        System.out.print("Enter the size of an array --> ");
22        int size=obj.nextInt();
23        int arr[]=new int[size];
24        for(int i=0; i<size; i++)
25        {
26            System.out.print("Kindly enter the element at index "+i+" ");
27            arr[i]=obj.nextInt();
28        }
29        System.out.println("The unsorted array is here ");
30        for(int i=0; i<arr.length; i++)
31        {
32            System.out.print(arr[i]+" ");
33        }
34    }
35 }
```



This screenshot shows the completion of the `Insertion` sort implementation. The `main` method now includes a call to `I.Insertion(arr);` to sort the array. After sorting, it displays the sorted array by iterating through it and printing each element.

```
17 public static void main(String[] args) {
18     Scanner obj=new Scanner(System.in);
19     System.out.print("Enter the size of an array --> ");
20     int size=obj.nextInt();
21     int arr[]=new int[size];
22     for(int i=0; i<size; i++)
23     {
24         System.out.print("Kindly enter the element at index "+i+" ");
25         arr[i]=obj.nextInt();
26     }
27     System.out.println("The unsorted array is here ");
28     for(int i=0; i<arr.length; i++)
29     {
30         System.out.print(arr[i]+" ");
31     }
32     System.out.println();
33     I.Insertion(arr);
34     System.out.println("The sorted array is here ");
35     for(int i=0; i<arr.length; i++)
36     {
37         System.out.print(arr[i]+" ");
38     }
39 }
40 }
41 }
42 }
43 }
44 }
```

# Output



The screenshot displays an IDE window titled "Lab 6 task". The Explorer panel on the left shows a project named "LAB 6 TASK" containing files: Insertion.class, Insertion.java, Selection.java, TwoSum.class, and TwoSum.java. The main editor shows the code for Insertion.java, which implements an insertion sort algorithm. The code prompts the user to enter the size of an array, followed by elements at specific indices, and then prints the unsorted and sorted arrays.

```
1 public class Insertion {
2     public static void main(String[] args) {
3         Scanner obj=new Scanner(System.in);
4         System.out.print("Enter the size of an array --> ");
5         int size=obj.nextInt();
6         int arr[]=new int[size];
7         for(int i=0; i<size; i++)
8         {
9             System.out.print("Kindly enter the element at index "+i+" ");
10            arr[i]=obj.nextInt();
11        }
12        System.out.println("The unsorted array is here ");
13        for(int i=0; i<arr.length; i++)
```

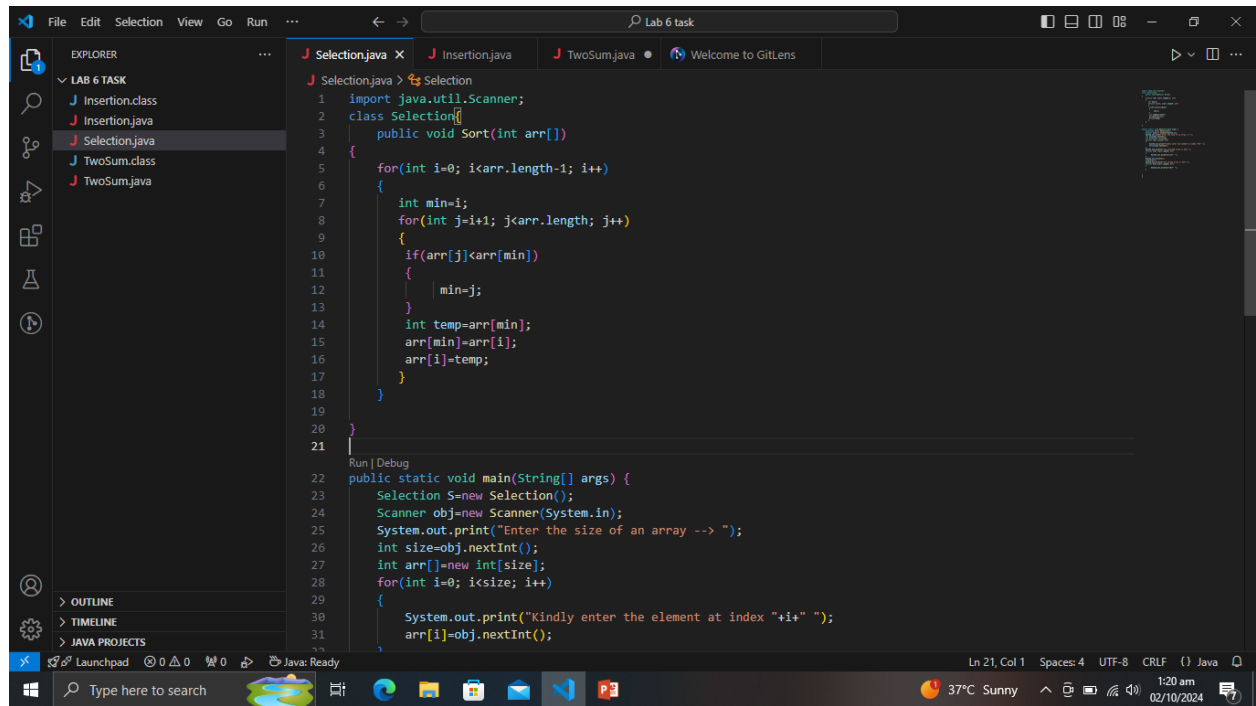
The Terminal panel at the bottom shows the execution of the program. It displays the commands to compile and run the Java code, followed by the user's input and the program's output.

```
PS C:\Users\pc\Desktop\Lab 6 task> javac Insertion.java
PS C:\Users\pc\Desktop\Lab 6 task> java Insertion
Enter the size of an array --> 5
Kindly enter the element at index 0 1
Kindly enter the element at index 1 6
Kindly enter the element at index 2 2
Kindly enter the element at index 3 9
Kindly enter the element at index 4 2
The unsorted array is here
1 6 2 9 2
The sorted array is here
1 2 2 6 9
PS C:\Users\pc\Desktop\Lab 6 task>
```

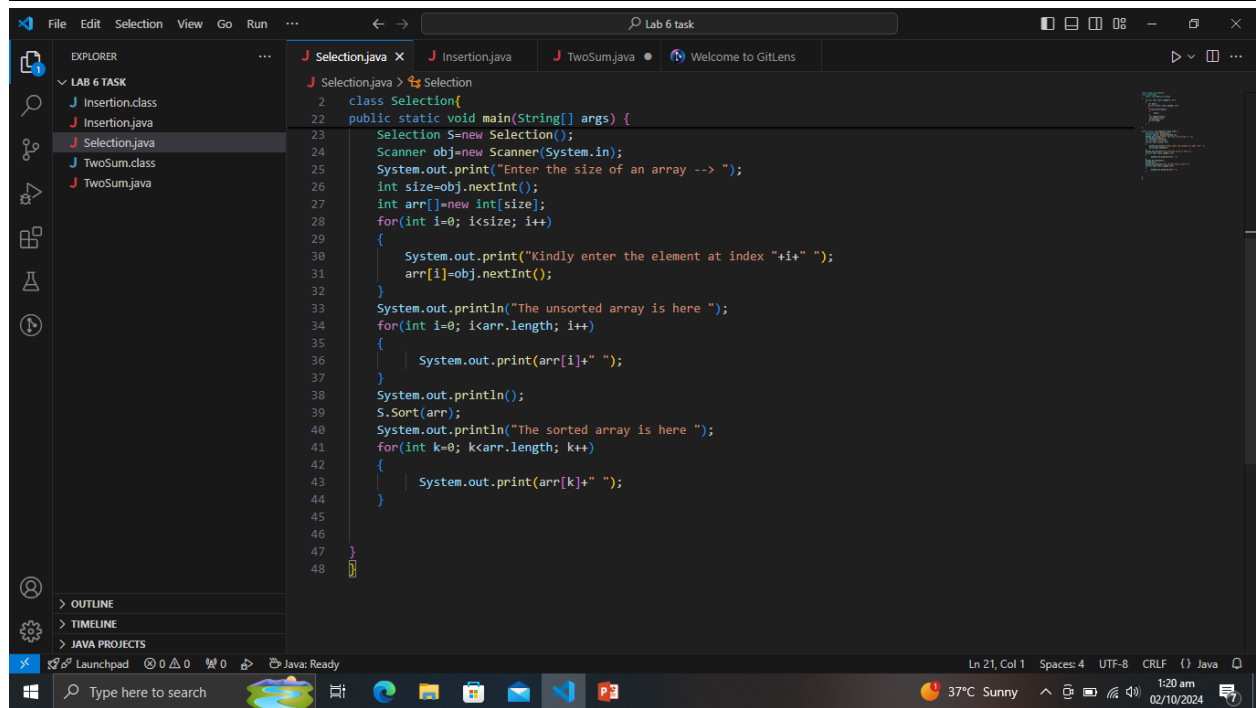
The status bar at the bottom indicates the current cursor position (Ln 14, Col 11), encoding (UTF-8), and line endings (CRLF). The system tray shows the date and time as 1:20 am on 02/10/2024.

# 1. Implement Selection Sort.

## Code

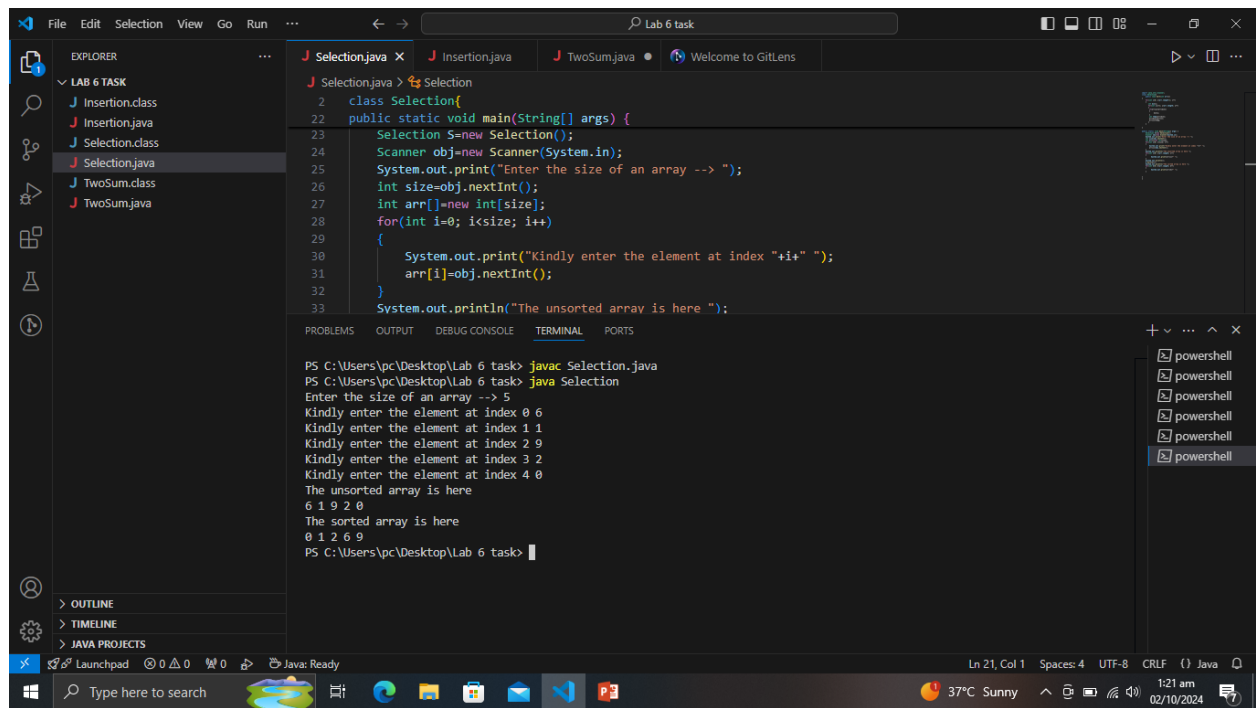


```
1 import java.util.Scanner;
2 class Selection{
3     public void Sort(int arr[])
4     {
5         for(int i=0; i<arr.length-1; i++)
6         {
7             int min=i;
8             for(int j=i+1; j<arr.length; j++)
9             {
10                 if(arr[j]<arr[min])
11                 {
12                     min=j;
13                 }
14                 int temp=arr[min];
15                 arr[min]=arr[i];
16                 arr[i]=temp;
17             }
18         }
19     }
20 }
21
22 Run | Debug
23 public static void main(String[] args) {
24     Selection S=new Selection();
25     Scanner obj=new Scanner(System.in);
26     System.out.print("Enter the size of an array --> ");
27     int size=obj.nextInt();
28     int arr[]=new int[size];
29     for(int i=0; i<size; i++)
30     {
31         System.out.print("Kindly enter the element at index "+i+" ");
32         arr[i]=obj.nextInt();
33     }
34 }
```



```
2 class Selection{
22     public static void main(String[] args) {
23         Selection S=new Selection();
24         Scanner obj=new Scanner(System.in);
25         System.out.print("Enter the size of an array --> ");
26         int size=obj.nextInt();
27         int arr[]=new int[size];
28         for(int i=0; i<size; i++)
29         {
30             System.out.print("Kindly enter the element at index "+i+" ");
31             arr[i]=obj.nextInt();
32         }
33         System.out.println("The unsorted array is here ");
34         for(int i=0; i<arr.length; i++)
35         {
36             System.out.print(arr[i]+" ");
37         }
38         System.out.println();
39         S.Sort(arr);
40         System.out.println("The sorted array is here ");
41         for(int k=0; k<arr.length; k++)
42         {
43             System.out.print(arr[k]+" ");
44         }
45     }
46 }
47
48 }
```

## Output



The screenshot shows an IDE window titled "Lab 6 task". The Explorer panel on the left lists files under "LAB 6 TASK": Insertion.class, Insertion.java, Selection.class, Selection.java (selected), TwoSum.class, and TwoSum.java. The main editor displays the code for Selection.java:

```
2 class Selection{
22 public static void main(String[] args) {
23     Selection S=new Selection();
24     Scanner obj=new Scanner(System.in);
25     System.out.print("Enter the size of an array --> ");
26     int size=obj.nextInt();
27     int arr[]=new int[size];
28     for(int i=0; i<size; i++)
29     {
30         System.out.print("Kindly enter the element at index "+i+" ");
31         arr[i]=obj.nextInt();
32     }
33     System.out.println("The unsorted array is here ");
```

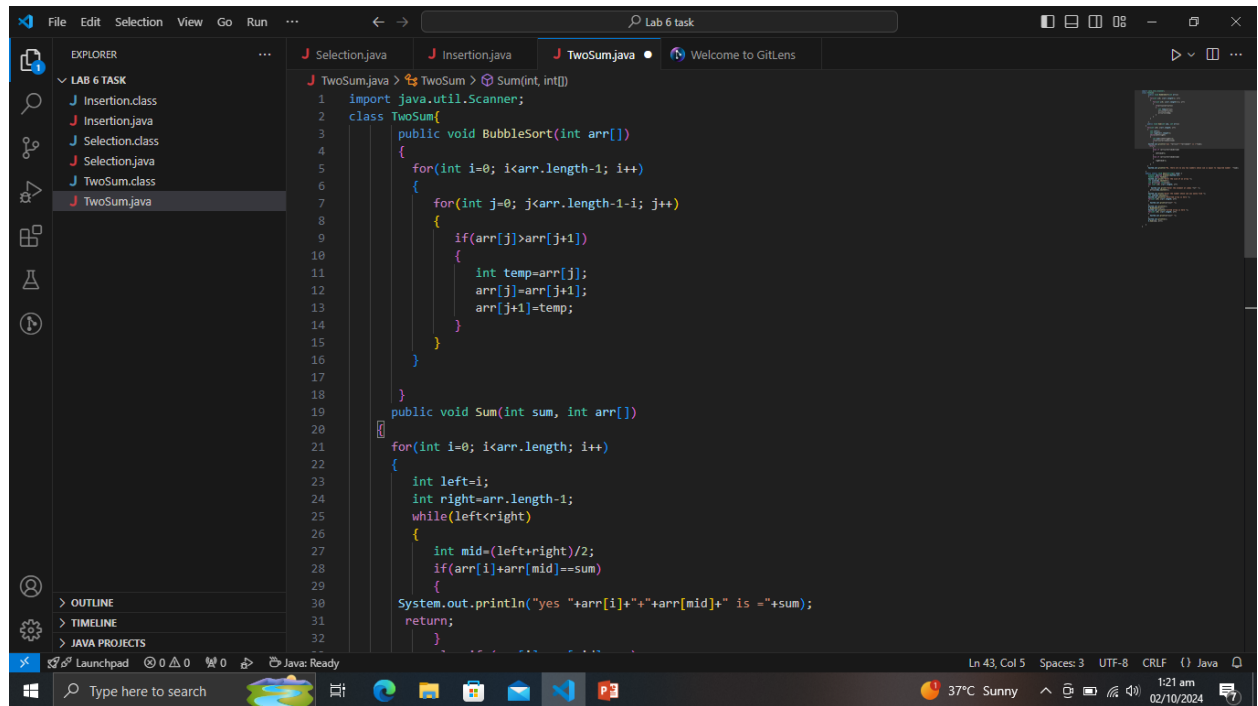
The bottom panel shows the output of the program:

```
PS C:\Users\pc\Desktop\Lab 6 task> javac Selection.java
PS C:\Users\pc\Desktop\Lab 6 task> java Selection
Enter the size of an array --> 5
Kindly enter the element at index 0 6
Kindly enter the element at index 1 1
Kindly enter the element at index 2 9
Kindly enter the element at index 3 2
Kindly enter the element at index 4 0
The unsorted array is here
6 1 9 2 0
The sorted array is here
0 1 2 6 9
PS C:\Users\pc\Desktop\Lab 6 task>
```

The right side of the bottom panel shows a list of open PowerShell windows.

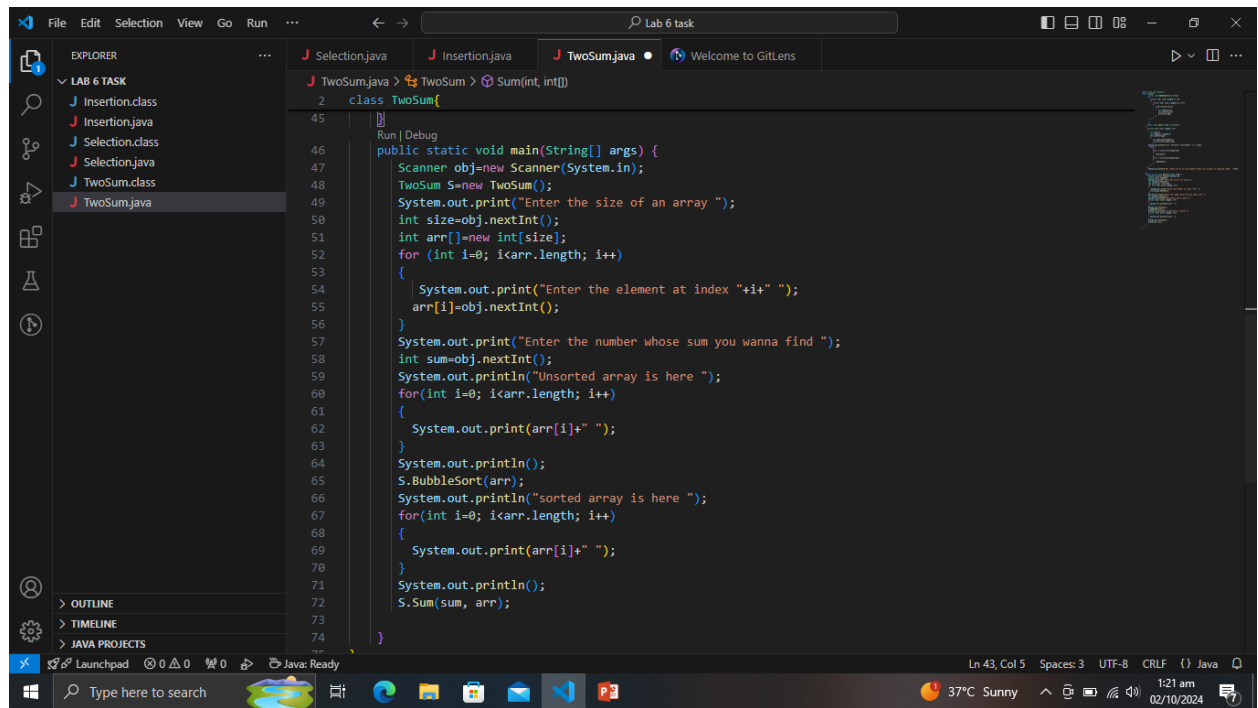
**4. (Solve in NlogN): We are given an array that contains N numbers. We want to determine if there are two numbers whose sum equals a given number K. For instance, if the input is 8, 4, 1, and 6, and K is 10, then the answer is yes (4 plus 6 is 10). A number n may appear more than once in the input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

# Code



This screenshot shows the first part of the Java code in the TwoSum.java file. The Explorer panel on the left lists the files in the 'LAB 6 TASK' project: Insertion.class, Insertion.java, Selection.class, Selection.java, TwoSum.class, and TwoSum.java. The main editor displays the following code:

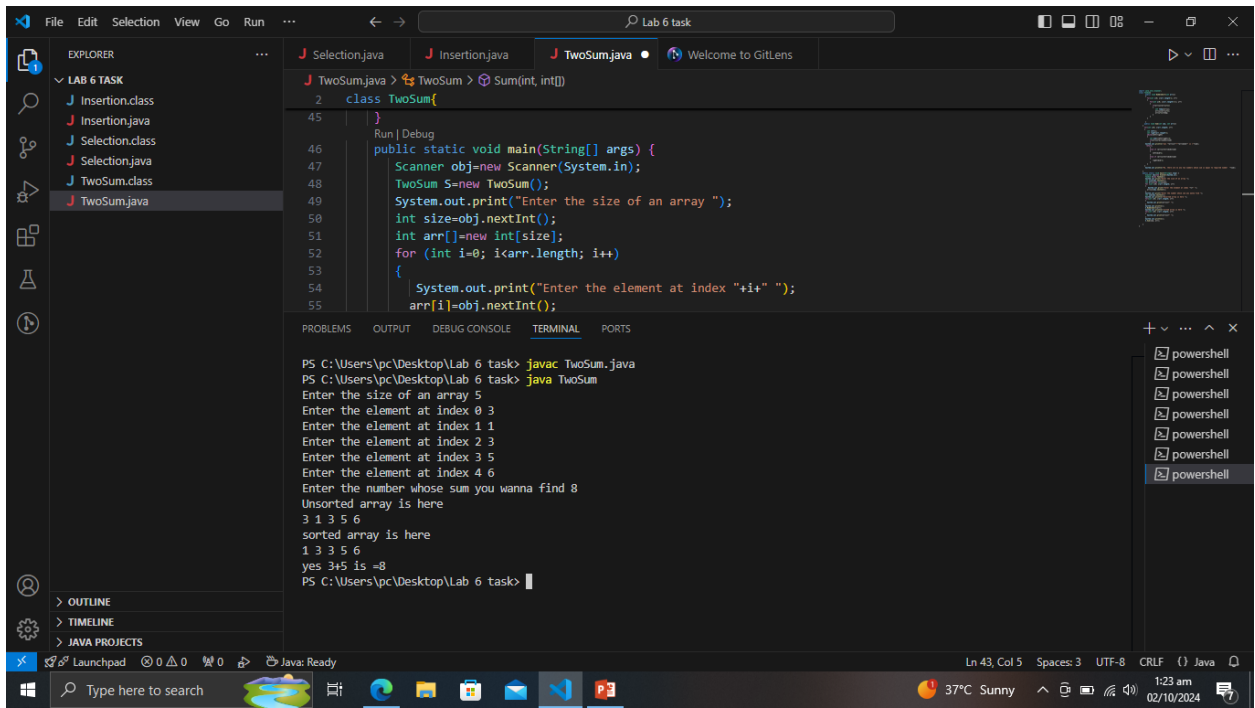
```
1 import java.util.Scanner;
2 class TwoSum{
3     public void BubbleSort(int arr[])
4     {
5         for(int i=0; i<arr.length-1; i++)
6         {
7             for(int j=0; j<arr.length-1-i; j++)
8             {
9                 if(arr[j]>arr[j+1])
10                {
11                    int temp=arr[j];
12                    arr[j]=arr[j+1];
13                    arr[j+1]=temp;
14                }
15            }
16        }
17    }
18
19    public void Sum(int sum, int arr[])
20    {
21        for(int i=0; i<arr.length; i++)
22        {
23            int left=i;
24            int right=arr.length-1;
25            while(left<right)
26            {
27                int mid=(left+right)/2;
28                if(arr[i]+arr[mid]==sum)
29                {
30                    System.out.println("yes "+arr[i]+"+"+arr[mid]+" is =" +sum);
31                    return;
32                }
33            }
34        }
35    }
36 }
```



This screenshot shows the second part of the Java code in the TwoSum.java file, including the main method. The Explorer panel on the left is the same as in the first screenshot. The main editor displays the following code:

```
45 }
46 public static void main(String[] args) {
47     Scanner obj=new Scanner(System.in);
48     TwoSum S=new TwoSum();
49     System.out.print("Enter the size of an array ");
50     int size=obj.nextInt();
51     int arr[]=new int[size];
52     for (int i=0; i<arr.length; i++)
53     {
54         System.out.print("Enter the element at index "+i+" ");
55         arr[i]=obj.nextInt();
56     }
57     System.out.print("Enter the number whose sum you wanna find ");
58     int sum=obj.nextInt();
59     System.out.println("Unsorted array is here ");
60     for(int i=0; i<arr.length; i++)
61     {
62         System.out.print(arr[i]+" ");
63     }
64     System.out.println();
65     S.BubbleSort(arr);
66     System.out.println("sorted array is here ");
67     for(int i=0; i<arr.length; i++)
68     {
69         System.out.print(arr[i]+" ");
70     }
71     System.out.println();
72     S.Sum(sum, arr);
73 }
74 }
```

## Output.



The screenshot displays an IDE window titled "Lab 6 task". The Explorer panel on the left shows a project named "LAB 6 TASK" containing files: Insertion.class, Insertion.java, Selection.class, Selection.java, TwoSum.class, and TwoSum.java. The main editor shows the code for TwoSum.java, which includes a class TwoSum with a static method main. The main method uses a Scanner to take user input for array size and elements, then prints the sorted array and its sum. The bottom panel shows the output of the program, which matches the expected results from the problem statement. The terminal shows the commands used to compile and run the program.

```
class TwoSum {
    public static void main(String[] args) {
        Scanner obj = new Scanner(System.in);
        TwoSum S = new TwoSum();
        System.out.print("Enter the size of an array ");
        int size = obj.nextInt();
        int arr[] = new int[size];
        for (int i = 0; i < arr.length; i++)
        {
            System.out.print("Enter the element at index "+i+" ");
            arr[i] = obj.nextInt();
        }
    }
}
```

```
PS C:\Users\pc\Desktop\Lab 6 task> javac TwoSum.java
PS C:\Users\pc\Desktop\Lab 6 task> java TwoSum
Enter the size of an array 5
Enter the element at index 0 3
Enter the element at index 1 1
Enter the element at index 2 3
Enter the element at index 3 5
Enter the element at index 4 6
Enter the number whose sum you wanna find 8
Unsorted array is here
3 1 3 5 6
sorted array is here
1 3 3 5 6
yes 3+5 is =8
PS C:\Users\pc\Desktop\Lab 6 task>
```

The End