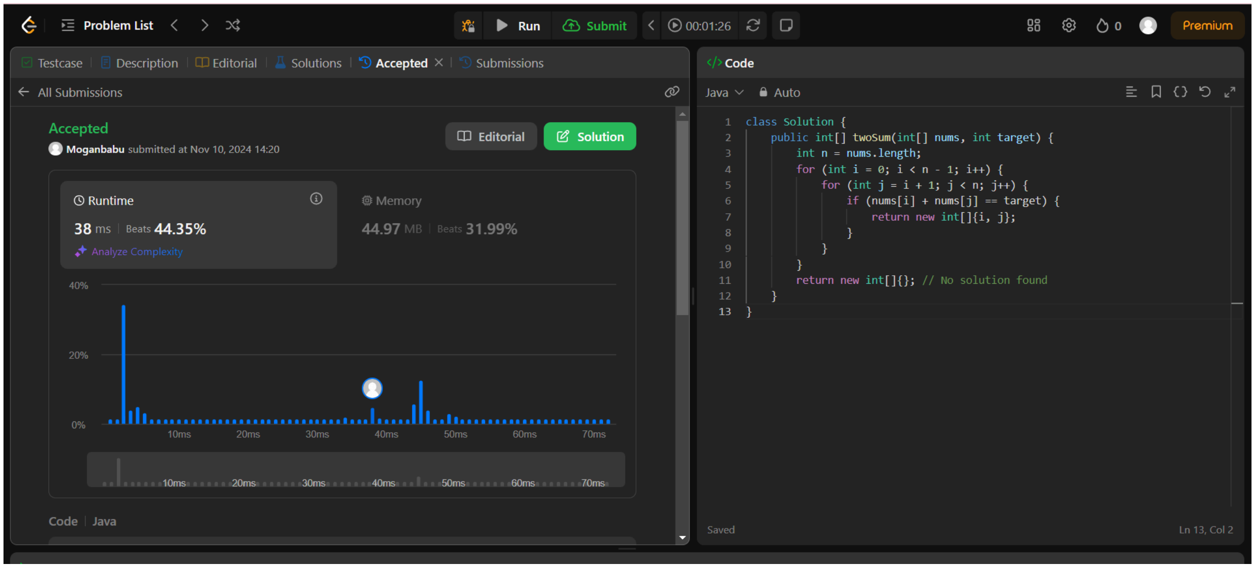
MOGAN BABU C
22CZ031

1.Two Sum



Accepted
Moganbabu submitted at Nov 10, 2024 14:20

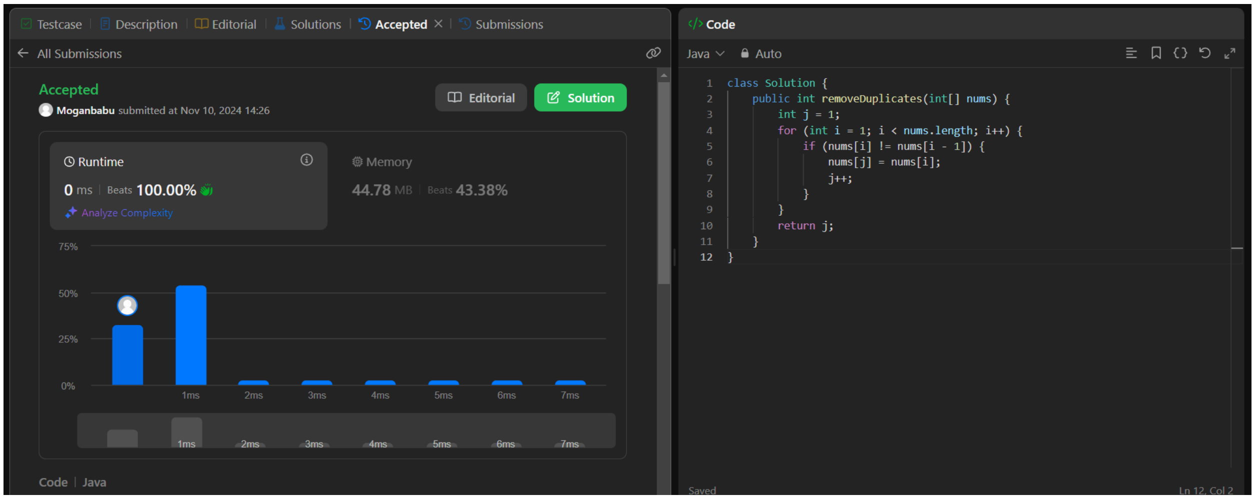
Runtime
38 ms | Beats 44.35%

Memory
44.97 MB | Beats 31.99%

Code | Java

```java
class Solution {
    public int[] twoSum(int[] nums, int target) {
        int n = nums.length;
        for (int i = 0; i < n - 1; i++) {
            for (int j = i + 1; j < n; j++) {
                if (nums[i] + nums[j] == target) {
                    return new int[]{i, j};
                }
            }
        }
        return new int[]{}; // No solution found
    }
}
```

2. Leetcode 26



Accepted
Moganbabu submitted at Nov 10, 2024 14:26

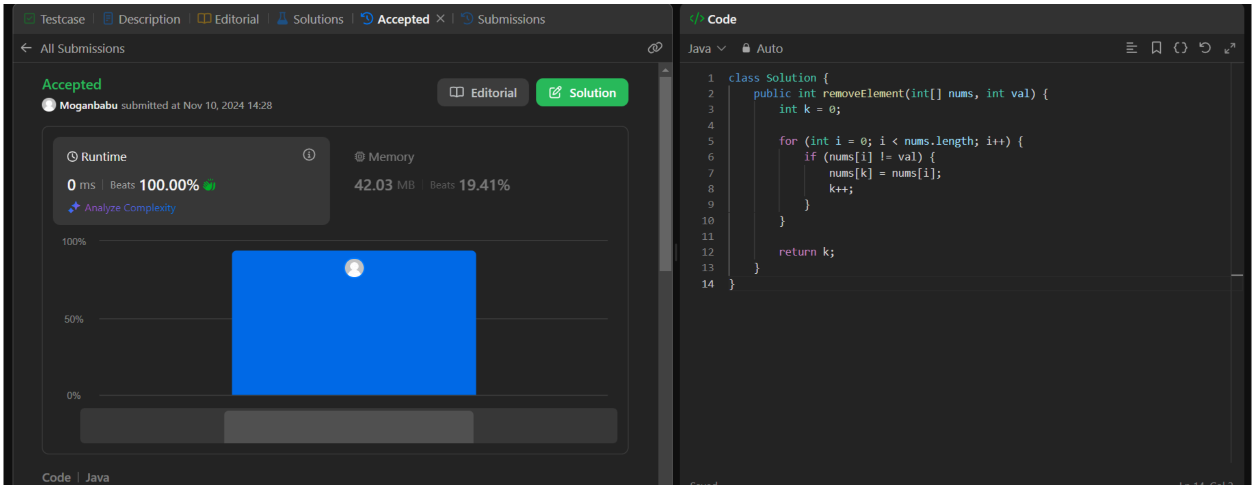
Runtime
0 ms | Beats 100.00%

Memory
44.78 MB | Beats 43.38%

Code | Java

```java
class Solution {
    public int removeDuplicates(int[] nums) {
        int j = 1;
        for (int i = 1; i < nums.length; i++) {
            if (nums[i] != nums[i - 1]) {
                nums[j] = nums[i];
                j++;
            }
        }
        return j;
    }
}
```

3. Leetcode 27



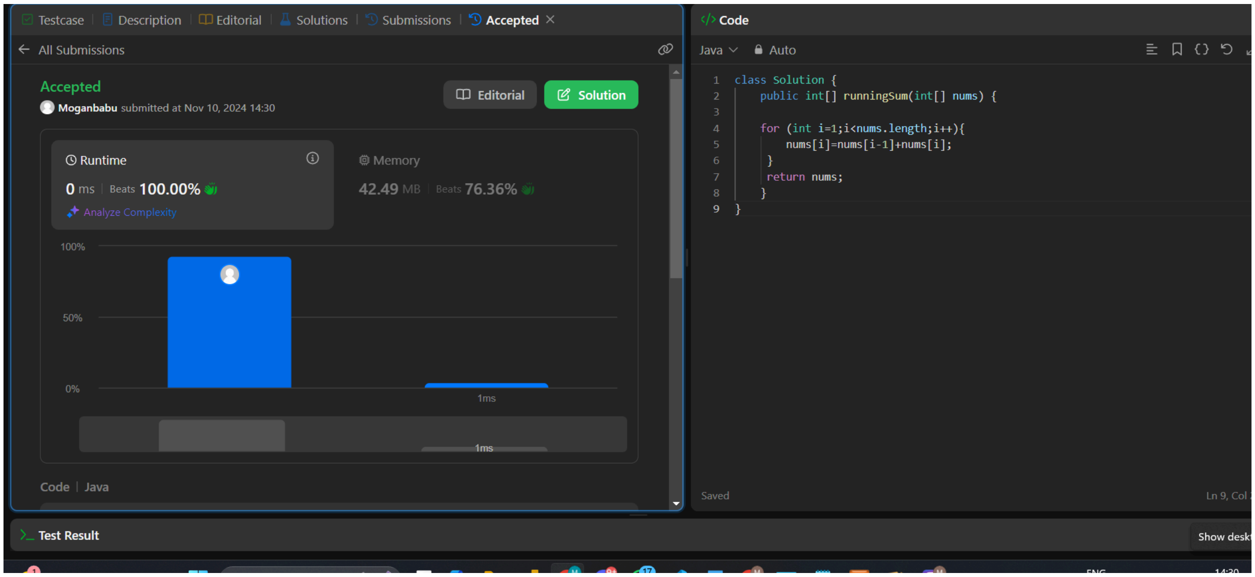
Accepted
Moganbabu submitted at Nov 10, 2024 14:28

Runtime
0 ms | Beats 100.00%

Memory
42.03 MB | Beats 19.41%

```java
class Solution {
    public int removeElement(int[] nums, int val) {
        int k = 0;

        for (int i = 0; i < nums.length; i++) {
            if (nums[i] != val) {
                nums[k] = nums[i];
                k++;
            }
        }

        return k;
    }
}
```

4.Leetcode 1480



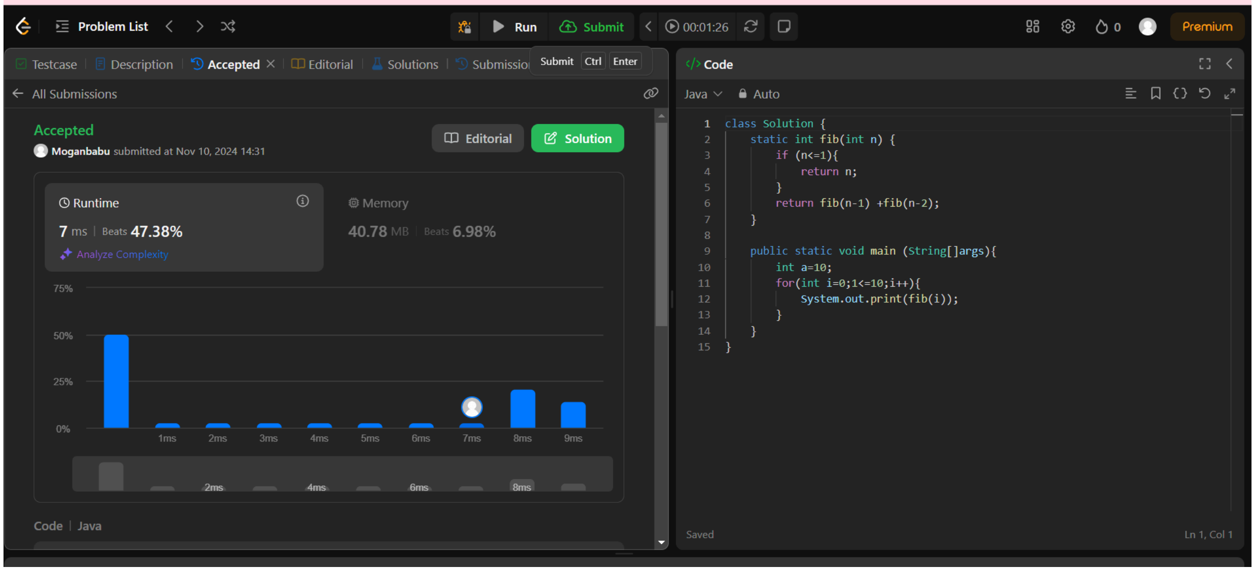
Accepted
Moganbabu submitted at Nov 10, 2024 14:30

Runtime
0 ms | Beats 100.00%

Memory
42.49 MB | Beats 76.36%

```java
class Solution {
    public int[] runningSum(int[] nums) {

    for (int i=1;i<nums.length;i++){
        nums[i]=nums[i-1]+nums[i];
     }
     return nums;
    }
}
```

5.Leetcode 509



Accepted

Moganbabu submitted at Nov 10, 2024 14:31

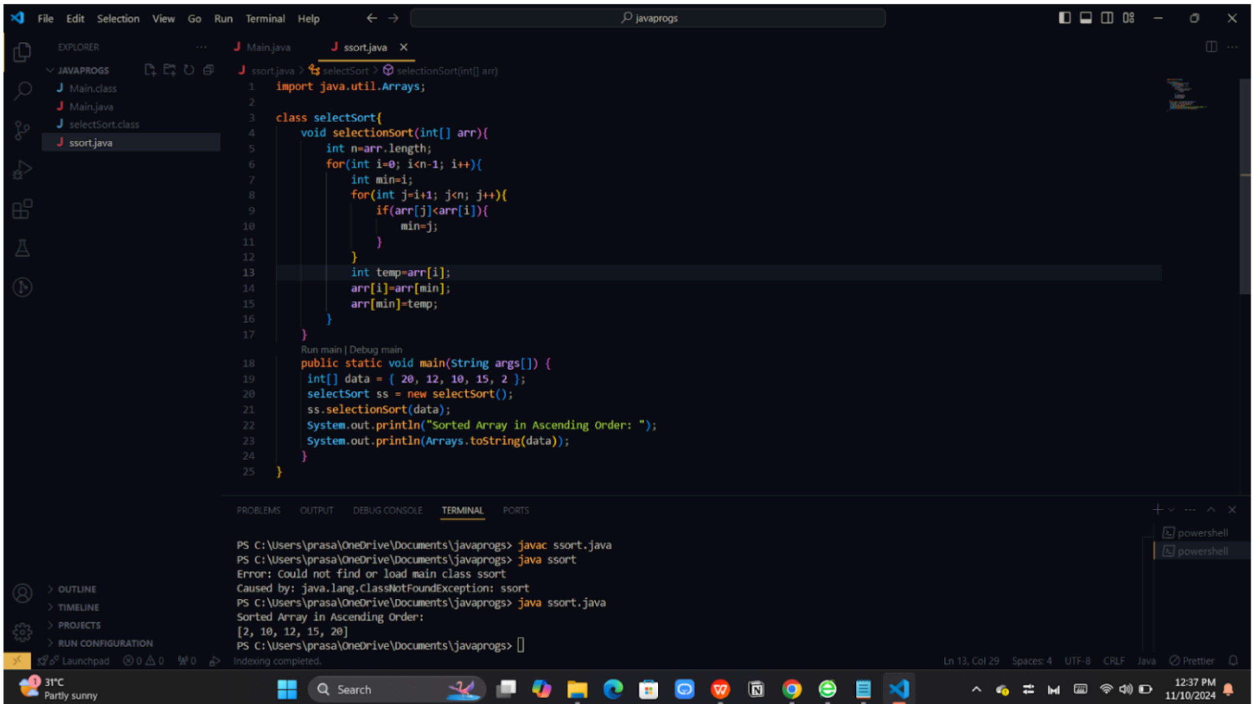
Runtime: 7 ms | Beats 47.38%

Analyze Complexity

Memory: 40.78 MB | Beats 6.98%

```java
class Solution {
    static int fib(int n) {
        if (n<=1){
            return n;
        }
        return fib(n-1) +fib(n-2);
    }

    public static void main (String[]args){
        int a=10;
        for(int i=0;1<=10;i++){
            System.out.print(fib(i));
        }
    }
}
```

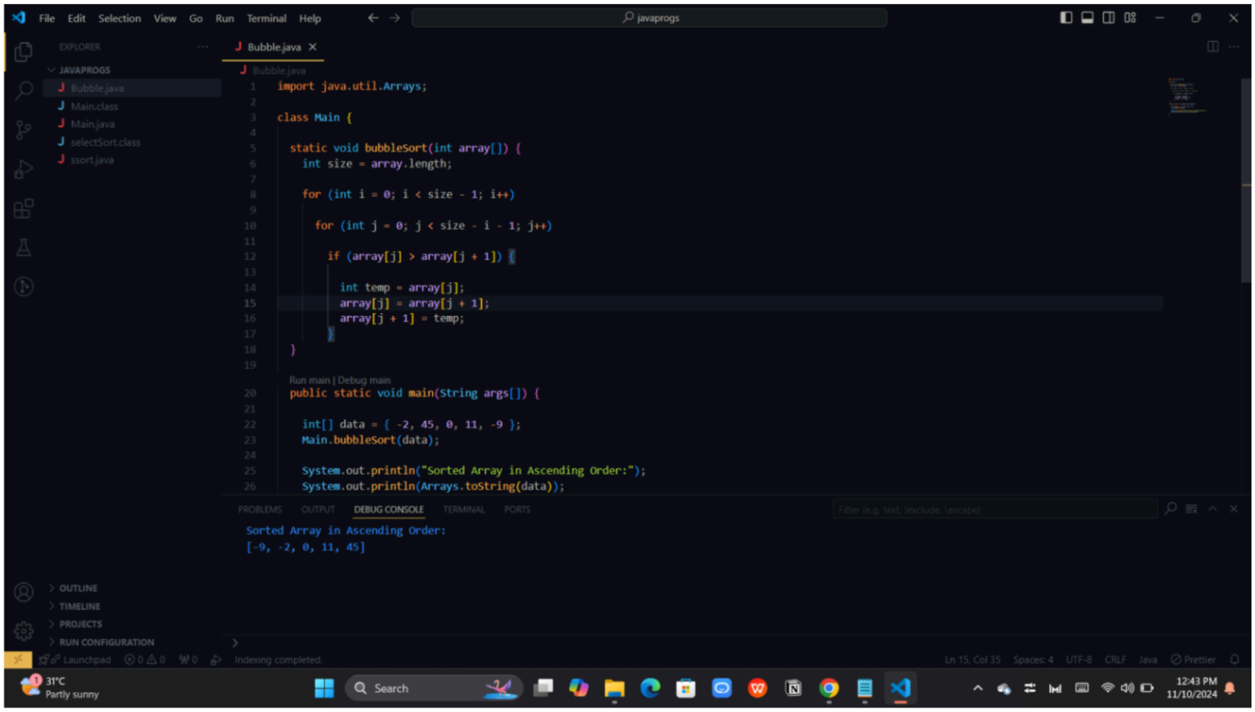
6.selection sort



```java
import java.util.Arrays;

class selectSort{
    void selectionSort(int[] arr){
        int n=arr.length;
        for(int i=0; i<n-1; i++){
            int min=i;
            for(int j=i+1; j<n; j++){
                if(arr[j]<arr[i]){
                    min=j;
                }
            }
            int temp=arr[i];
            arr[i]=arr[min];
            arr[min]=temp;
        }
    }
    public static void main(String args[]) {
     int[] data = { 20, 12, 10, 15, 2 };
     selectSort ss = new selectSort();
     ss.selectionSort(data);
     System.out.println("Sorted Array in Ascending Order: ");
     System.out.println(Arrays.toString(data));
    }
}
```

```
PS C:\Users\prasa\OneDrive\Documents\javaprogs> javac ssort.java
PS C:\Users\prasa\OneDrive\Documents\javaprogs> java ssort
Error: Could not find or load main class ssort
Caused by: java.lang.ClassNotFoundException: ssort
PS C:\Users\prasa\OneDrive\Documents\javaprogs> java ssort.java
Sorted Array in Ascending Order:
[2, 10, 12, 15, 20]
PS C:\Users\prasa\OneDrive\Documents\javaprogs> 
```

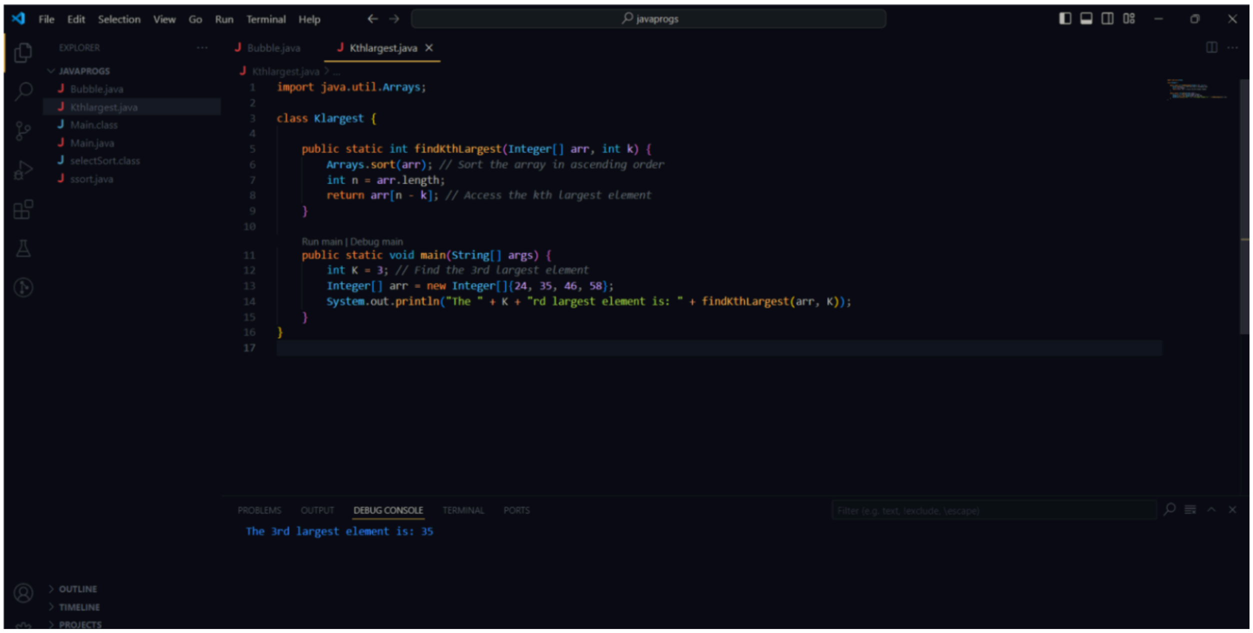
7.Bubble sort



```java
import java.util.Arrays;

class Main {

  static void bubbleSort(int array[]) {
    int size = array.length;

    for (int i = 0; i < size - 1; i++)

      for (int j = 0; j < size - i - 1; j++)

        if (array[j] > array[j + 1]) {

          int temp = array[j];
          array[j] = array[j + 1];
          array[j + 1] = temp;
        }
  }

  public static void main(String args[]) {

    int[] data = { -2, 45, 0, 11, -9 };
    Main.bubbleSort(data);

    System.out.println("Sorted Array in Ascending Order:");
    System.out.println(Arrays.toString(data));
```

```
Sorted Array in Ascending Order:
[-9, -2, 0, 11, 45]
```

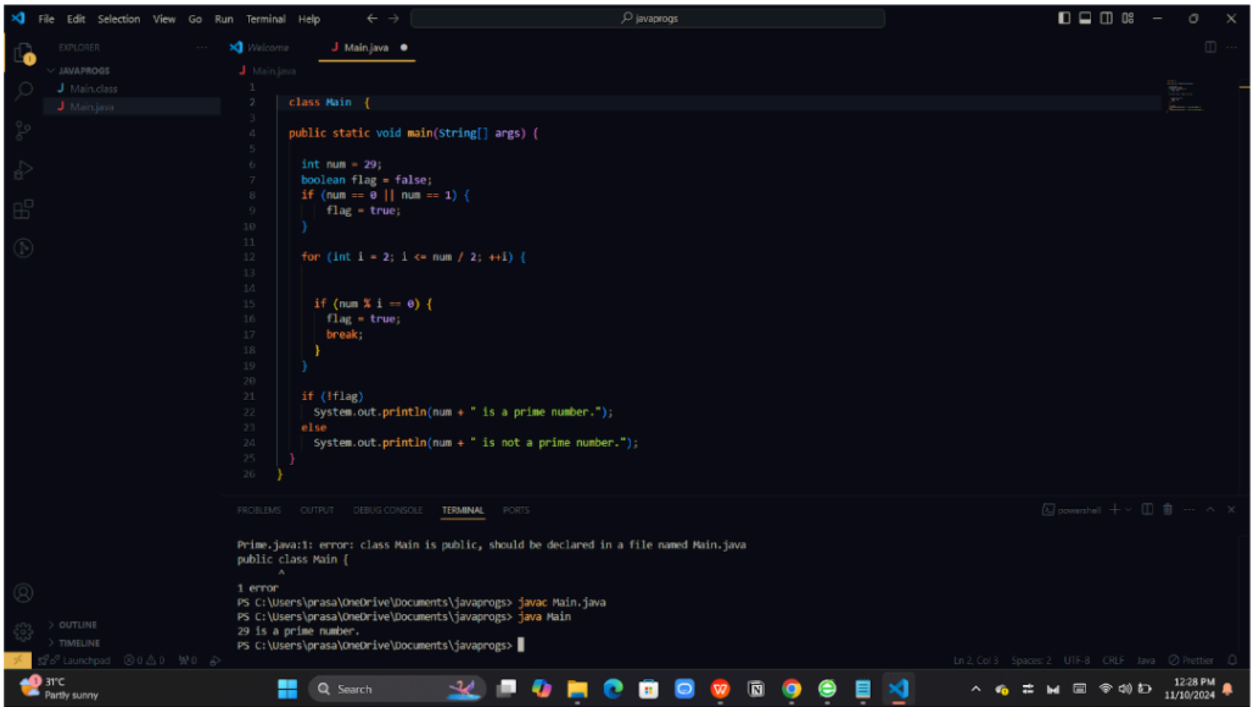
8.Kth largest element



```java
import java.util.Arrays;

class Klargest {

    public static int findKthLargest(Integer[] arr, int k) {
        Arrays.sort(arr); // Sort the array in ascending order
        int n = arr.length;
        return arr[n - k]; // Access the kth largest element
    }

    public static void main(String[] args) {
        int K = 3; // Find the 3rd largest element
        Integer[] arr = new Integer[]{24, 35, 46, 58};
        System.out.println("The " + K + "rd largest element is: " + findKthLargest(arr, K));
    }
}
```

```
The 3rd largest element is: 35
```

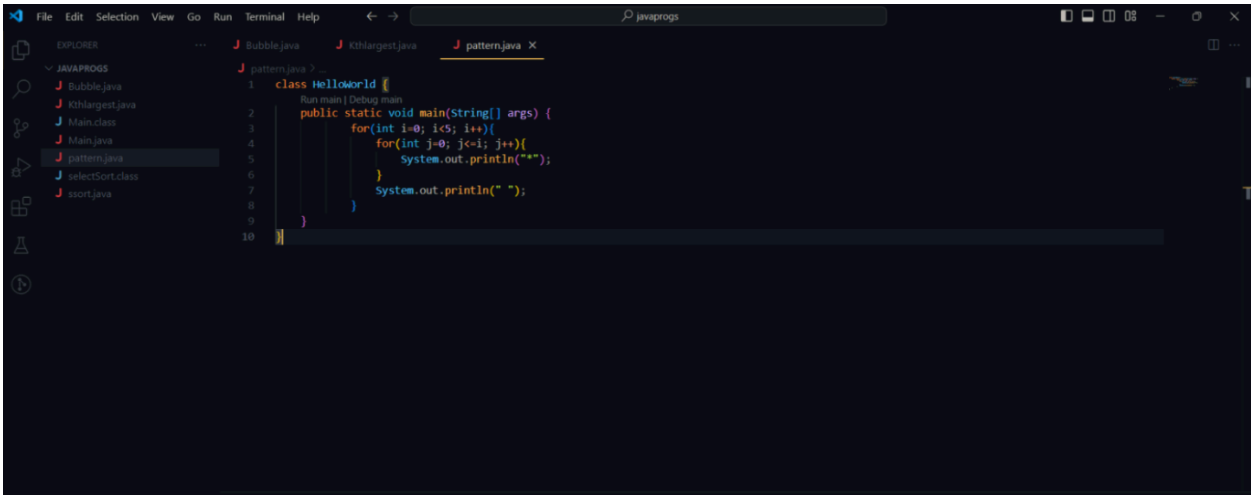
9.Prime number or not



```

class Main  {

public static void main(String[] args) {

  int num = 29;
  boolean flag = false;
  if (num == 0 || num == 1) {
      flag = true;
  }

  for (int i = 2; i <= num / 2; ++i) {


    if (num % i == 0) {
      flag = true;
      break;
    }
  }

  if (!flag)
    System.out.println(num + " is a prime number.");
  else
    System.out.println(num + " is not a prime number.");
 }
}
```

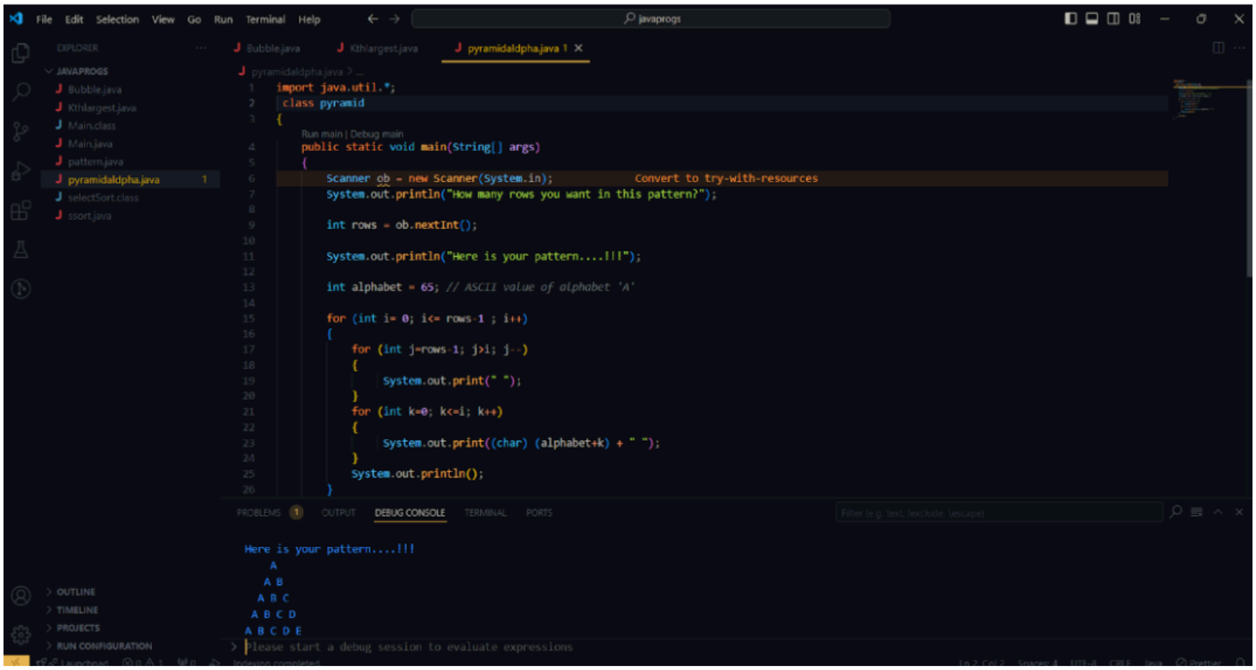
```
Prime.java:1: error: class Main is public, should be declared in a file named Main.java
public class Main {
       ^
1 error
PS C:\Users\prasa\OneDrive\Documents\javaprogs> javac Main.java
PS C:\Users\prasa\OneDrive\Documents\javaprogs> java Main
29 is a prime number.
PS C:\Users\prasa\OneDrive\Documents\javaprogs>
```

10. Pattern program



```
class HelloWorld {
    public static void main(String[] args) {
            for(int i=0; i<5; i++){
                for(int j=0; j<=i; j++){
                    System.out.println("*");
                }
                System.out.println(" ");
            }
    }
}
```

11.pyramid alphabits



```java
import java.util.*;
class pyramid
{
    public static void main(String[] args)
    {
        Scanner ob = new Scanner(System.in);
        System.out.println("How many rows you want in this pattern?");

        int rows = ob.nextInt();

        System.out.println("Here is your pattern....!!!");

        int alphabet = 65; // ASCII value of alphabet 'A'

        for (int i= 0; i<= rows-1 ; i++)
        {
            for (int j=rows-1; j>i; j--)
            {
                System.out.print(" ");
            }
            for (int k=0; k<=i; k++)
            {
                System.out.print((char) (alphabet+k) + " ");
            }
            System.out.println();
        }
```

```
Here is your pattern....!!!
    A
   A B
  A B C
 A B C D
A B C D E
> Please start a debug session to evaluate expressions
```