

1) Duplicate numbers in array

```
File Edit View

import java.util.ArrayList;
import java.util.HashSet;
import java.util.List;
import java.util.Set;

public class RemoveDuplicates {
    public static void main(String[] args) {
        List<Integer> listWithDuplicates = new ArrayList<>();
        listWithDuplicates.add(1);
        listWithDuplicates.add(2);
        listWithDuplicates.add(3);
        listWithDuplicates.add(1);
        listWithDuplicates.add(2);

        Set<Integer> set = new HashSet<>(listWithDuplicates);

        List<Integer> listWithoutDuplicates = new ArrayList<>(set);

        System.out.println("List with duplicates: " + listWithDuplicates);
        System.out.println("List without duplicates: " + listWithoutDuplicates);
    }
}

Ln 7, Col 37 711 characters
```

```
C:\Windows\System32\cmd.e X + v

Microsoft Windows [Version 10.0.22631.3958]
(c) Microsoft Corporation. All rights reserved.

C:\Users\chara\OneDrive\Desktop\Java Prog>java PrimeCheck.java
29 is a prime number.

C:\Users\chara\OneDrive\Desktop\Java Prog>java AnagramCheck.java
The strings are anagrams.

C:\Users\chara\OneDrive\Desktop\Java Prog>java RemoveDuplicates.java
List with duplicates: [1, 2, 3, 1, 2]
List without duplicates: [1, 2, 3]

C:\Users\chara\OneDrive\Desktop\Java Prog>
```

2) Anagram of String

```
File Edit View

import java.util.Arrays;

public class AnagramCheck {
    public static void main(String[] args) {
        String str1 = "listen";
        String str2 = "silent";

        char[] charArray1 = str1.toCharArray();
        char[] charArray2 = str2.toCharArray();

        Arrays.sort(charArray1);
        Arrays.sort(charArray2);

        boolean isAnagram = Arrays.equals(charArray1, charArray2);

        if (isAnagram) {
            System.out.println("The strings are anagrams.");
        } else {
            System.out.println("The strings are not anagrams.");
        }
    }
}

Ln 1, Col 25 583 characters
```

```
C:\Windows\System32\cmd.e X + v

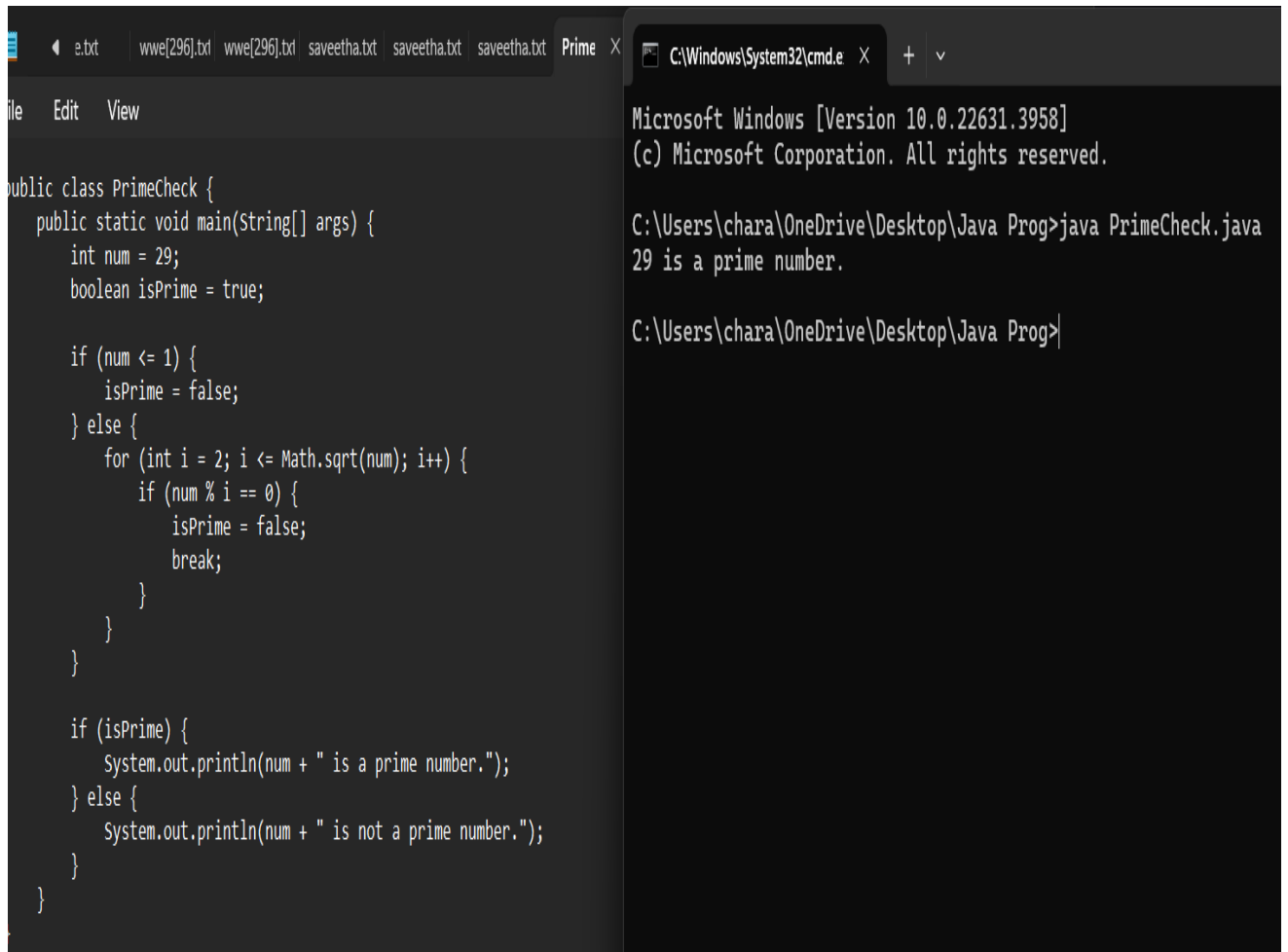
Microsoft Windows [Version 10.0.22631.3958]
(c) Microsoft Corporation. All rights reserved.

C:\Users\chara\OneDrive\Desktop\Java Prog>java PrimeCheck.java
29 is a prime number.

C:\Users\chara\OneDrive\Desktop\Java Prog>java AnagramCheck.java
The strings are anagrams.

C:\Users\chara\OneDrive\Desktop\Java Prog>
```

3) Prime number or not



The image shows a code editor on the left and a command prompt window on the right. The code editor contains a Java program named PrimeCheck.java. The program defines a class PrimeCheck with a main method. In the main method, a variable num is set to 29, and a boolean variable isPrime is set to true. The program then checks if num is less than or equal to 1. If so, isPrime is set to false. Otherwise, it enters a for loop from i = 2 to Math.sqrt(num). If num is divisible by i, isPrime is set to false and the loop breaks. After the loop, it checks if isPrime is true. If true, it prints "29 is a prime number."; otherwise, it prints "29 is not a prime number." The command prompt window shows the execution of the program, displaying the output "29 is a prime number."

```
public class PrimeCheck {  
    public static void main(String[] args) {  
        int num = 29;  
        boolean isPrime = true;  
  
        if (num <= 1) {  
            isPrime = false;  
        } else {  
            for (int i = 2; i <= Math.sqrt(num); i++) {  
                if (num % i == 0) {  
                    isPrime = false;  
                    break;  
                }  
            }  
        }  
  
        if (isPrime) {  
            System.out.println(num + " is a prime number.");  
        } else {  
            System.out.println(num + " is not a prime number.");  
        }  
    }  
}
```

```
Microsoft Windows [Version 10.0.22631.3958]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\chara\OneDrive\Desktop\Java Prog>java PrimeCheck.java  
29 is a prime number.  
  
C:\Users\chara\OneDrive\Desktop\Java Prog>
```