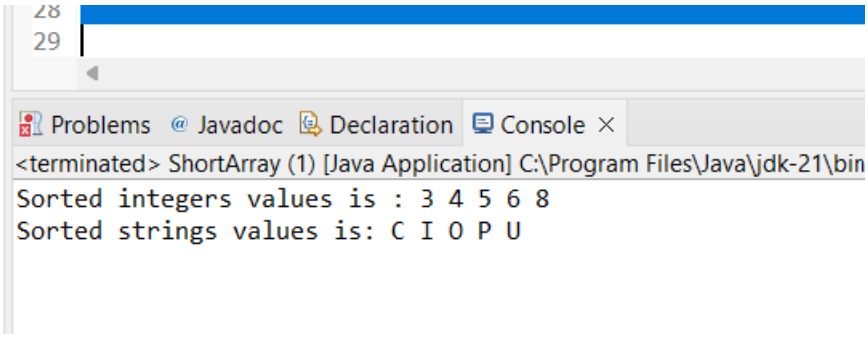
1. Write the programme to sort the integers 8, 4, 3,5,6 and the alphabetical string C, O, I, P, U, in ascending order. Show the resulting output.

CODE:-

```
package lab_6;

import java.util.Arrays;


public class ShortArray {


	public static void main(String[] args) {

		// Integer array

		int[] intArray = {8, 4, 3, 5, 6};

		// String array

		String[] strArray = {"C", "O", "I", "P", "U"};

		// Sort the integer array

		Arrays.sort(intArray);

		// Sort the string array

		Arrays.sort(strArray);

		// Print sorted integer array

		System.out.print("Sorted integers values is : ");

		for (int num : intArray) {

			System.out.print(num + " ");

		}

		System.out.println();

		// Print sorted string array

		System.out.print("Sorted strings values is: ");

		for (String str : strArray) {

			System.out.print(str + " ");

		}

	}

}
```

OUTPUT:-



---

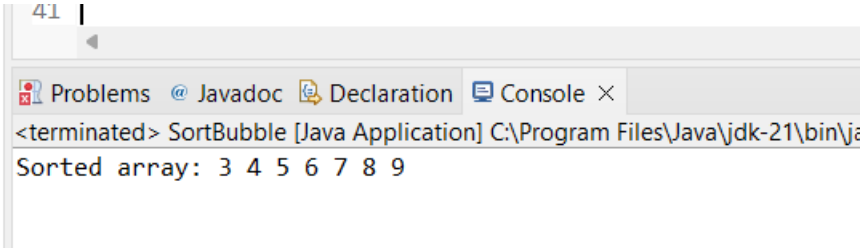
2. Write a Java program to implement the bubble sort algorithm to sort an array of integers in ascending order.

CODE:-

```
package lab_6;


public class SortBubble {


    public static void main(String[] args) {
        // Array of integers to be sorted
        int[] intArray = {8, 4, 3, 5, 6, 7, 9};
        // Perform bubble sort
        bubbleSort(intArray);
        // Print the sorted array
        System.out.print("Sorted array: ");
        for (int num : intArray) {
            System.out.print(num + " ");
        }
    }
    // Bubble sort algorithm
    public static void bubbleSort(int[] array) {
        int n = array.length;
```

```
        boolean swapped;
        // Traverse through all elements in the array
        for (int i = 0; i < n - 1; i++) {
            swapped = false;
            // Last i elements are already sorted, no need to check them
            for (int j = 0; j < n - 1 - i; j++) {
                // Swap if the current element is greater than next
element
                if (array[j] > array[j + 1]) {
                    int temp = array[j];
                    array[j] = array[j + 1];
                    array[j + 1] = temp;
                    swapped = true;
                }
            }

        // If no two elements were swapped in inner loop,the array is sorted
        if (!swapped) break;
    }
}
}
// If no two elements were swapped i
```

OUTPUT:-



3. Write a program to input an array 10 elements and print the cube of prime numbers in it.

CODE:-

```
package lab_6;

import java.util.Scanner;

public class CubeOfNum {


	public static void main(String[] args) {
		Scanner scanner = new Scanner(System.in);
		int[] array = new int[10];
		// Input 10 elements into the array
		System.out.println("Enter 10 elements:");
		for (int i = 0; i < 10; i++) {
			array[i] = scanner.nextInt();
		}
		// Print the cube of prime numbers in the array
		System.out.println("Cubes of prime numbers in the array:");
		for (int num : array) {
			if (isPrime(num)) {
				System.out.println(num + "^3 = " + (num * num * num));
			}
		}
		scanner.close();
	}
	// Method to check if a number is prime
	public static boolean isPrime(int num) {
		if (num <= 1) return false;
		for (int i = 2; i <= Math.sqrt(num); i++) {
			if (num % i == 0) return false;
		}
```

```
                return true;

        }

}
```

OUTPUT:-

```
Enter 10 elements:
1
2
3
4
5
6
7
8
9
10
Cubes of prime numbers in the array:
2^3 = 8
3^3 = 27
5^3 = 125
7^3 = 343
```

---

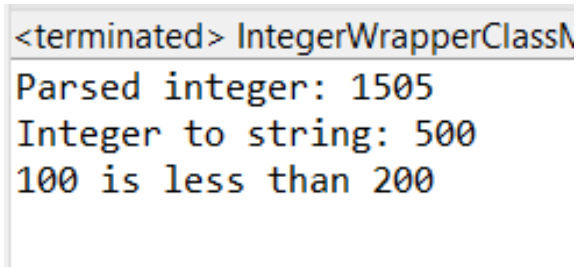
4. Write a java program to implement integer wrapper class methods.(any 3 methods)

CODE:-

```
package Hellow;


public class IntegerWrapperClassMethods {

public static void main(String[] args) {

// Method 1: parseInt String numberStr = "1505";

int number = Integer.parseInt(numberStr); System.out.println("Parsed integer: " + number);


// Method 2: toString

int anotherNumber = 500;

String anotherNumberStr = Integer.toString(anotherNumber); System.out.println("Integer to string: " + anotherNumberStr);


// Method 3: compareTo Integer num1 = 100; Integer num2 = 200;
```

```
int comparisonResult = num1.compareTo(num2);

if (comparisonResult < 0) {

System.out.println(num1 + " is less than " + num2);

} else if (comparisonResult > 0) {

System.out.println(num1 + " is greater than " + num2);

} else {

System.out.println(num1 + " is equal to " + num2);

}

}

}
```

OUTPUT:-



---

5. Write a java program to implement double wrapper class methods.(any 3 methods)

CODE:-

```
package Hellow;


public class DoubleWrapperClassMethods {

public static void main(String[] args) {

// Method 1: parseDouble String doubleStr = "123.45";

double number = Double.parseDouble(doubleStr); System.out.println("Parsed double: " +
number);


// Method 2: toString

double anotherNumber = 456.78;

String anotherNumberStr = Double.toString(anotherNumber); System.out.println("Double to
string: " + anotherNumberStr);
```

```
// Method 3: compareTo Double num1 = 100.25; Double num2 = 200.50;
int comparisonResult = num1.compareTo(num2);
if (comparisonResult < 0) {
System.out.println(num1 + " is less than " + num2);
} else if (comparisonResult > 0) {
System.out.println(num1 + " is greater than " + num2);
} else {
System.out.println(num1 + " is equal to " + num2);
}
}
}
```

OUTPUT:-

```
<terminated> DoubleWrapperClassM
Parsed double: 123.45
Double to string: 456.78
100.25 is less than 200.5
```

---

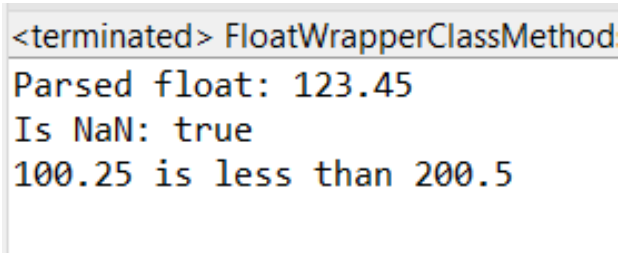
6. Write a java program to implement float wrapper class methods.(any 3 methods)

CODE:-

```
package Hellow;

public class FloatWrapperClassMethods {
public static void main(String[] args) {
// Method 1: parseFloat String floatStr = "123.45";
float parsedFloat = Float.parseFloat(floatStr); System.out.println("Parsed float: " + parsedFloat);

// Method 2: isNaN
Float nanValue = Float.NaN; System.out.println("Is NaN: " + nanValue.isNaN());

// Method 3: compareTo Float num1 = 100.25f; Float num2 = 200.50f;
int comparisonResult = num1.compareTo(num2);
```

```
if (comparisonResult < 0) {
System.out.println(num1 + " is less than " + num2);
} else if (comparisonResult > 0) {
System.out.println(num1 + " is greater than " + num2);
} else {
System.out.println(num1 + " is equal to " + num2);
}
}
}
```

OUTPUT:-



---

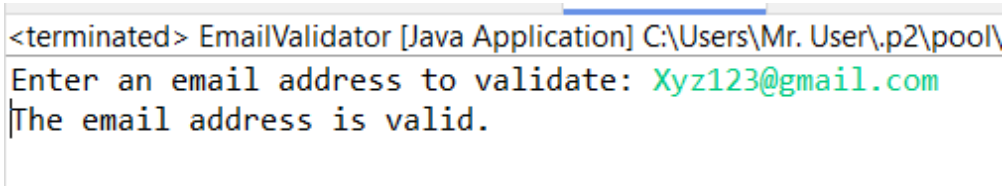
7. Write a Java program to validate email addresses using regular expressions. The email should have the format username@domain.com where username and domain can contain alphanumeric characters, dots, and hyphens.

CODE:-

```
package Hellow;

import java.util.regex.Matcher; import java.util.regex.Pattern; import java.util.Scanner;

public class EmailValidator {
// Regular expression for validating email addresses
private static final String EMAIL_REGEX = "^[a-zA-Z0-9._-]+@[a-zA-Z0-9.-
]+\\.[a-zA-Z]{2,}$";

public static void main(String[] args) { Scanner scanner = new Scanner(System.in);
```

```
// Input email address

System.out.print("Enter an email address to validate: "); String email = scanner.nextLine();


// Validate email address

if (isValidEmail(email)) {

System.out.println("The email address is valid.");

} else {

System.out.println("The email address is invalid.");

}


scanner.close();

}


// Method to validate email address using regex

public static boolean isValidEmail(String email) { Pattern pattern =
Pattern.compile(EMAIL_REGEX); Matcher matcher = pattern.matcher(email); return
matcher.matches();

}

}
```

OUTPUT:-



---

8. Create a Java program to validate phone numbers. The format should be (xxx) xxx-xxxx where x is a digit.

CODE:-

```
package Hellow;


import java.util.Scanner;

import java.util.regex.Pattern;
```

```
public class PhoneNumberValidator {

// Regular expression for validating phone numbers

private static final Pattern PHONE_PATTERN = Pattern.compile("^\\(\\d{3}\\)

\\d{3}-\\d{4}$");


public static void main(String[] args) { Scanner scanner = new Scanner(System.in);


                // Input phone number

                System.out.print("Enter a phone number to validate (format: (xxx) xxx- String phoneNumber = scanner.nextLine();
xxxx): ");

                // Validate phone number and print result

                if (PHONE_PATTERN.matcher(phoneNumber).matches()) { System.out.println("The phone number is valid.");

                } else {

                System.out.println("The phone number is invalid.");
```

```
}


scanner.close();

}

}
```

OUTPUT:-