

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 MyPackage;
//importing packages
import java.util.Arrays;
public class ArraySorter
{
    public static void main(String[] args)
    {
        //declaring array of given number
        int[] numbers= {8, 4, 3, 5, 6};

        //declaring array of given alphabets
        String[] alphabets = {"C", "O", "I", "P", "U"};

        //printing original array of numbers
        System.out.println("Given array of numbers : " +
        Arrays.toString(numbers));
        //sorting array of numbers
        Arrays.sort(numbers);
        //printing sorted array of numbers
        System.out.println("Sorted array of numbers : " +
        Arrays.toString(numbers));

        //printing original array of alphabets
        System.out.println("Given array of alphabets : " +
        Arrays.toString(alphabets));
        //sorting array of alphabets
        Arrays.sort(alphabets);
        //printing sorted array of alphabets
        System.out.println("Sorted array of alphabets : " +
        Arrays.toString(alphabets));
    }
}
```

**Output:-**

```
Given array of numbers : [8, 4, 3, 5, 6]
Sorted array of numbers : [3, 4, 5, 6, 8]
Given array of alphabets : [C, O, I, P,
    U]
Sorted array of alphabets : [C, I, O, P,
    U]
```

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

**Code:-**

```
package MyPackage;
//importing packages
import java.util.Arrays;
public class ArraySorter
{
    //creating a function which takes array as arguments
    static void bubbleSort(int[] arr)
    {
        //getting the length of array
        int n = arr.length;

        int temp;
        //outer loop iterate on array elements and inner compare 2 elements of
        array and place the greatest element at 2nd position and by swapping
        elements it place the greatest element at end
        for (int i=0; i<n-1; i++) {
            for (int j=0; j<n-i-1; j++) {
                if (arr[j] > arr[j+1]) {
                    temp = arr[j];
                    arr[j] = arr[j+1];
                    arr[j+1] = temp;
                }
            }
        }

        System.out.println("Sorted array using bubble sort algorithm : " +
        Arrays.toString(arr));
    }

    public static void main (String[] args)
    {
        //declaring the array of integers
        int[] num= {8, 4, 7, 3, 9, 5, 2, 6, 1};

        //calling the bubbleSort function to sort array by swapping elements
        bubbleSort(num);
    }
}
```

**Output:-**

```
Sorted array using bubble sort algorithm
: [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

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

Code:-

```
package MyPackage;
//importing packages
import java.util.Arrays;
import java.util.Scanner;
public class PrimeNumber
{
    public static void main(String[] args)
    {
        //creating the object of Scanner class
        Scanner sc=new Scanner(System.in);

        //declaring array of size 10
        int[] num = new int[10];

        //taking input from user and storing in array
        System.out.println("Enter the elements of array : ");
        for (int i=0; i<10; i++) {
            System.out.print("Enter element no. " + (i+1) + " : ");
            num[i] = sc.nextInt();
        }

        //printing the array
        System.out.println("Array of 10 numbers : " + Arrays.toString(num));

        System.out.println("Prime numbers and their cube : ");

        //advanced for loop iterate over array
        for (int n : num) {
            int count = 0;
            //inner loop take the element of array and divide it by number from 1
            //to number equal to array element
            for (int j=1; j<=n; j++){
                //if number is divided by then int increment the count variable by
                1
                if (n % j == 0) {
                    count++;
                }
            }
            //if value of count variable is equals to 2 means the number is has
            //only to divisor 1 and number itself means the number is prime number
            //since the prime number has only one divisor 1 and number itself

            if (count == 2) {
                //it will print the number and its output
                System.out.println(n + " = " + n*n*n);
            }
        }
    }
}
```

Output:-

```
Enter the elements of array :
Enter element no. 1 : 2
Enter element no. 2 : 9
Enter element no. 3 : 5
Enter element no. 4 : 7
Enter element no. 5 : 6
Enter element no. 6 : 15
Enter element no. 7 : 13
Enter element no. 8 : 21
Enter element no. 9 : 17
Enter element no. 10 : 27
Array of 10 numbers : [2, 9, 5, 7, 6, 15,
    13, 21, 17, 27]
Prime numbers and their cube :
2 = 8
5 = 125
7 = 343
13 = 2197
17 = 4913
```

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

Code:-

```
package MyPackage;
public class IntegerMethods
{
    public static void main (String[] args)
    {
        //creating object of integer wrapper
        Integer num1 = new Integer(50);
        Integer num2 = new Integer(70);

        // valueOf method
        System.out.println("Printing the value of num1: " +
            Integer.valueOf(num1));

        // compare to method
```

```

        //num1 < num2 = -1
        //num1 = num2 = 0
        //num1 > num2 = 1
        System.out.println("Comparing num1 and num2 : " +
num1.compareTo(num2));

        // parse int method
        String strNum = "25";
        Int parsedNum = Integer.parseInt(strNum);

        System.out.println("Parsed Integer value : " + parsedNum);
    }
}

```

Output:-

```

Printing the value of num1: 50
Comparing num1 and num2 : -1
Parsed Integer value : 25

```

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

Code:-

```

package MyPackage;
public class DoubleMethods
{
    Public static void main (String[] args)
    {
        //creating object of Double wrapper
        Double num1 = new Double(40.5);
        Double num2 = new Double(30.5);

        // valueOf method
        System.out.println("Printing the value of num1: " +
Double.valueOf(num1));

        // compare to method
        //num1 < num2 = -1
        //num1 = num2 = 0
        //num1 > num2 = 1
        System.out.println("Comparing num1 and num2 : " +
num1.compareTo(num2));

        // parse int method
        String strNum = "25";
        Double parsedNum = Double.parseDouble(strNum);
        System.out.println("Parsed Double value : " + parsedNum);

        //NaN method
        System.out.println("Checking num1 is not a number : " + num1.isNaN());
    }
}

```

Output:-

```
Printing the value of num1: 40.5
Comparing num1 and num2 : 1
Parsed Double value : 25.0
Checking num1 is not a number : false
```

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

Code:-

```
package MyPackage;
public class FloatMethods
{
    public static void main (String[] args)
    {
        //creating object of Float wrapper
        Float num1 = new Float(30.5f);
        Float num2 = new Float(30.5f);

        // valueOf method
        System.out.println("Printing the value of num1: " +
            Float.valueOf(num1));

        // compare to method
        //num1 < num2 = -1
        //num1 = num2 = 0
        //num1 > num2 = 1
        System.out.println("Comparing num1 and num2 : " +
            num1.compareTo(num2));

        // parse int method
        String strNum = "25";
        Float parsedNum = Float.parseFloat(strNum);
        System.out.println("Parsed Float value : " + parsedNum);

        //NaN method
        System.out.println("Checking num1 is not a number : " + num1.isNaN());
    }
}
```

Output:-

```
Printing the value of num1: 30.5
Comparing num1 and num2 : 0
Parsed Float value : 25.0
Checking num1 is not a number : false
```

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 MyPackage;
//importing packages
import java.util.regex.*;
import java.util.Scanner;
public class EmailValidator
{
    public static void main(String[] args)
    {
        //creating a email pattern
        String emailPattern = "[a-zA-Z0-9.-]+@[a-zA-Z0-9.-]+\\.com$";

        //storing pattern
        Pattern p = Pattern.compile(emailPattern);

        //creating object of Scanner
        Scanner sc = new Scanner(System.in);
        //taking user input
        System.out.print("Enter your email : ");
        //storing user input
        String email = sc.nextLine();

        //matching email with pattern
        Matcher m = p.matcher(email);

        //validating email if matches
        If (m.matches()) {
            System.out.println(email + " is a valid email.");
        } else {
            System.out.println(email + " is an invalid email.");
        }
    }
}
```

Output:-

```
Enter your email : abc-123@xyz.com
abc-123@xyz.com is a valid email.
```

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

Code:-

```
package MyPackage;
//importing packages
import java.util.regex.*;
import java.util.Scanner;
Public class PhoneNumberValidator
{
    Public static void main(String[] args)
    {
        //creating a phone number pattern
        String numberPattern = "^\\(\\d{3}\\) \\d{3}-\\d{4}$";

        //storing number pattern
        Pattern p = Pattern.compile(numberPattern);

        //creating object of Scanner
        Scanner sc = new Scanner(System.in);
        //taking user input
        System.out.print("Enter your phone number : ");
        //storing user input
        String number = sc.nextLine();

        //matching number with pattern
        Matcher m = p.matcher(number);

        //validating number if matches with number pattern
        If (m.matches()) {
            System.out.println(number + " is a vailid number.");
        } else {
            System.out.println(number + " is an invailid number.");
        }
    }
}
```

Output:-

```
Enter your phone number : (123) 636-5373
(123) 636-5373 is a vailid number.
```