1. Write a function that takes in two numbers and returns their sum.
2. import java.util.Scanner;
3. class Sum {
4. static int Summation(int a, int b) {
5. return a + b;
6. }
7. public static void main(String[] args) {
8. int a, b, c;
9. Scanner sc = new Scanner(System.in);
10. System.out.println("enter a");
11. a = sc.nextInt();
12. System.out.println("enter b");
13. b = sc.nextInt();
14. c = Summation(a, b);
15. System.out.println(c);
16. }
17. }

2. Write a function that takes in an array of numbers and returns the sum of all the numbers.

import java.util.Scanner;

public class ArraySum {

    static int ArraySummation(int a[])

    {

        int ans=0,i;

        for(i=0;i<=a.length;i++)

        {

            ans = ans +i;

        }

        return ans;

    }

    public static void main(String[] args) {

        int n;

        Scanner sc = new Scanner(System.in);

        System.out.println("enter n");

        n = sc.nextInt();

        int[] arr = new int[n];

        for(int i =0;i<arr.length;i++)

        {

            arr[i]=sc.nextInt();

        }

        System.out.println(ArraySummation(arr));

    }

}

3. Write a function that takes in a string and returns the reverse of the string

import java.util.Scanner;

public class StringReverse {

    static String Revesed(String str)

    {

        char ch;

        String rstr="";

        int l=str.length()-1;

        for(int i=l; i>=0; i--)

        {

            ch = str.charAt(i);

            rstr = rstr + ch;

        }

        return rstr;

    }

    public static void main(String[] args) {

        String str;

        Scanner sc=new Scanner(System.in);

        System.out.println("enter a word");

        str = sc.nextLine();

        System.out.println("reverse is "+ Revesed(str));

    }

}

//we can also used string builder built in method

// Java program to ReverseString using StringBuilder

// import java.lang.\*;

// import java.io.\*;

// import java.util.\*;

// Class of ReverseString

// class ReverseString {

//  public static void main(String[] args)

//  {

//      String input = "Geeks for Geeks";

//      StringBuilder input1 = new StringBuilder();

//      // append a string into StringBuilder input1

//      input1.append(input);

//      // reverse StringBuilder input1

//      input1.reverse();

//      // print reversed String

//      System.out.println(input1);

//  }

// }

4. Write a function that takes in an array of strings and returns a new array with all the

strings capitalized.

import java.util.Scanner;

public class StringCapitaliza {

    static String Ggh(String arr[])

{

    String str="";

    String demo;

    for (int i = 0; i < arr.length; i++) {

        demo = arr[i].substring(0, 1).toUpperCase() + arr[i].substring(1);

        str = str +" "+demo;

    }

    return str;

}

    public static void main(String[] args) {

        String str;

        Scanner sc=new Scanner(System.in);

        System.out.println("enter a word");

        str = sc.nextLine();

        String arr[] = str.split(" ",5);

        System.out.println(Ggh(arr));

    }

}

5. Write a function that takes in a number and returns true if the number is even and false if

it is odd.

import java.util.Scanner;

public class EvenOdd {

    static boolean EvOdd(int n)

    {

        if (n%2==0)

        {

            return true;

        }

        return false;

    }

    public static void main(String[] args) {

        int n;

        Scanner sc = new Scanner(System.in);

        n = sc.nextInt();

        System.out.println(EvOdd(n));

    }

}

6. Write a function that takes in an array of numbers and returns the largest number in the

Array

import java.util.Scanner;

public class LargeNoInArray {

    static int LargeNo(int arr[])

    {

        int max = 0;

        for (int i = 0; i < arr.length; i++) {

            if (arr[i]>max)

            {

                max=arr[i];

            }

            else

            {

                continue;

            }

        }

        return max;

    }

    public static void main(String[] args) {

        int n;

        Scanner sc = new Scanner(System.in);

        System.out.println("how many no of elements : ");

        n = sc.nextInt();

        int arr[] = new int[n];

        for (int i = 0; i < arr.length; i++) {

            arr[i] = sc.nextInt();

        }

        System.out.println("max is :"+LargeNo(arr));

    }

}

7. Write a function that takes in an array of numbers and returns a new array with only the

even numbers

import java.lang.reflect.Array;

import java.util.ArrayList;

import java.util.Scanner;

public class EvenArray {

    static ArrayList<Integer> EvenArr(int arr[])

    {

        int arr1[];

       ArrayList<Integer> arrayList = new ArrayList<Integer>();

       for (int i = 0; i < arr.length; i++) {

        if(arr[i]%2==0)

        {

            arrayList.add(arr[i]);

        }

       }

    return arrayList;

    }

    public static void main(String[] args) {

        int n;

        Scanner sc = new Scanner(System.in);

        System.out.println("how many no of elements : ");

        n = sc.nextInt();

        int arr[] = new int[n];

        for (int i = 0; i < arr.length; i++) {

            arr[i] = sc.nextInt();

        }

        System.out.println(EvenArr(arr));

    }

}

8. Write a function that takes in a string and returns true if the string is a palindrome (reads

the same backwards and forwards) and false otherwise

//in these program i have used the previous class StringReversed which contains reverse of string

import java.util.Scanner;

public class Palindrome {

public static void main(String[] args) {

    String str1;

    String str2;

    Scanner sc = new Scanner(System.in);

    System.out.println("enter string to check palindrome : ");

    str1 = sc.nextLine();

    StringReverse sr = new StringReverse();

    str2=StringReverse.Revesed(str1);

    if (str1.equals(str2))

    {

        System.out.println("palindrome");

    }

    else{

        System.out.println("not a palindrome");

    }

}

}

9. Write a function that takes in an array of strings and returns a new array with all the

strings sorted in alphabetical order

import java.util.\*;

public class StringSort {

    static ArrayList<String> SortArr(String arr[])

    {

        ArrayList<String> arrlt = new ArrayList<String>();

        for (int i = 0; i < arr.length; i++) {

            arrlt.add(arr[i]);

        }

        Collections.sort(arrlt);

        return arrlt;

    }

    public static void main(String[] args) {

        int n;

        Scanner sc = new Scanner(System.in);

        System.out.println("how many no of elements : ");

        n = sc.nextInt();

        String arr[] = new String[n];

        for (int i = 0; i <arr.length; i++) {

            arr[i] = sc.nextLine();

        }

        System.out.println(SortArr(arr));

    }

}

10. Write a function that takes in a number and returns the factorial of that number (the

product of all positive integers up to and including that number).

import java.util.Scanner;

public class FactorialOfNo {

    static int factorial(int n){

        if (n == 0)

          return 1;

        else

          return(n \* factorial(n-1));

    }

    public static void main(String[] args) {

     int n;

     Scanner sc = new Scanner(System.in);

     System.out.println("enter n : ");

     n = sc.nextInt();

     System.out.println(factorial(n));

    }

}

11. Write a function that takes in two strings and returns true if they are anagrams (contain

the same letters in a different order) and false otherwise.

import java.util.Arrays;

import java.util.Scanner;

public class CheckAnagram {

    static boolean Anagram(String s1,String s2)

    {

        char[] arrayS1 = s1.toLowerCase().toCharArray();

        char[] arrayS2 = s2.toLowerCase().toCharArray();

        Arrays.sort(arrayS1);

        Arrays.sort(arrayS2);

        boolean status = Arrays.equals(arrayS1, arrayS2);

        if (status == true)

        {

            return true;

        }

        return false;

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("enter string 1 : ");

        String str1 = sc.nextLine();

        System.out.println("enter string 2 : ");

        String str2 = sc.nextLine();

        System.out.println(Anagram(str1, str2));

    }

}

//steps

/\*

 Read or initialize two strings str1 and str2.

Find the length of both the strings.

Compare the length of the strings.

If length is not equal, print strings are not an anagram.

Else, do the following:

Convert the string into a character array.

Sort both the arrays by using the sort() method.

After sorting, compare the strings by using the equals() method. Store the value in a Boolean variable (status) returned by the equals() method.

Pass the variable in the if statement. If it returns true, the given strings are anagram. Else, not an anagram

\*/

12. Write a function that takes in an array of numbers and returns the second smallest number

in the array.

import java.util.\*;

public class SecondSmalles {

    static int Secondsmall(int arr[])

    {

        Arrays.sort(arr);

        //int k =arr.length-2;//if it is second max

        return arr[1];

    }

    public static void main(String[] args) {

        int n;

        Scanner sc = new Scanner(System.in);

        System.out.println("how many no of elements : ");

        n = sc.nextInt();

        int arr[] = new int[n];

        for (int i = 0; i < arr.length; i++) {

            arr[i] = sc.nextInt();

        }

        System.out.println("2nd small number  is :"+Secondsmall(arr));

    }

}

13. Write a function that takes in an array of strings and returns a new array with all the

strings that are palindromes.

import java.util.ArrayList;

import java.util.Scanner;

public class PalindromeArray {

    static ArrayList<String> ArrPalindrome(String arr[])

    {

        ArrayList<String> list = new ArrayList<String>();

        for (int i = 0; i < arr.length; i++) {

            if (arr[i].equals(StringReverse.Revesed(arr[i])))

            {

                list.add(arr[i]);

            }

        }

        return list;

    }

    public static void main(String[] args) {

        int n;

        Scanner sc = new Scanner(System.in);

        System.out.println("how many no of elements : ");

        n = sc.nextInt();

        String arr[] = new String[n];

        for (int j = 0; j < arr.length; j++) {

            System.out.println("enter "+j);

            arr[j] = sc.next();

        }

        System.out.println(ArrPalindrome(arr));

    }

}

14. Write a function that takes in an array of numbers and returns a new array with all the

numbers sorted in descending order.

import java.lang.reflect.Array;

import java.util.\*;

public class ArraySortDesc {

    static ArrayList<Integer> SortDesc(int arr[])

    {

        ArrayList<Integer> list=new ArrayList<Integer>();

        for (int i = 0; i < arr.length; i++) {

            list.add(arr[i]);

        }

        Collections.sort(list, Collections.reverseOrder());

        return list;

    }

    public static void main(String[] args) {

        int n;

        Scanner sc = new Scanner(System.in);

        System.out.println("how many no of elements : ");

        n = sc.nextInt();

        int arr[] = new int[n];

        for (int i = 0; i < arr.length; i++) {

            arr[i] = sc.nextInt();

        }

        System.out.println(SortDesc(arr));

    }

}

15. Write a function that takes in a number and returns true if it is a prime number (only

divisible by 1 and itself) and false otherwise.

import java.util.Scanner;

public class PrimeNo {

    static boolean isPrime(int n)

    {

        for(int i = 2;i<n;i++)

        {

            if (n%i==0)

            {

                return false;

            }

        }

        return true;

    }

    public static void main(String[] args) {

        int n;

        Scanner sc = new Scanner(System.in);

        System.out.println("enter any no : ");

        n = sc.nextInt();

        if (n==0|n==1)

        {

            System.out.println(n+" is not prime number");

        }

        else

        {

            System.out.println(isPrime(n));

        }

    }

}