END TERM PRACTICAL EXAM Date- 7-8-2021

**SUBJECT- JAVA Programming (PMC-202)**

**NAME-Garima Bisht SEM-2 COURSE-MCA CAMPUS- Haldwani**

**STUDENT ID- 20712004 UNIVERSITY ROLL NO-2098004**

Q1-Write an applet program which can perform the arithmetic operations like Sum, Subtract, Multiplication & Division.

/\*

Answer1

Garima Bisht

\*/

import java.awt.\*;

import java.applet.\*;

import java.awt.event.\*;

public class calculator extends Applet implements

ActionListener

{

Label label1, label2, label3;

TextField tf1, tf2, tf3;

Button b1, b2, b3, b4;

String whichButtonClk;

@Override

public void init()

{

System.out.println("Initializing an applet");

label1 = new Label("Number1");

tf1= new TextField(10);

label2 = new Label("Number2");

tf2= new TextField(10);

b1 = new Button("Add");

b2= new Button("Subtract");

b3 = new Button("Multiply");

b4= new Button("Divide");

add(label1);

add(tf1);

add(label2);

add(tf2);

add(b1);

add(b2);

add(b3);

add(b4);

tf1.addActionListener(this); // first textfield event

tf2.addActionListener(this); // second textfield

event

b1.addActionListener(this); // first button event

b2.addActionListener(this); // second button event

b3.addActionListener(this); // third button event

b4.addActionListener(this); // fourth button event

}

public void actionPerformed(ActionEvent ae)

{

if(ae.getActionCommand().equals("Add") ||

ae.getActionCommand().equals("Subtract")

||ae.getActionCommand().equals("Multiply")

||ae.getActionCommand().equals("Divide"))// checking

if an event of clicking the add/subtract/multiply/divide

button is generated

{

whichButtonClk=ae.getActionCommand();

//initializing whichButtonClk to a String value of

Button which is clicked

repaint();

}

}

public void paint(Graphics g)

{

g.drawString("Please enter two numbers to perform

math operations", 10,130);

if(tf1.getText().equals("") && tf2.getText().equals(""))

//if the add button is clicked when textfields are

empty

{

}

else

{

Integer i1= new Integer(tf1.getText());

Integer i2= new Integer(tf2.getText());

int sum = i1+i2;

int subtract=i1-i2;

int multiply=i1\*i2;

float divide=(float)i1/(float)i2; //Casting int to

float, to get precise division of two values in float

if(whichButtonClk.equals("Add"))

g.drawString("Your sum is "+ sum, 10,190);

if(whichButtonClk.equals("Subtract"))

g.drawString("Your subtract is "+ subtract,

10,190);

if(whichButtonClk.equals("Multiply"))

g.drawString("Your multiply is "+ multiply,

10,190);

if(whichButtonClk.equals("Divide"))

g.drawString("Your divide is "+ divide,

10,190);

}

}

}

**OUTPUT**

Q2- Write a program which writes the content on a file, save this file with your name, read this file and count the number of words in the file and print it in the console.

/\*

Answer2

Garima Bisht

\*/

**Code**

import java.util.\*;

import java.io.\*;

class Sentence

{

public static void main(String args[])

throws IOException

{

int nl=1,nw=0; char ch;

Scanner scr=new Scanner(System.in);

System.out.print("\nEnter File name: ");

String str=scr.nextLine();

FileInputStream f=new FileInputStream(str);

int n=f.available();

for(int i=0;i<n;i++)

{

ch=(char)f.read();

if(ch=='\n')

nl++;

else if(ch==' ')

nw++;

}

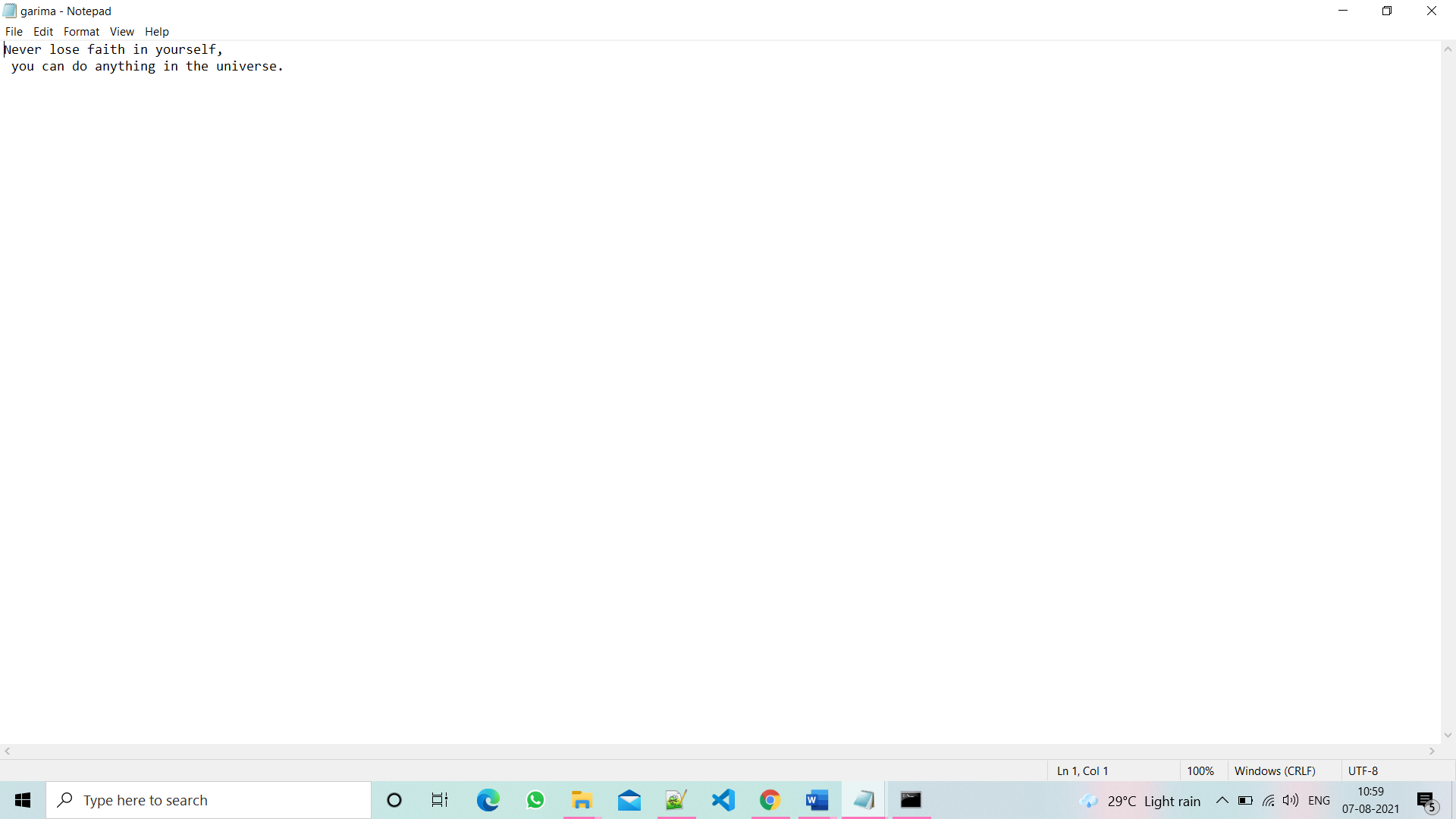
System.out.println("\nNumber of lines : "+nl);

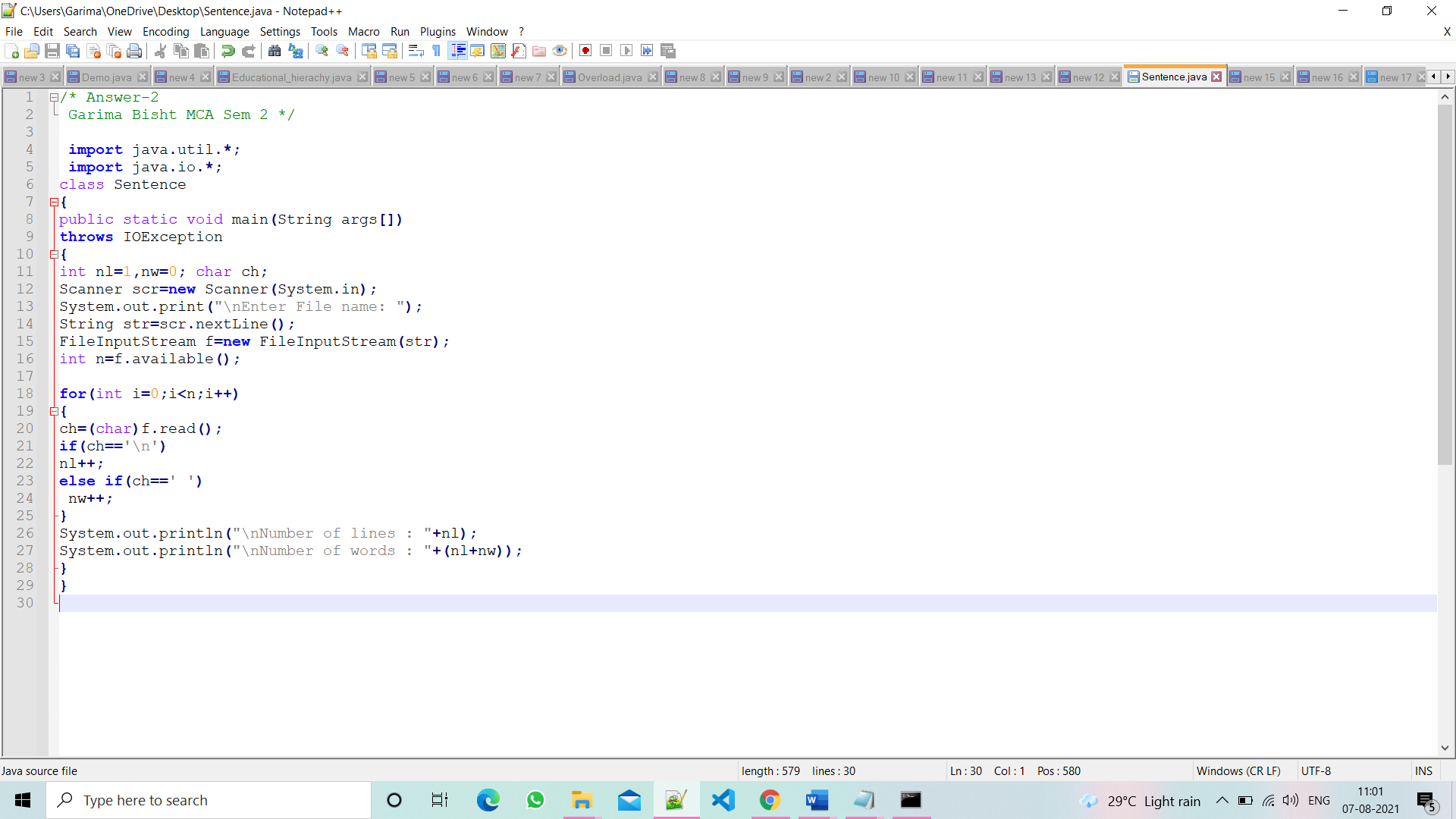
System.out.println("\nNumber of words : "+(nl+nw));

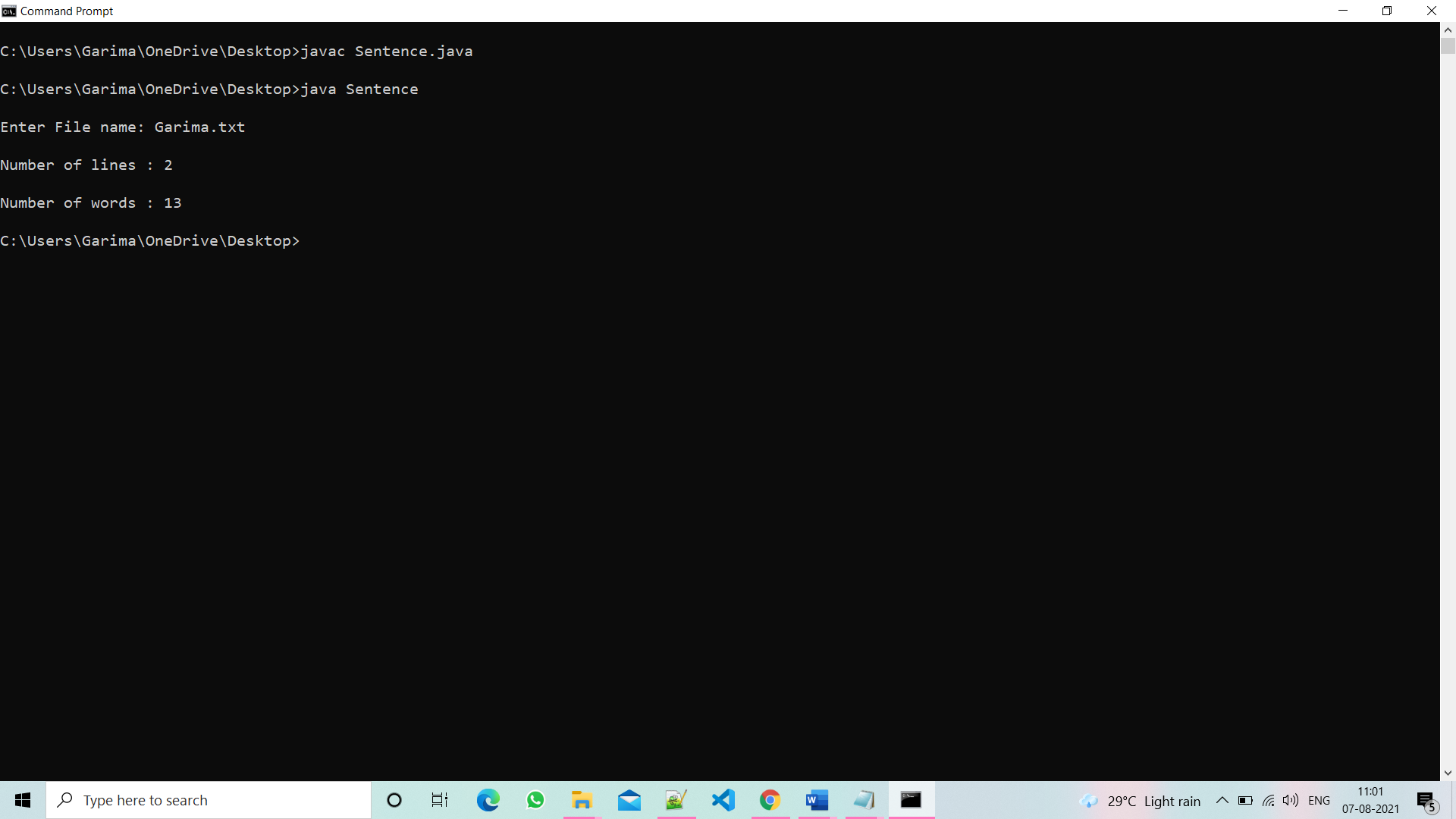
}

}

**OutPut**







Q3. Write a program where client sends a string and server returns the reverse of the string Using TCP/IP Socket Programming

/\*

Answer3 Client Sends

Garima Bisht

\*/

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.io.OutputStream;

import java.io.OutputStreamWriter;

import java.net.Socket;

import java.util.Scanner;

public class Client1 {

private static Socket socket;

public static void main(String args[]) {

try {

socket = new Socket("127.0.0.1", 4000);

System.out.println("Client Running...");

OutputStream os = socket.getOutputStream();

OutputStreamWriter osw = new OutputStreamWriter(os);

BufferedWriter bw = new BufferedWriter(osw);

System.out.println("Type in a string and Press Enter...");

Scanner sc = new Scanner(System.in);

String string = sc.next();

System.out.println("string = " + string);

String sendMessage = string + "\n"; ////Next to line

bw.write(sendMessage);

bw.flush();

System.out.println("Message sent to the server : " + sendMessage);

InputStream is = socket.getInputStream();

InputStreamReader isr = new InputStreamReader(is);

BufferedReader br = new BufferedReader(isr);

String message = br.readLine();

System.out.println("Message received from the server : " + message);

} catch (Exception exception) {

exception.printStackTrace();

} finally {

try {

socket.close();

} catch (Exception e) {

e.printStackTrace();

}

}

}

}

/\*

Server returns the String

\*/

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.io.OutputStream;

import java.io.OutputStreamWriter;

import java.net.ServerSocket;

import java.net.Socket;

public class Server1 {

private static Socket socket;

public static void main(String[] args) {

try {

ServerSocket serverSocket = new ServerSocket(4000);

System.out.println("Server Running...");

while (true) {

socket = serverSocket.accept();

InputStream is = socket.getInputStream();

InputStreamReader isr = new InputStreamReader(is);

BufferedReader br = new BufferedReader(isr);

String string = br.readLine();

System.out.println("Message received from client is " + string);

try {

StringBuilder input = new StringBuilder();

input.append(string);

input = input.reverse();

string = input + "\n";

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

System.out.println(input.charAt(i));

}

} catch (Exception e) {

string = "Please send a proper text message\n";

}

OutputStream os = socket.getOutputStream();

OutputStreamWriter osw = new OutputStreamWriter(os);

BufferedWriter bw = new BufferedWriter(osw);

bw.write(string);

System.out.println("Message sent to the client is " + string);

bw.flush();

}

} catch (Exception e) {

e.printStackTrace();

} finally {

try {

socket.close();

} catch (Exception e) {

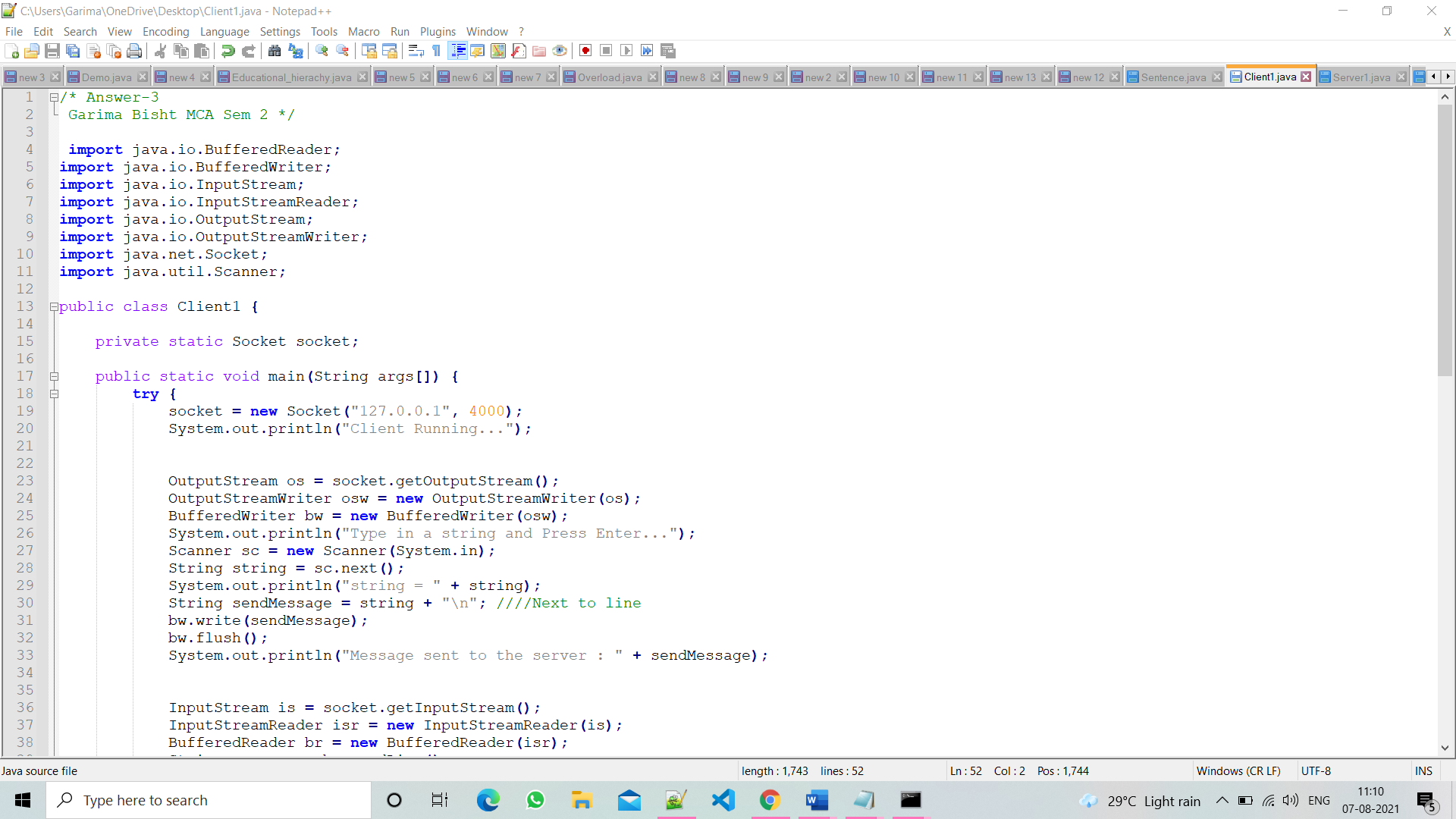
}

}

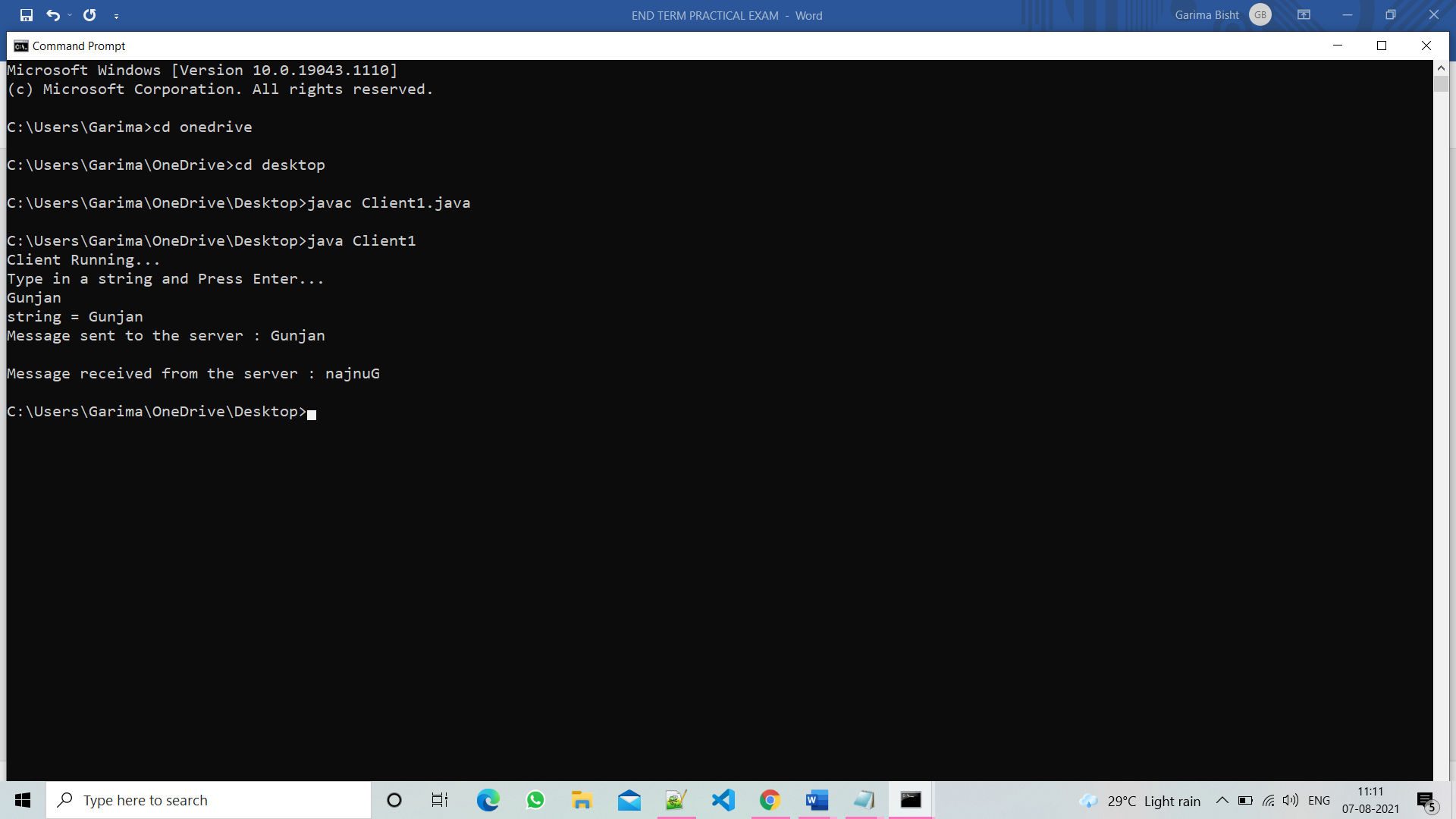
}

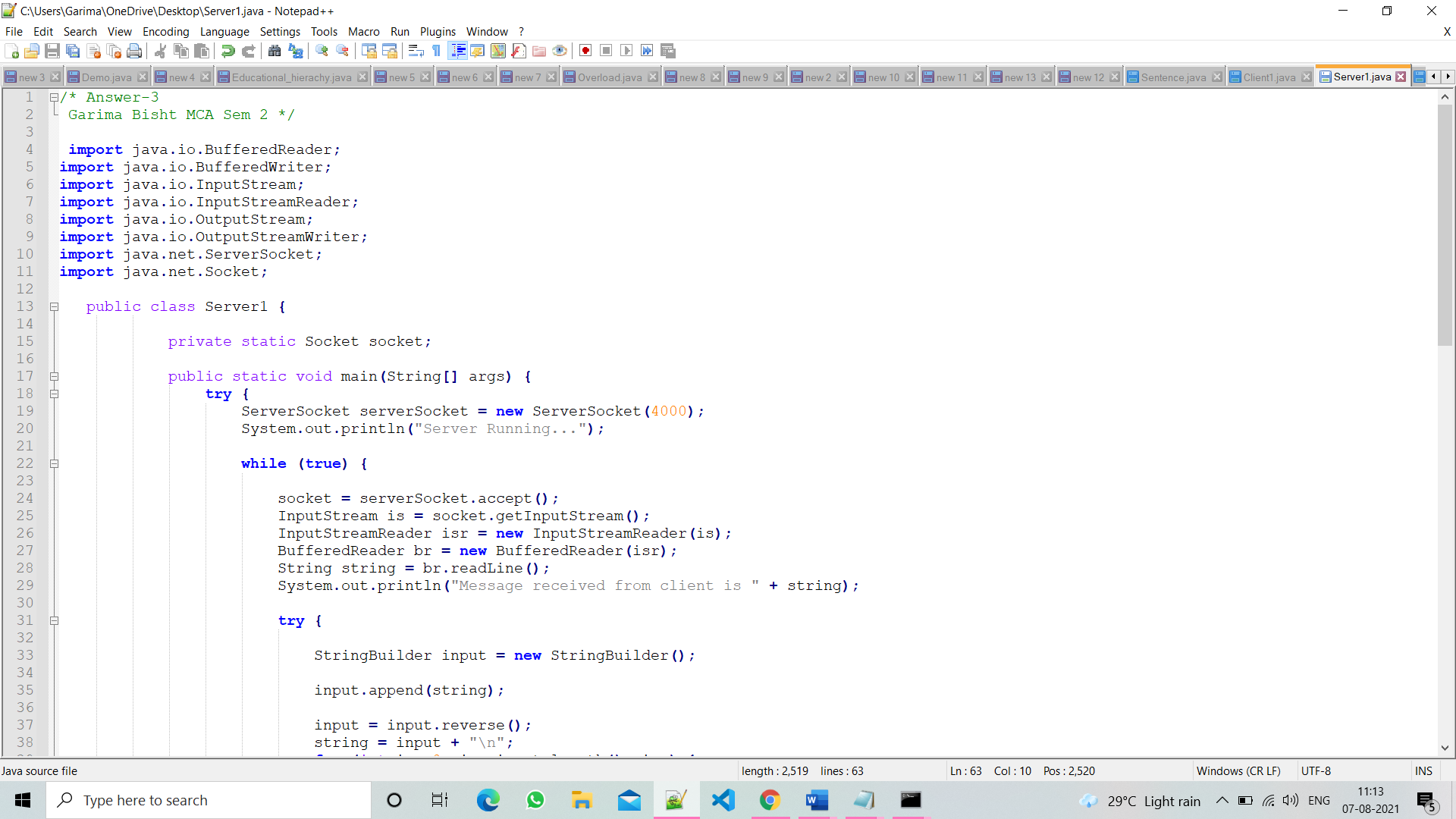
}

**Output**



**Client Running**





**Server Running**

