**Cloud Computing Practical 1 using TCP**

Write a client program to enter the number and server program to

calculate the square, square root, cube and cube root of the entered

number using TCP Communication.

**squarecubeclient.java**

import java.net.\*;

import java.io.\*;

import java.util.Scanner;

class squarecubeclient {

public static void main(String args[]) {

try {

Socket cs = new Socket("LocalHost", 8001);

BufferedReader infu = new BufferedReader(new InputStreamReader(System.in));

System.out.println("Enter a number : ");

int a = Integer.parseInt(infu.readLine());

DataOutputStream out = new DataOutputStream(cs.getOutputStream());

out.writeInt(a);

DataInputStream in = new DataInputStream(cs.getInputStream());

System.out.println(in.readUTF());

System.out.println(in.readUTF());

System.out.println(in.readUTF());

System.out.println(in.readUTF());

cs.close();

} catch (Exception e) {

System.out.println(e.toString());

}

}

}

**squarecubeserver.java**

import java.net.\*;

import java.io.\*;

import java.util.Scanner;

class squarecubeserver {

public static void main(String args[]) {

try {

ServerSocket ss = new ServerSocket(8001);

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

Socket s = ss.accept();

DataInputStream in = new DataInputStream(s.getInputStream());

int x = in.readInt();

DataOutputStream otc = new DataOutputStream(s.getOutputStream());

int y = x;

int sqr = x \* x;

int cube = y \* y \* y;

double sqrtValue = Math.sqrt(x);

double cbrtValue = Math.cbrt(y);

otc.writeUTF("Square root of " + x + " is = " + sqrtValue);

otc.writeUTF("Square of " + x + " is = " + sqr);

otc.writeUTF("Cube root of " + y + " is = " + cbrtValue);

otc.writeUTF("Cube of " + y + " is = " + cube);

} catch (Exception e) {

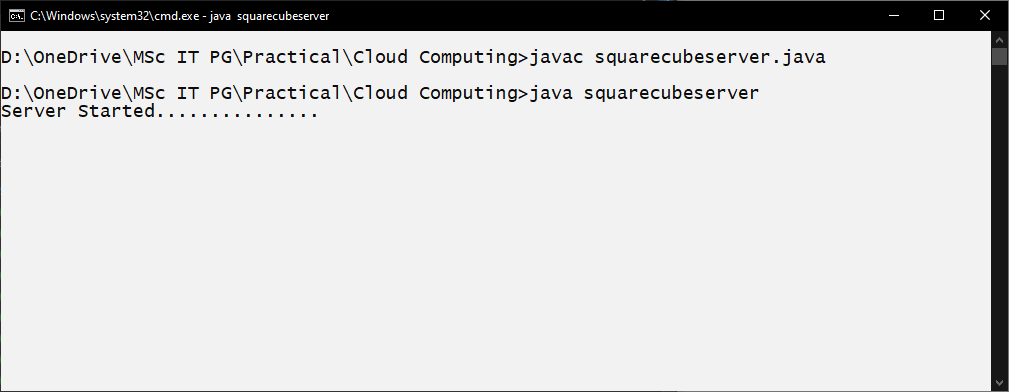
System.out.println(e.toString());

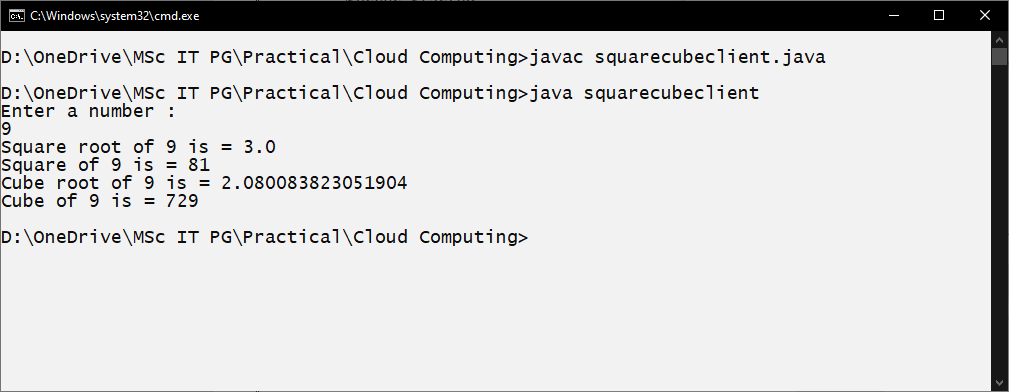
}

}

}

**Output:**





Write a client program to enter the number and server program to check

whether the number is prime or not using TCP Communication.

**tcpclientprime.java**

/\*Client program to check prime or not \*/

import java.net.\*;

import java.io.\*;

class tcpClientPrime {

public static void main(String args[]) {

try {

Socket cs = new Socket("LocalHost", 8001);

BufferedReader infu = new BufferedReader(new InputStreamReader(System.in));

System.out.println("Enter a number : ");

int a = Integer.parseInt(infu.readLine());

DataOutputStream out = new DataOutputStream(cs.getOutputStream());

out.writeInt(a);

DataInputStream in = new DataInputStream(cs.getInputStream());

System.out.println(in.readUTF());

cs.close();

} catch (Exception e) {

System.out.println(e.toString());

}

}

}

**tcpserverprime.java**

/\* Server program to check given no is prime or not in response to client request \*/

import java.net.\*;

import java.io.\*;

class tcpServerPrime {

public static void main(String args[]) {

try {

ServerSocket ss = new ServerSocket(8001);

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

Socket s = ss.accept();

DataInputStream in = new DataInputStream(s.getInputStream());

int x = in.readInt();

DataOutputStream otc = new DataOutputStream(s.getOutputStream());

int y = x / 2;

if (x == 1 || x == 2 || x == 3) {

otc.writeUTF(x + "is Prime");

System.exit(0);

}

for (int i = 2; i <= y; i++) {

if (x % i != 0) {

otc.writeUTF(x + " is Prime");

} else {

otc.writeUTF(x + " is not Prime");

}

}

} catch (Exception e) {

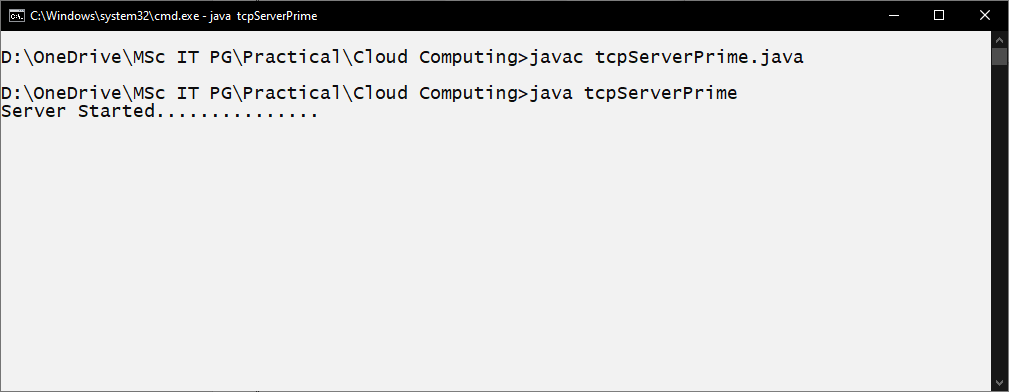
System.out.println(e.toString());

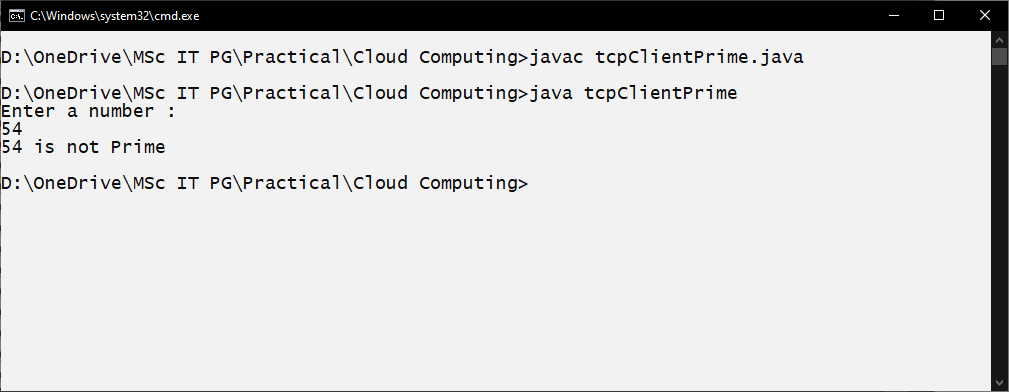
}

}

}

**Output:**





Program to implement chat client and Server.

**chatclient.java**

/\*Program to implement chat client \*/

import java.net.\*;

import java.io.\*;

class Chatclient {

public static void main(String args[]) {

try {

Socket s = new Socket("Localhost", 8000);

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

DataOutputStream out = new DataOutputStream(s.getOutputStream());

BufferedReader in = new BufferedReader(new InputStreamReader(s.getInputStream()));

String msg;

System.out.println("To stop chatting with server type STOP");

System.out.print("Client Says: ");

while ((msg = br.readLine()) != null) {

out.writeBytes(msg + "\n");

if (msg.equals("STOP")) {

break;

}

System.out.println("Server Says : " + in.readLine());

System.out.print("Client Says : ");

}

br.close();

in.close();

out.close();

s.close();

} catch (Exception e) {

e.printStackTrace();

}

}

}

**chatserver.java**

/\* Program to implement chat server \*/

import java.net.\*;

import java.io.\*;

class Chatserver {

public static void main(String args[]) {

try {

ServerSocket ss = new ServerSocket(8000);

System.out.println("Waiting for client to connect..");

Socket s = ss.accept();

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

DataOutputStream out = new DataOutputStream(s.getOutputStream());

BufferedReader in = new BufferedReader(new InputStreamReader(s.getInputStream()));

String receive, send;

while ((receive = in.readLine()) != null) {

if (receive.equals("STOP"))

break;

System.out.println("Client Says : " + receive);

System.out.print("Server Says : ");

send = br.readLine();

out.writeBytes(send + "\n");

}

br.close();

in.close();

out.close();

s.close();

} catch (Exception e) {

e.printStackTrace();

}

}

}

**Output:**

