**Practical 1.6**

**Aim:**

Write a java program using java.net library. Client Write a client program to send any string from its standard input to the server program. The server program reads the string, finds number of digits and characters and sends it back to client program. Use connection oriented and connection-less communication.

**PROGRAM CODE:**

**//TCP Client**

import java.io.\*;

import java.net.\*;

public class countclient {

public static void main(String[] args) throws IOException {

String serverHostname = new String ("127.0.0.1");

if (args.length > 0)

serverHostname = args[0];

System.out.println ("Attemping to connect to host " +

serverHostname + " on port 10007.");

Socket echoSocket = null;

PrintWriter out = null;

BufferedReader in = null;

try {

// echoSocket = new Socket("taranis", 7);

echoSocket = new Socket(serverHostname, 10007);

out = new PrintWriter(echoSocket.getOutputStream(), true);

in = new BufferedReader(new InputStreamReader(

echoSocket.getInputStream()));

}

catch (UnknownHostException e) {

System.err.println("Don't know about host: " + serverHostname);

System.exit(1);

}

catch (IOException e) {

System.err.println("Couldn't get I/O for "

+ "the connection to: " + serverHostname);

e.printStackTrace();

System.exit(1);

}

BufferedReader stdIn = new BufferedReader(

new InputStreamReader(System.in));

String userInput;

System.out.print ("input: ");

userInput = stdIn.readLine();

out.println(userInput);

System.out.println("output: " + in.readLine**());**

out.close();

in.close();

stdIn.close();

echoSocket.close();

}

}

**//TCP Server**

import java.net.\*;

import java.io.\*;

public class countserver

{

public static void main(String[] args) throws IOException

{

ServerSocket serverSocket = null;

try {

serverSocket = new ServerSocket(10007);

}

catch (IOException e)

{

System.err.println("Could not listen on port: 10007.");

System.exit(1);

}

Socket clientSocket = null;

System.out.println ("Waiting for connection.....");

try {

clientSocket = serverSocket.accept();

}

catch (IOException e)

{

System.err.println("Accept failed.");

System.exit(1);

}

System.out.println ("Connection successful");

System.out.println ("Waiting for input.....");

PrintWriter out = new PrintWriter(clientSocket.getOutputStream(), true);

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

String inputLine;

while ((inputLine = in.readLine()) != null)

{

System.out.println ("Server: " + inputLine);

int i=0,d=0,c=0;

while(i<=inputLine.length()-1){

if (Character.isDigit(inputLine.charAt(i))) {

d++;

}

else if (Character.isLetter(inputLine.charAt(i))) {

c++;

}

i++;

}

String ans="digit="+d+"letter="+c;

out.println(ans);

//if (inputLine.equals("Bye."))

// break;

}

out.close();

in.close();

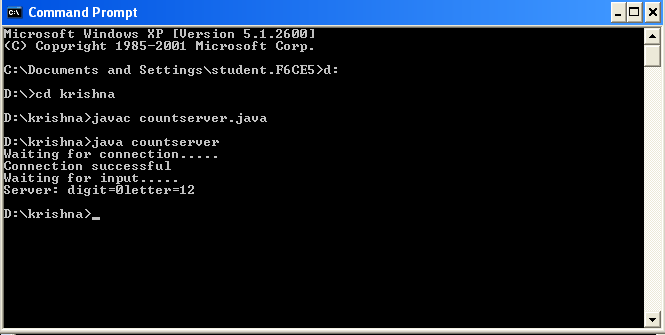
clientSocket.close();

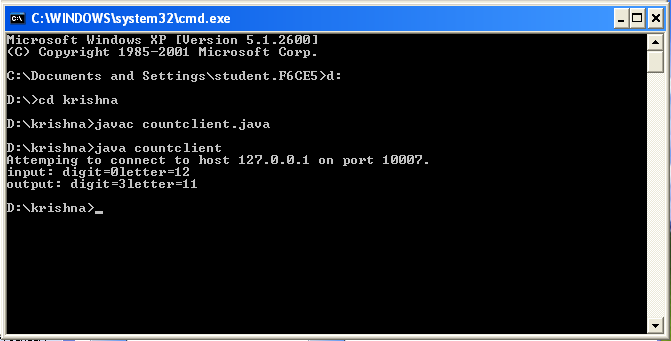
serverSocket.close();

}

}

**Input Output:**

****

****