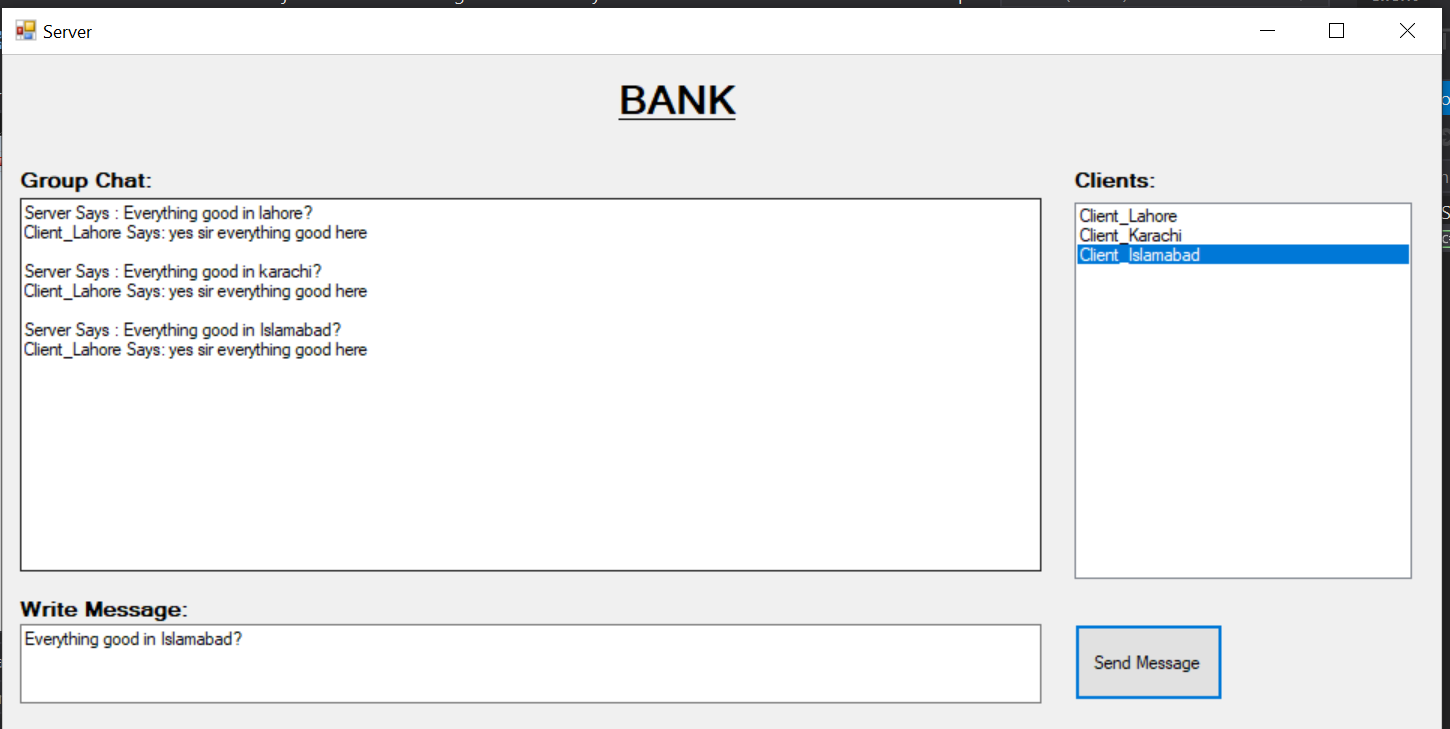
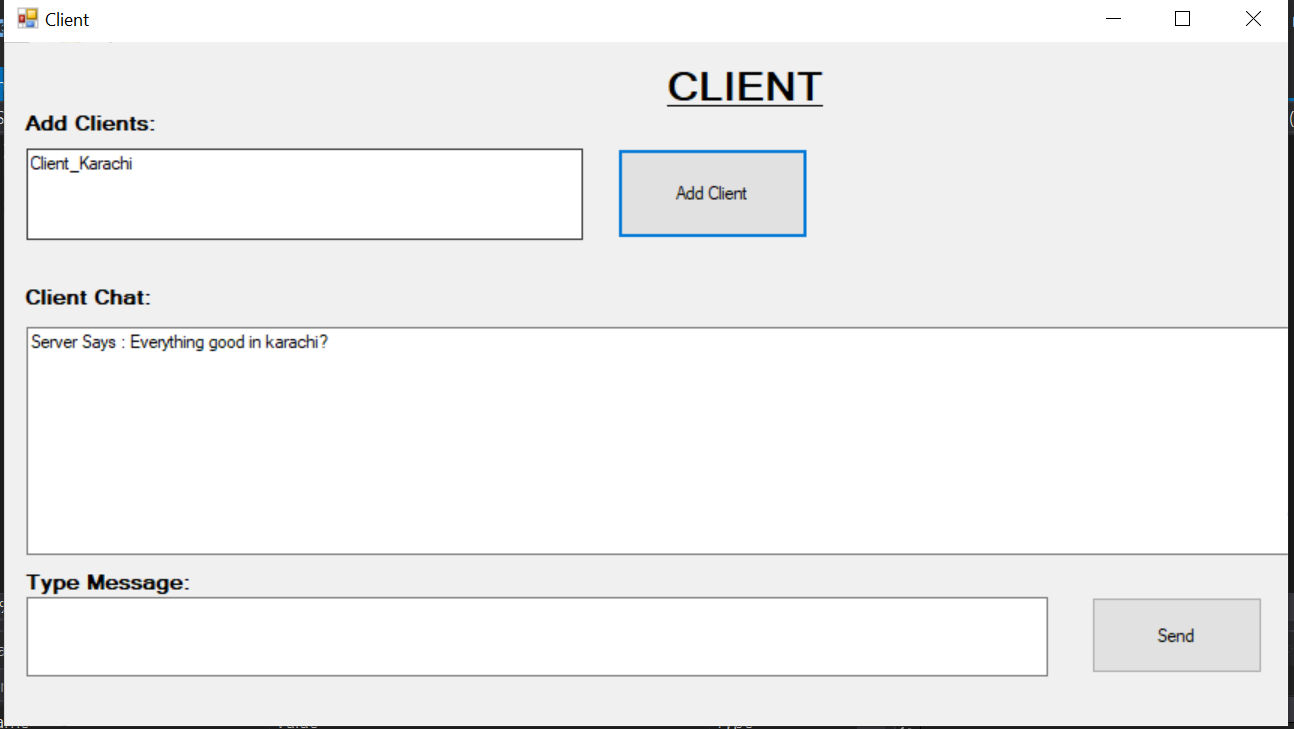
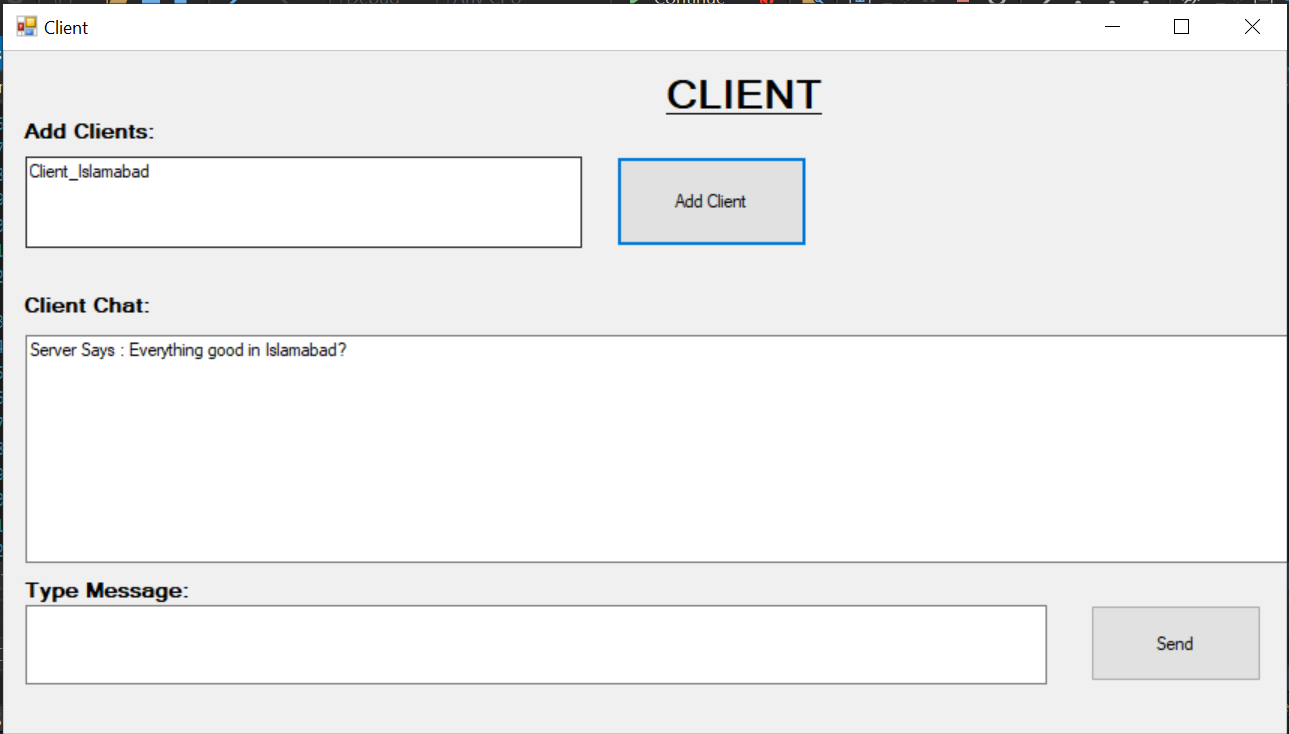
QUESTION NO#03 (A):

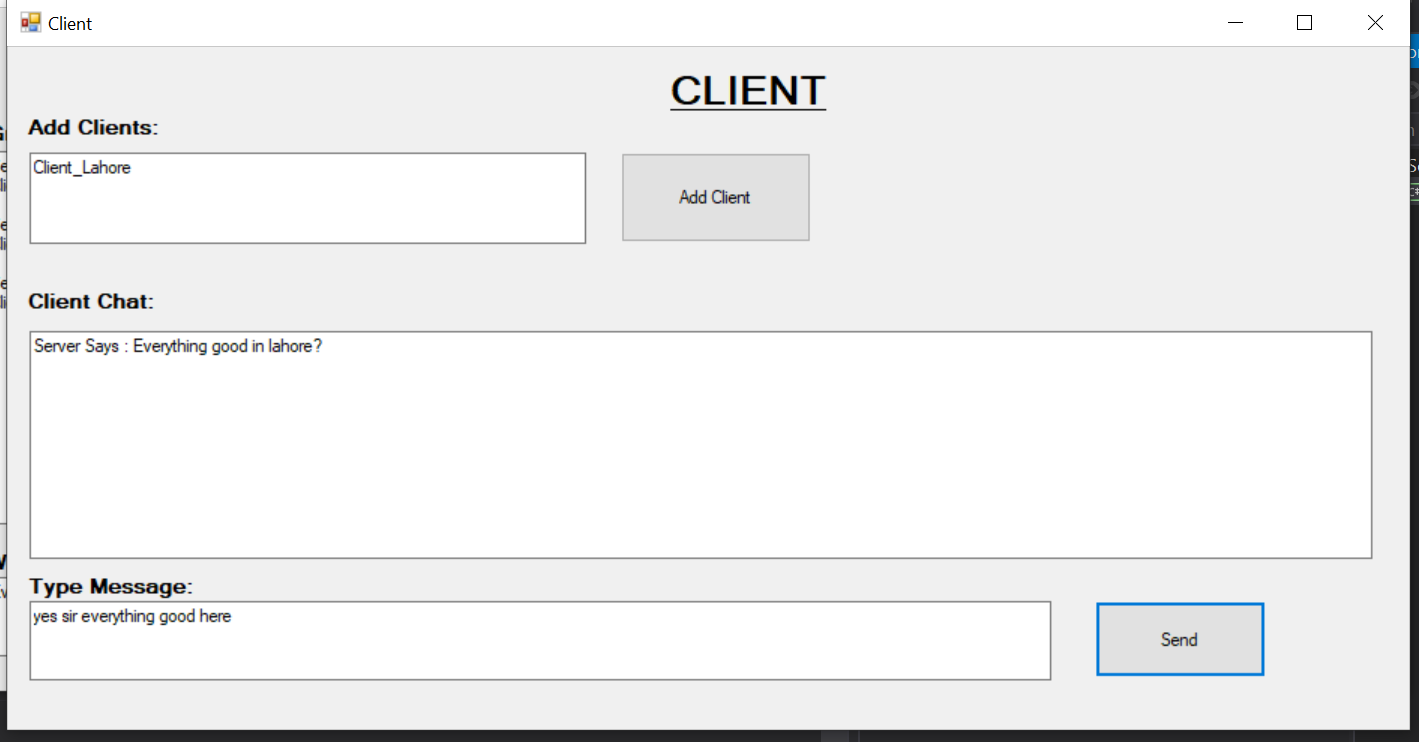
**SERVER OUTPUT:**

****

**CLIENT OUTPUT:**







**SERVER CODE:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Threading;

using System.Net;

using System.Net.Sockets;

using System.IO;

namespace Server

{

    public partial class Form1 : Form

    {

        public Form1()

        {

            InitializeComponent();

        }

        private void Form1\_Load(object sender, EventArgs e)

        {

            CheckForIllegalCrossThreadCalls = false;

            TcpListener listener = new TcpListener(IPAddress.Loopback, 11000);

            listener.Start(10);

            listener.BeginAcceptTcpClient(new AsyncCallback(ClientConnect), listener);

        }

        Dictionary<string,TcpClient> lstClients = new Dictionary<string,TcpClient>();

        byte[] b = new byte[1024];

        private void ClientConnect(IAsyncResult ar)

        {

            TcpListener listener = (TcpListener)ar.AsyncState;

            TcpClient client = listener.EndAcceptTcpClient(ar);

            NetworkStream ns = client.GetStream();

            object[] a = new object[2];

            a[0] = ns;

            a[1] = client;

            ns.BeginRead(b, 0, b.Length, new AsyncCallback(ReadMsg), a);

            listener.BeginAcceptTcpClient(new AsyncCallback(ClientConnect), listener);

        }

        private void ReadMsg(IAsyncResult ar)

        {

            object[] a = (object[])ar.AsyncState;

            NetworkStream ns = (NetworkStream)a[0];

            TcpClient client = (TcpClient)a[1];

            int count = ns.EndRead(ar);

            string msg = ASCIIEncoding.ASCII.GetString(b, 0, count);

            if (msg.Contains("@name@"))

            {

                string name = msg.Replace("@name@", "");

                lstClients.Add(name, client);

                lstbxClients.Items.Add(name);

            }

            else

            {

                txtDisplay.Text += msg + Environment.NewLine;

            }

            ns.BeginRead(b, 0, b.Length, new AsyncCallback(ReadMsg), a);

        }

        private void button1\_Click(object sender, EventArgs e)

        {

            TcpClient client = (TcpClient)lstClients[lstbxClients.SelectedItem.ToString()];

            NetworkStream ns = client.GetStream();

            StreamWriter sw = new StreamWriter(ns);

            string textToSend = "Server Says : " + txtMsg.Text;

            sw.WriteLine(textToSend);

            txtDisplay.Text += textToSend + Environment.NewLine;

            sw.Flush();

        }

    }

}

**CLIENT CODE:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Threading;

using System.Net;

using System.Net.Sockets;

using System.IO;

namespace Client

{

    public partial class Form1 : Form

    {

        public Form1()

        {

            InitializeComponent();

        }

        byte[] b = new byte[1024];

        TcpClient client = new TcpClient();

        private void Form1\_Load(object sender, EventArgs e)

        {

        }

        private void ReadMsg(IAsyncResult ar)

        {

            NetworkStream ns = (NetworkStream)ar.AsyncState;

            int count = ns.EndRead(ar);

            txtDisplay.Text += ASCIIEncoding.ASCII.GetString(b, 0, count);

            ns.BeginRead(b, 0, b.Length, ReadMsg, ns);

        }

        private void button1\_Click(object sender, EventArgs e)

        {

            NetworkStream ns = client.GetStream();

            StreamWriter sw = new StreamWriter(ns);

            sw.WriteLine(txtName.Text + " Says: " + txtMsg.Text);

            sw.Flush();

        }

        private void button2\_Click(object sender, EventArgs e)

        {

            CheckForIllegalCrossThreadCalls = false;

            client.Connect(IPAddress.Loopback, 11000);

            NetworkStream ns = client.GetStream();

            StreamWriter sw = new StreamWriter(ns);

            sw.WriteLine("@name@" + txtName.Text);

            sw.Flush();

            ns.BeginRead(b, 0, b.Length, ReadMsg, ns);

        }

    }

}