



Your Company Name

Tel: +44 1234 567 9898

Fax: +44 1234 545 9999

email: info@@company.com

Microsoft Visual Studio

C# Project

Source Code Output

Created using

VScodePrint

Macro Variables Substitution Example

General Date	24/01/2017 6:37:36 PM
Long Date	Tuesday, 24 January 2017
Short Date	24/01/2017
Long Time	6:37:36 PM
Short Time	18:37
Today	24/01/2017 6:37:36 PM
Project Name	ServerApp

Prepared by

Joginder S Nahil

on

24/01/2017 6:37:36 PM

WWW.STARPRINTTOOLS.COM

ClientApp	3
ClientApp	3
My Project	3
AssemblyInfo.vb	3
Form1.vb	4
ClientApp	4
Form1	4
New	4
RtbClientKeyDown	4
SendMessage	4
Program.vb	5
ClientApp	5
Program	5
New	5
Main	5
ServerApp	7
My Project	7
AssemblyInfo.vb	7
Form1.vb	8
ServerApp	8
Form1	8
New	8
Server	8
ListenForClients	8
HandleClientComm	8
WriteMessage	9
Echo	9
Program.vb	10
ServerApp	10
Program	10
New	10
Main	10

```
1 Imports System.Reflection
2 Imports System.Runtime.CompilerServices
3 Imports System.Runtime.InteropServices
4
5 ' General Information about an assembly is controlled through the following
6 ' set of attributes. Change these attribute values to modify the information
7 ' associated with an assembly.
8 <Assembly:AssemblyTitle("ClientApp" )>
9 <Assembly:AssemblyDescription("")>
10 <Assembly:AssemblyConfiguration("")>
11 <Assembly:AssemblyCompany("")>
12 <Assembly:AssemblyProduct("ClientApp" )>
13 <Assembly:AssemblyCopyright("Copyright © 2013" )>
14 <Assembly:AssemblyTrademark("")>
15 <Assembly:AssemblyCulture("")>
16
17 ' Setting ComVisible to false makes the types in this assembly not visible
18 ' to COM components. If you need to access a type in this assembly from
19 ' COM, set the ComVisible attribute to true on that type.
20 <Assembly:ComVisible(False)>
21
22 ' The following GUID is for the ID of the typelib if this project is exposed
23 ' to COM
24 <Assembly:Guid("3b710eb5-e17c-4521-b6e5-e76f91a19958" )>
25
26 ' Version information for an assembly consists of the following four values:
27 ' Major Version
28 ' Minor Version
29 ' Build Number
30 ' Revision
31
32 ' You can specify all the values or you can default the Build and Revision
33 ' Numbers
34 ' by using the '*' as shown below:
35 ' [assembly: AssemblyVersion("1.0.*")]
36 <Assembly:AssemblyVersion("1.0.0.0" )>
   <Assembly:AssemblyFileVersion("1.0.0.0" )>
```

```

1 Imports System
2 Imports System.Collections.Generic
3 Imports System.ComponentModel
4 Imports System.Data
5 Imports System.Drawing
6 Imports System.Linq
7 Imports System.Net
8 Imports System.Net.Sockets
9 Imports System.Text
10 Imports System.Windows.Forms
11
12 Namespace ClientApp
13     Partial Public Class Form1
14         Inherits Form
15
16         Private myMessage As String = ""
17         Private client As New TcpClient()
18         Private serverEndPoint As New IPEndPoint(IPAddress.Parse("127.0.0.1"),
19             3000)
20
21         Public Sub New()
22             InitializeComponent()
23             client.Connect(serverEndPoint)
24         End Sub
25
26         Private Sub RtbClientKeyDown(ByVal sender As Object, ByVal e As
27             KeyEventArgs) Handles rtbClient.KeyDown
28             If e.KeyData <> Keys.Enter OrElse e.KeyData <> Keys.Return Then
29                 myMessage &= ChrW(e.KeyValue)
30             Else
31                 SendMessage(myMessage)
32                 myMessage = ""
33             End If
34         End Sub
35
36         Private Sub SendMessage(ByVal msg As String)
37             Dim clientStream As NetworkStream = client.GetStream()
38
39             Dim encoder As New ASCIIEncoding()
40             Dim buffer() As Byte = encoder.GetBytes(msg)
41
42             clientStream.Write(buffer, 0, buffer.Length)
43             clientStream.Flush()
44
45             ' Receive the TcpServer.response.
46
47             ' Buffer to store the response bytes.
48             Dim data(255) As Byte
49
50             ' String to store the response ASCII representation.
51             Dim responseData As String = String.Empty
52
53             ' Read the first batch of the TcpServer response bytes.
54             Dim bytes As Int32 = clientStream.Read(data, 0, data.Length)
55             responseData = System.Text.Encoding.ASCII.GetString(data, 0, bytes)
56
57             rtbClient.AppendText(Environment.NewLine & "From Server: " &
58                 responseData)
59         End Sub
60     End Class
61 End Namespace
62

```

```
1 Imports System
2 Imports System.Collections.Generic
3 Imports System.Linq
4 Imports System.Windows.Forms
5
6 Namespace ClientApp
7     Friend NotInheritable Class Program
8
9         Private Sub New()
10            End Sub
11
12         ''' <summary>
13         ''' The main entry point for the application.
14         ''' </summary>
15         <STAThread> _
16         Shared Sub Main()
17             Application.EnableVisualStyles()
18             Application.SetCompatibleTextRenderingDefault(False)
19             Application.Run(New Form1())
20         End Sub
21     End Class
22 End Namespace
```

- A**
AppendText, 4
Application, 5
ASCII, 4
ASCIIEncoding, 4
AssemblyCompany, 3
AssemblyConfiguration, 3
AssemblyCopyright, 3
AssemblyCulture, 3
AssemblyDescription, 3
AssemblyFileVersion, 3
AssemblyProduct, 3
AssemblyTitle, 3
AssemblyTrademark, 3
AssemblyVersion, 3
- B**
buffer, 4
bytes, 4
- C**
ChrW, 4
client, 4
ClientApp, 4, 5
clientStream, 4
Collections, 4, 5
CompilerServices, 3
ComponentModel, 4
ComVisible, 3
Connect, 4
- D**
Data, 4
data, 4
Drawing, 4
- E**
e, 4
Empty, 4
EnableVisualStyles, 5
encoder, 4
Encoding, 4
Enter, 4
Environment, 4
- F**
Flush, 4
Form, 4
Form1, 4, 5
Forms, 4, 5
- G**
Generic, 4, 5
GetBytes, 4
GetStream, 4
GetString, 4
Guid, 3
- I**
InitializeComponent, 4
Int32, 4
InteropServices, 3
IPAddress, 4
IPEndPoint, 4
- K**
KeyData, 4
KeyDown, 4
KeyEventArgs, 4
Keys, 4
KeyValue, 4
- L**
Length, 4
Linq, 4, 5
- M**
Main, 5
msg, 4
myMessage, 4
- N**
Net, 4
NetworkStream, 4
NewLine, 4
- P**
Parse, 4
Program, 5
- R**
Read, 4
Reflection, 3
responseData, 4
rtbClient, 4
RtbClientKeyDown, 4
Run, 5
Runtime, 3
- S**
sender, 4
SendMessage, 4
serverEndPoint, 4
SetCompatibleTextRenderingDefault, 5
Sockets, 4
STAThread, 5
System, 3-5
- T**
TcpClient, 4
Text, 4
- W**
Windows, 4, 5
Write, 4

```
1 Imports System.Reflection
2 Imports System.Runtime.CompilerServices
3 Imports System.Runtime.InteropServices
4
5 ' General Information about an assembly is controlled through the following
6 ' set of attributes. Change these attribute values to modify the information
7 ' associated with an assembly.
8 <Assembly:AssemblyTitle("ServerApp" )>
9 <Assembly:AssemblyDescription("")>
10 <Assembly:AssemblyConfiguration("")>
11 <Assembly:AssemblyCompany("")>
12 <Assembly:AssemblyProduct("ServerApp" )>
13 <Assembly:AssemblyCopyright("Copyright © 2013" )>
14 <Assembly:AssemblyTrademark("")>
15 <Assembly:AssemblyCulture("")>
16
17 ' Setting ComVisible to false makes the types in this assembly not visible
18 ' to COM components. If you need to access a type in this assembly from
19 ' COM, set the ComVisible attribute to true on that type.
20 <Assembly:ComVisible(False)>
21
22 ' The following GUID is for the ID of the typelib if this project is exposed
23 ' to COM
24 <Assembly:Guid("008241de-550f-4d70-adea-a0f346f20e56" )>
25
26 ' Version information for an assembly consists of the following four values:
27 ' Major Version
28 ' Minor Version
29 ' Build Number
30 ' Revision
31
32 ' You can specify all the values or you can default the Build and Revision
33 ' Numbers
34 ' by using the '*' as shown below:
35 ' [assembly: AssemblyVersion("1.0.*")]
36 <Assembly:AssemblyVersion("1.0.0.0" )>
   <Assembly:AssemblyFileVersion("1.0.0.0" )>
```

```

1 Imports System
2 Imports System.Collections.Generic
3 Imports System.ComponentModel
4 Imports System.Data
5 Imports System.Drawing
6 Imports System.Linq
7 Imports System.Net
8 Imports System.Net.Sockets
9 Imports System.Text
10 Imports System.Threading
11 Imports System.Windows.Forms
12
13 Namespace ServerApp
14     Partial Public Class Form1
15         Inherits Form
16
17         Private tcpListener As TcpListener
18         Private listenThread As Thread
19         Private connectedClients As Integer = 0
20         Private Delegate Sub WriteMessageDelegate(ByVal msg As String)
21
22         Public Sub New()
23             InitializeComponent()
24             Server()
25         End Sub
26
27         Private Sub Server()
28             Me.tcpListener = New TcpListener(IPAddress.Loopback, 3000) ' Change
29             Me.listenThread = New Thread(New ThreadStart(AddressOf ListenForClients))
30             Me.listenThread.Start()
31         End Sub
32
33         Private Sub ListenForClients()
34             Me.tcpListener.Start()
35
36             'blocks until a client has connected to the server
37             Do ' Never ends until the Server is closed.
38                 Dim client As TcpClient = Me.tcpListener.AcceptTcpClient()
39
40                 'create a thread to handle communication
41                 'with connected client
42                 connectedClients += 1 ' Increment the number of clients that have
43                 lblNumberOfConnections.Text = connectedClients.ToString()
44
45                 Dim clientThread As New Thread(New ParameterizedThreadStart(
46                     AddressOf HandleClientComm))
47                 clientThread.Start(client)
48             Loop
49         End Sub
50
51         Private Sub HandleClientComm(ByVal client As Object)
52             Dim tcpClient As TcpClient = DirectCast(client, TcpClient)
53             Dim clientStream As NetworkStream = tcpClient.GetStream()
54
55             Dim message(4095) As Byte
56             Dim bytesRead As Integer
57
58             Do
59                 bytesRead = 0
60
61                 Try
62                     'blocks until a client sends a message
63                     bytesRead = clientStream.Read(message, 0, 4096)
64                 Catch
65                     'a socket error has occurred
66                     Exit Do
67                 End Try

```



```

1 2 3 4
68 If bytesRead = 0 Then
69     'the client has disconnected from the server
70     connectedClients -= 1
71     lblNumberOfConnections.Text = connectedClients.ToString()
72     Exit Do
73 End If
74
75 'message has successfully been received
76 Dim encoder As New ASCIIEncoding()
77
78 ' Convert the Bytes received to a string and display it on the
79 Server Screen
80 Dim msg As String = encoder.GetString(message, 0, bytesRead)
81 WriteMessage(msg)
82
83 ' Now Echo the message back
84 Echo(msg, encoder, clientStream)
85 Loop
86
87 tcpClient.Close()
88 End Sub
89
90 Private Sub WriteMessage(ByVal msg As String)
91     If Me.rtbServer.InvokeRequired Then
92         Dim d As New WriteMessageDelegate(AddressOf WriteMessage)
93         Me.rtbServer.Invoke(d, New Object() { msg })
94     Else
95         Me.rtbServer.AppendText(msg & Environment.NewLine)
96     End If
97 End Sub
98
99 ''' <summary>
100 ''' Echo the message back to the sending client
101 ''' </summary>
102 ''' <param name="msg">
103 ''' String: The Message to send back
104 ''' </param>
105 ''' <param name="encoder">
106 ''' Our ASCIIEncoder
107 ''' </param>
108 ''' <param name="clientStream">
109 ''' The Client to communicate to
110 ''' </param>
111 Private Sub Echo(ByVal msg As String, ByVal encoder As ASCIIEncoding,
112 ByVal clientStream As NetworkStream)
113     ' Now Echo the message back
114     Dim buffer() As Byte = encoder.GetBytes(msg)
115
116     clientStream.Write(buffer, 0, buffer.Length)
117     clientStream.Flush()
118 End Sub
119 End Class
End Namespace

```

```
1 Imports System
2 Imports System.Collections.Generic
3 Imports System.Linq
4 Imports System.Windows.Forms
5
6 Namespace ServerApp
7     Friend NotInheritable Class Program
8
9         Private Sub New()
10            End Sub
11
12         ''' <summary>
13         ''' The main entry point for the application.
14         ''' </summary>
15         <STAThread> _
16         Shared Sub Main()
17             Application.EnableVisualStyles()
18             Application.SetCompatibleTextRenderingDefault(False)
19             Application.Run(New Form1())
20         End Sub
21     End Class
22 End Namespace
```

- A**
AcceptTcpClient, 8
AppendText, 9
Application, 10
ASCIIEncoding, 9
AssemblyCompany, 7
AssemblyConfiguration, 7
AssemblyCopyright, 7
AssemblyCulture, 7
AssemblyDescription, 7
AssemblyFileVersion, 7
AssemblyProduct, 7
AssemblyTitle, 7
AssemblyTrademark, 7
AssemblyVersion, 7
- B**
buffer, 9
bytesRead, 8, 9
- C**
client, 8
clientStream, 8, 9
clientThread, 8
Close, 9
Collections, 8, 10
CompilerServices, 7
ComponentModel, 8
ComVisible, 7
connectedClients, 8, 9
- D**
d, 9
Data, 8
Drawing, 8
- E**
Echo, 9
EnableVisualStyles, 10
encoder, 9
Environment, 9
- F**
Flush, 9
Form, 8
Form1, 8, 10
Forms, 8, 10
- G**
Generic, 8, 10
GetBytes, 9
GetStream, 8
GetString, 9
Guid, 7
- H**
HandleClientComm, 8
- I**
InitializeComponent, 8
InteropServices, 7
Invoke, 9
InvokeRequired, 9
IPAddress, 8
- L**
lblNumberOfConnections, 8, 9
Length, 9
Linq, 8, 10
ListenForClients, 8
listenThread, 8
Loopback, 8
- M**
Main, 10
message, 8, 9
msg, 8, 9
- N**
Net, 8
NetworkStream, 8, 9
NewLine, 9
- P**
ParameterizedThreadStart, 8
Program, 10
- R**
Read, 8
Reflection, 7
rtbServer, 9
Run, 10
Runtime, 7
- S**
Server, 8
ServerApp, 8, 10
SetCompatibleTextRenderingDefault, 10
Sockets, 8
Start, 8
STAThread, 10
System, 7, 8, 10
- T**
tcpClient, 8, 9
TcpClient, 8
TcpListener, 8
tcpListener, 8
Text, 8, 9
Thread, 8
Threading, 8
ThreadStart, 8
ToString, 8, 9
- W**
Windows, 8, 10
Write, 9
WriteMessage, 9
WriteMessageDelegate, 8, 9