

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace IDatLibrary
{
    public class Class1
    {
    }
}

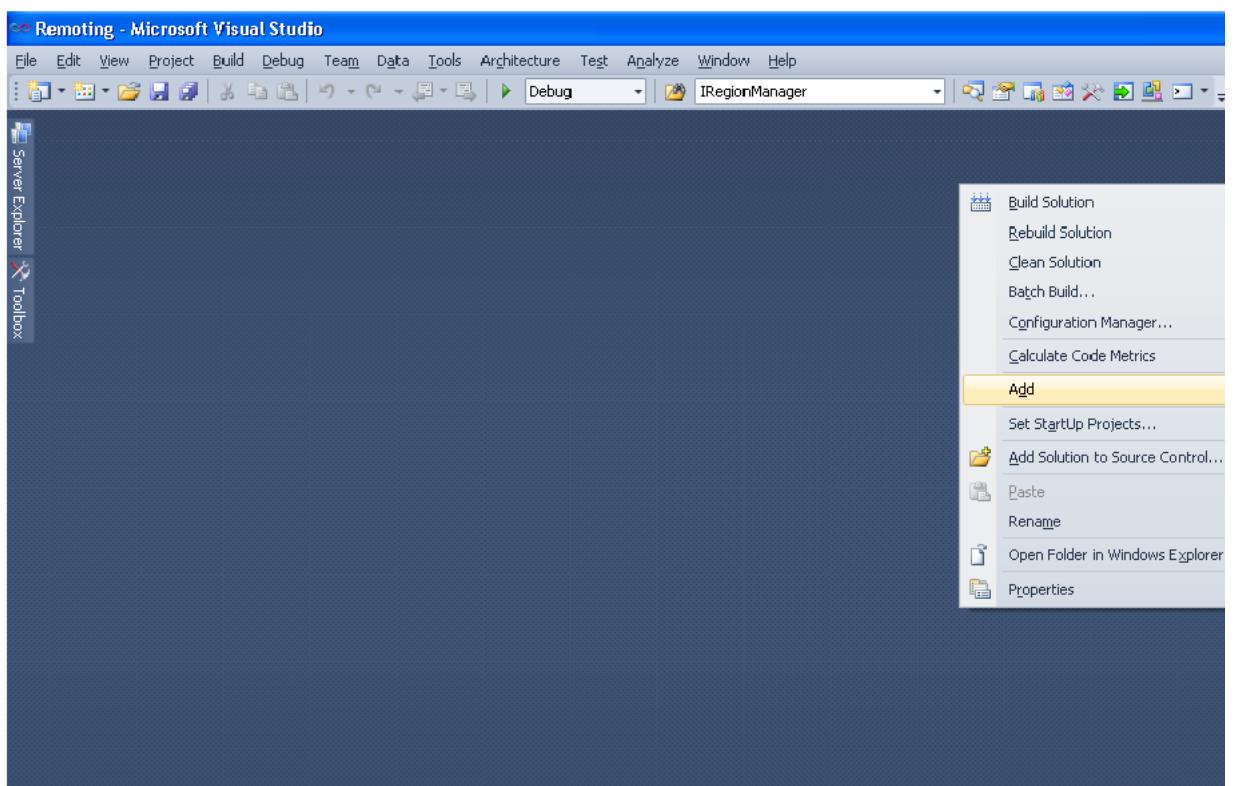
```

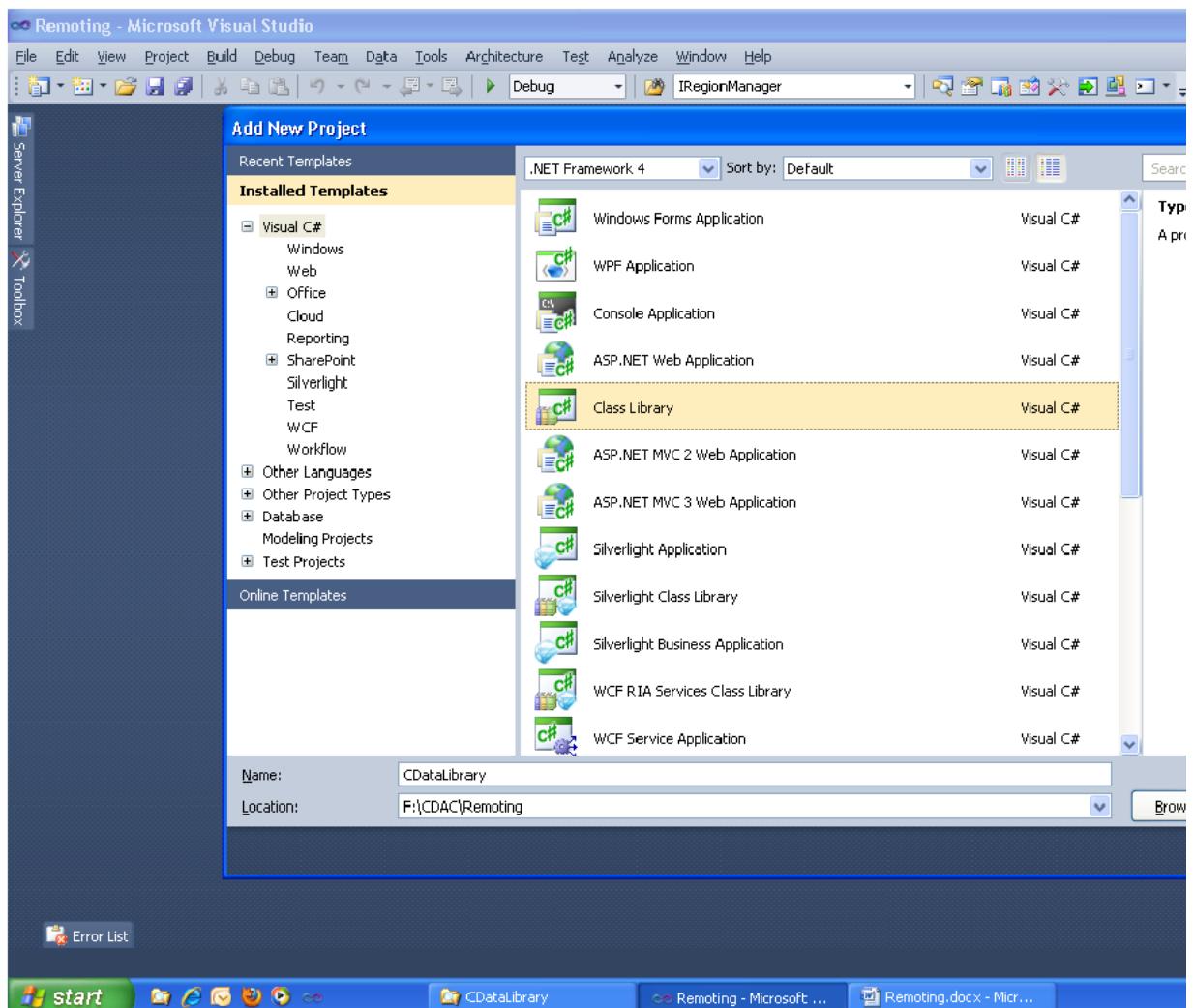
The screenshot shows the Microsoft Visual Studio IDE interface. The title bar reads "bonaventuresystems.com/dnet/cdac/05_Remoting.htm". The menu bar includes File, Edit, View, Refactor, Project, Build, Debug, Team, Data, Tools, Architecture, Test, Analyze, Window, Help, and Full Screen. The toolbar has icons for New, Open, Save, Print, and others. The main window displays the code for "IDataLibrary.cs":

```
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data;

namespace IDataLibrary
{
    public interface IData
    {
        int Add(int x, int y);
        DataSet GetData(string TableName, string ConnectionString);
    }
}
```

The code uses C# syntax with namespaces like System.Collections.Generic, System.Linq, System.Text, and System.Data. It defines a public interface IData with two methods: Add and GetData. The Add method takes two integers and returns an integer. The GetData method takes a string TableName and a string ConnectionString, and returns a DataSet.



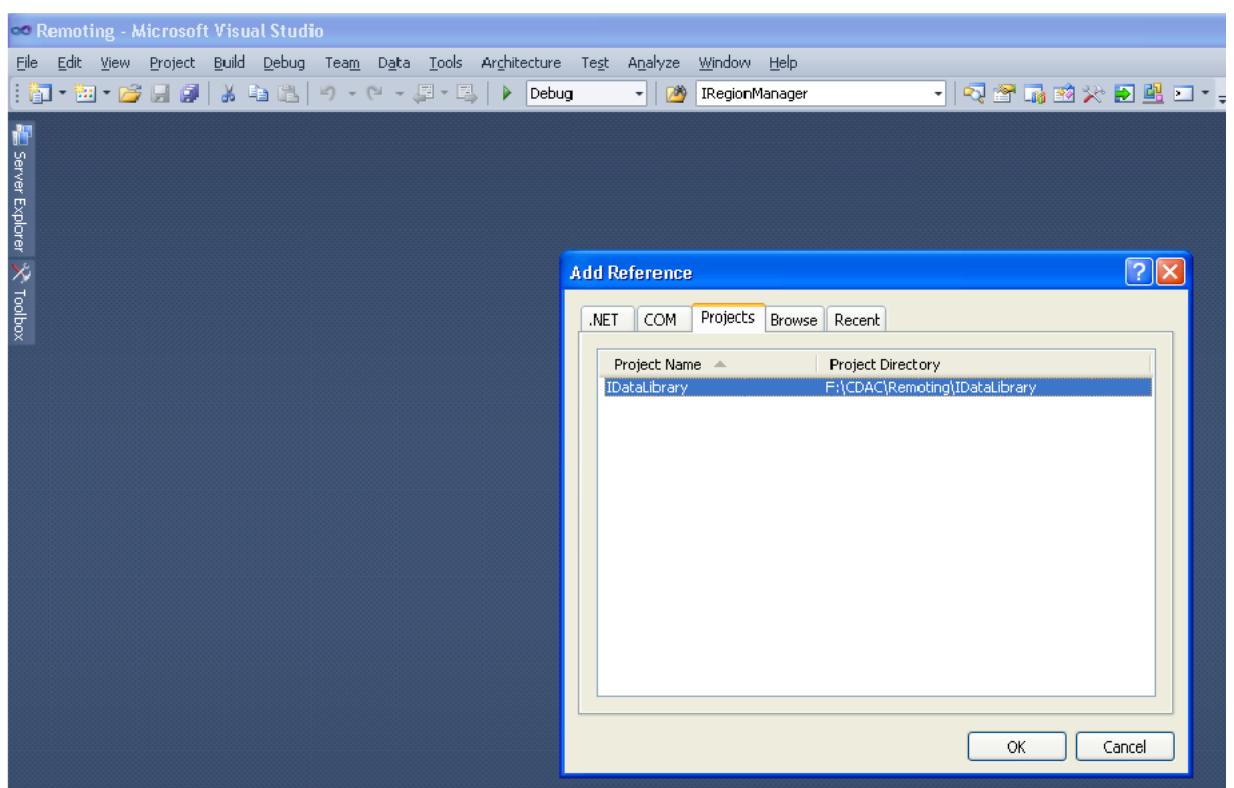
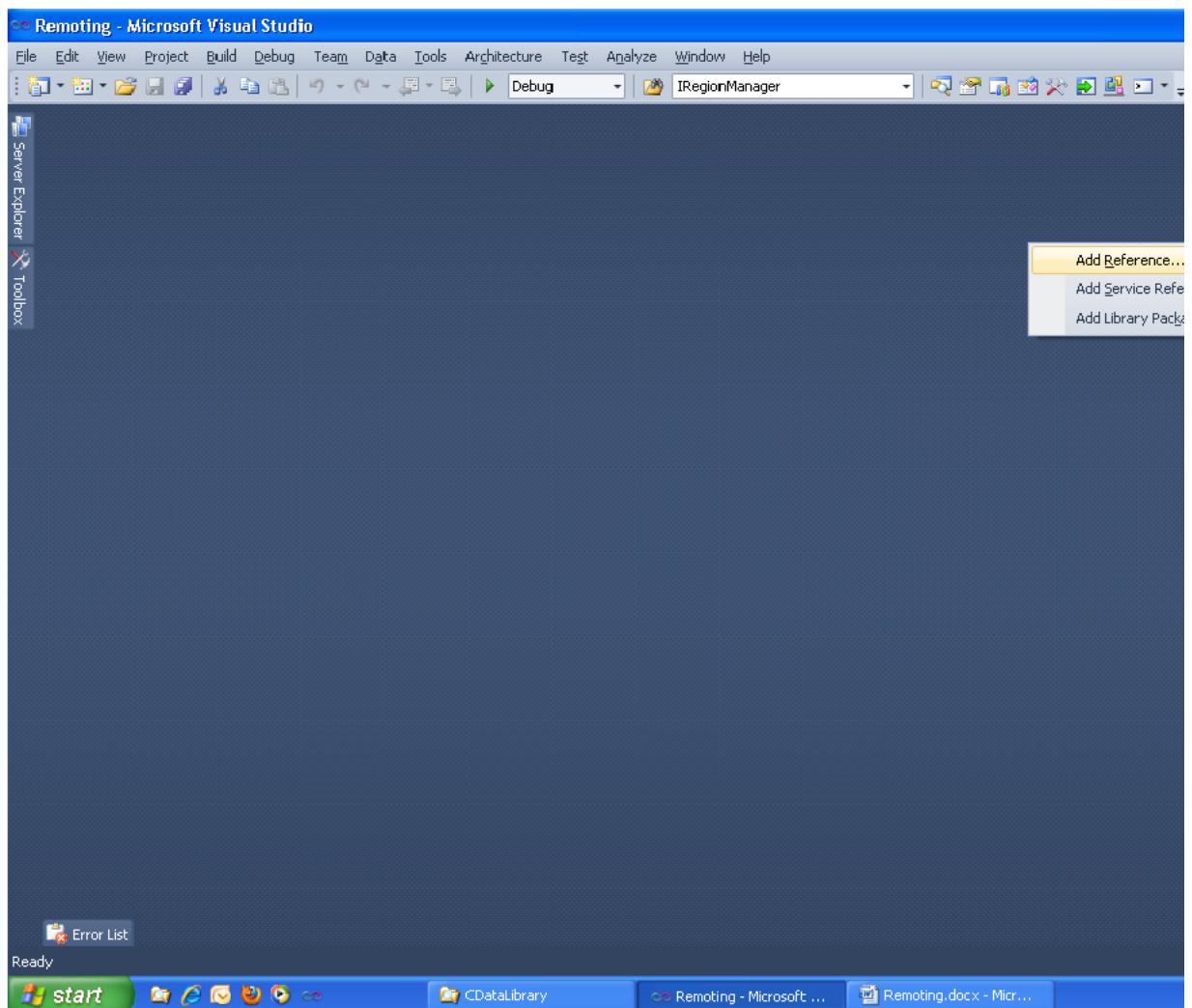


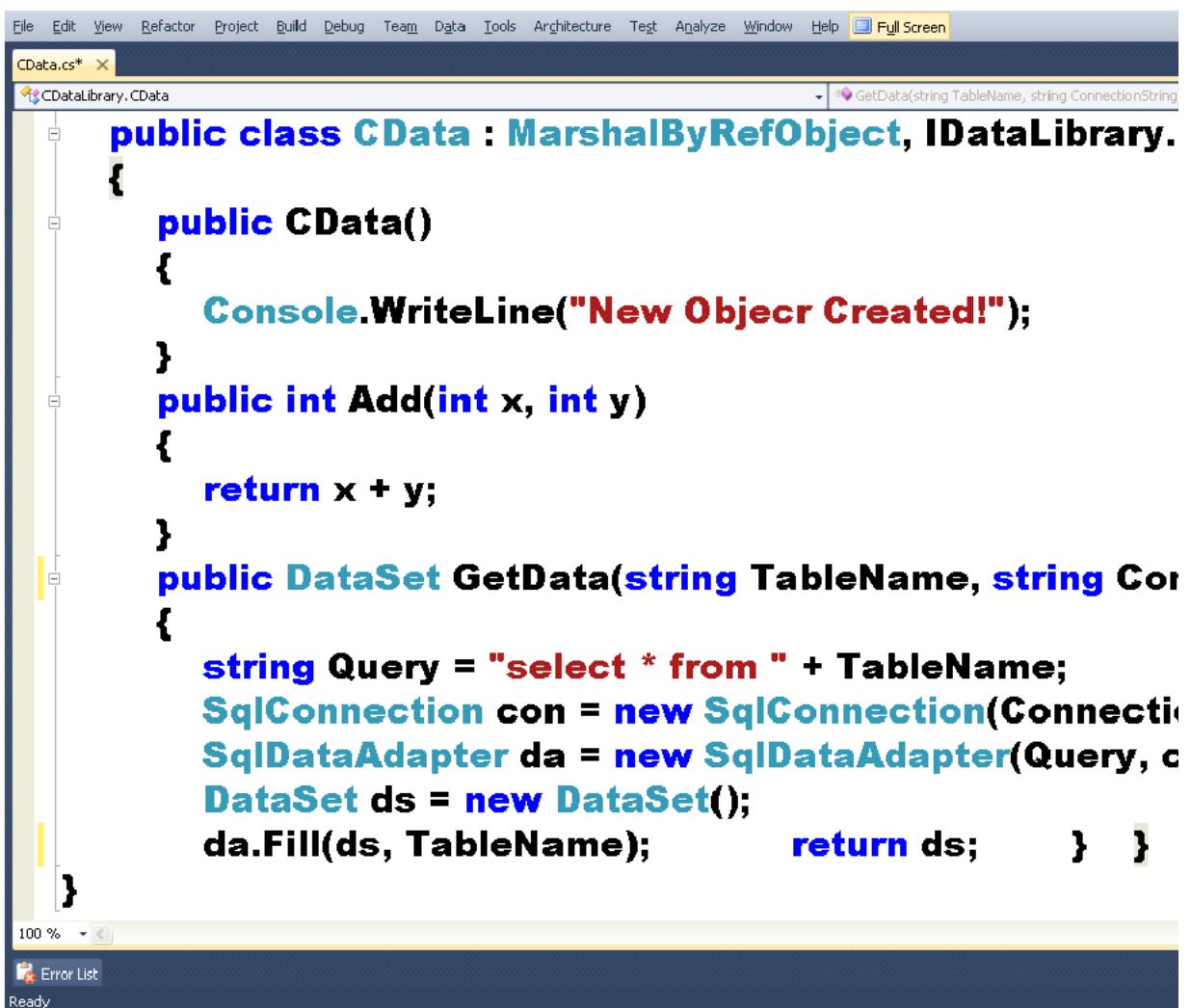
```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace CDataLibrary
{
    public class Class1
    {
    }
}

```

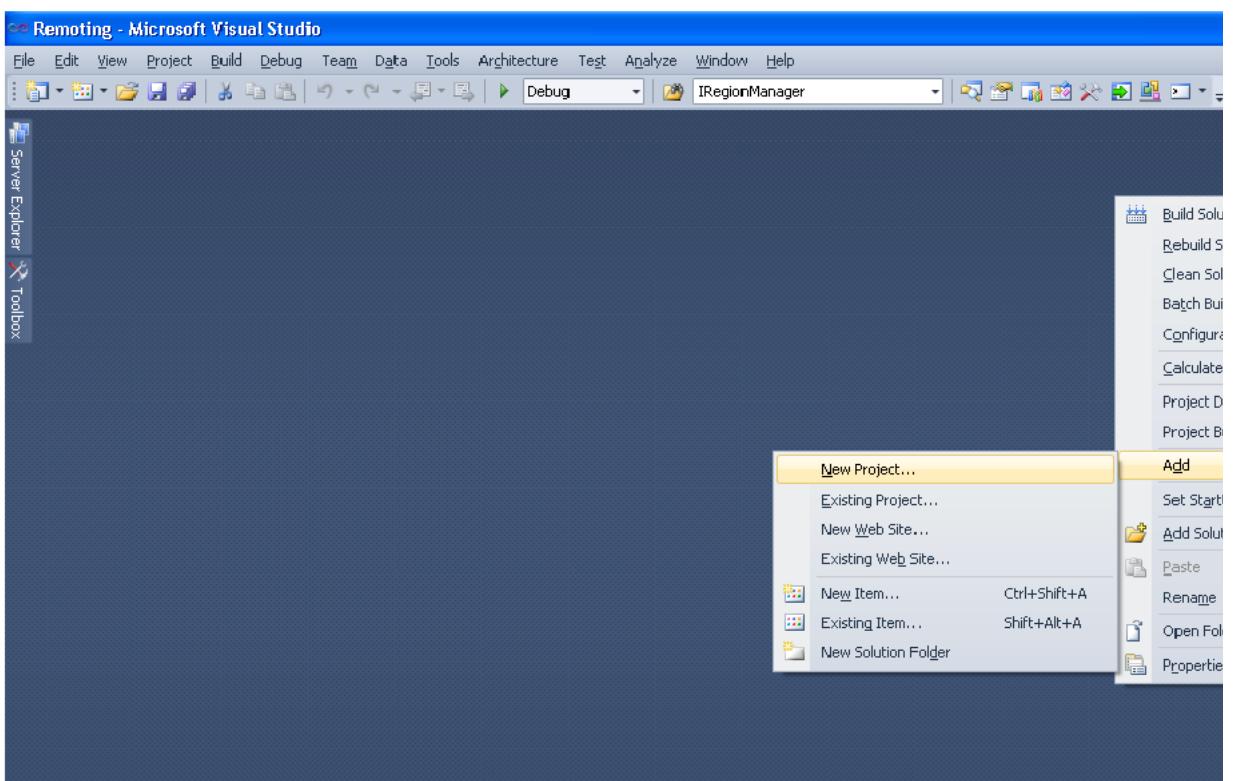


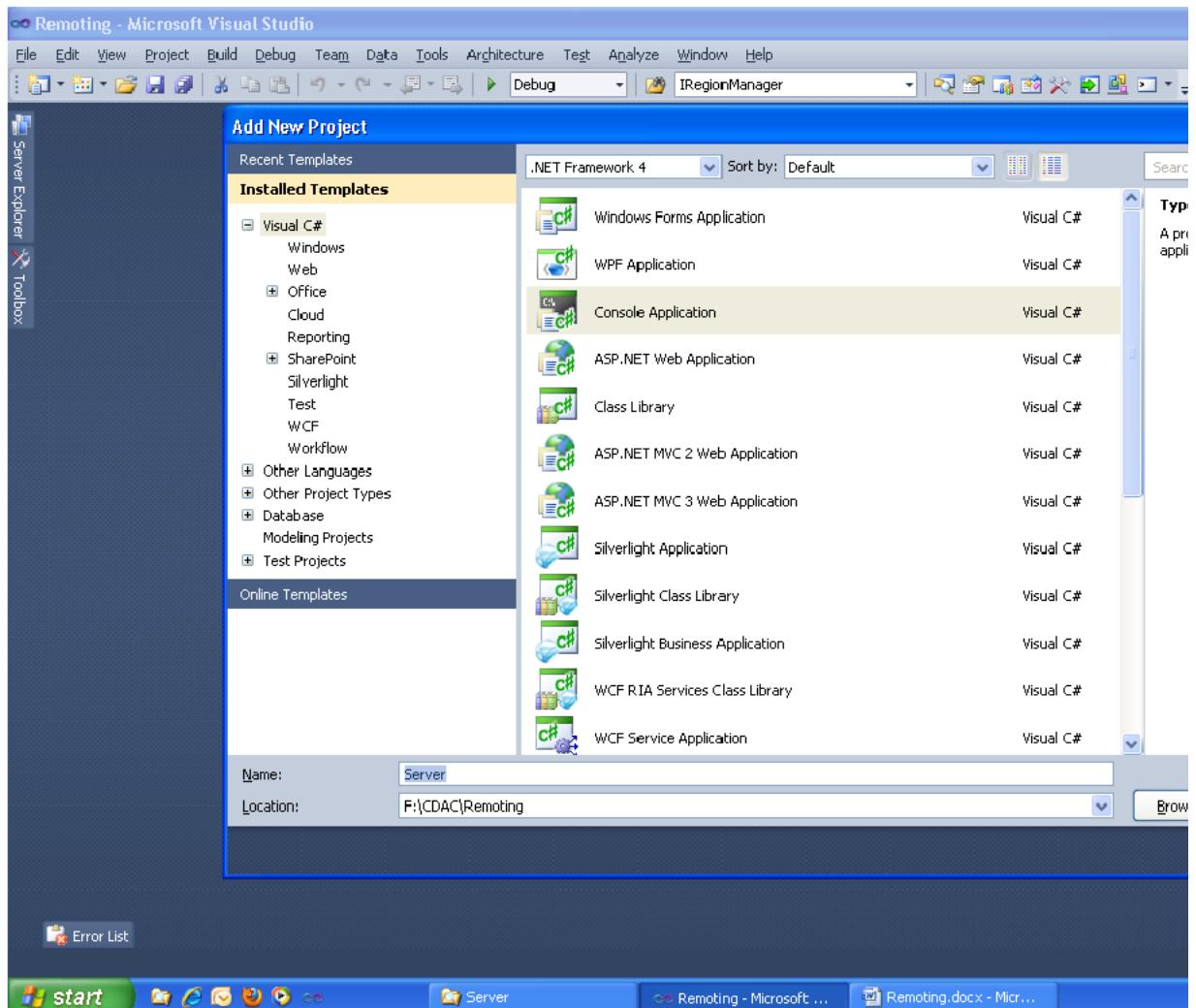


```

public class CData : MarshalByRefObject, IDataLibrary
{
    public CData()
    {
        Console.WriteLine("New Object Created!");
    }
    public int Add(int x, int y)
    {
        return x + y;
    }
    public DataSet GetData(string TableName, string ConnString)
    {
        string Query = "select * from " + TableName;
        SqlConnection con = new SqlConnection(ConnectionString);
        SqlDataAdapter da = new SqlDataAdapter(Query, con);
        DataSet ds = new DataSet();
        da.Fill(ds, TableName);
        return ds;
    }
}

```



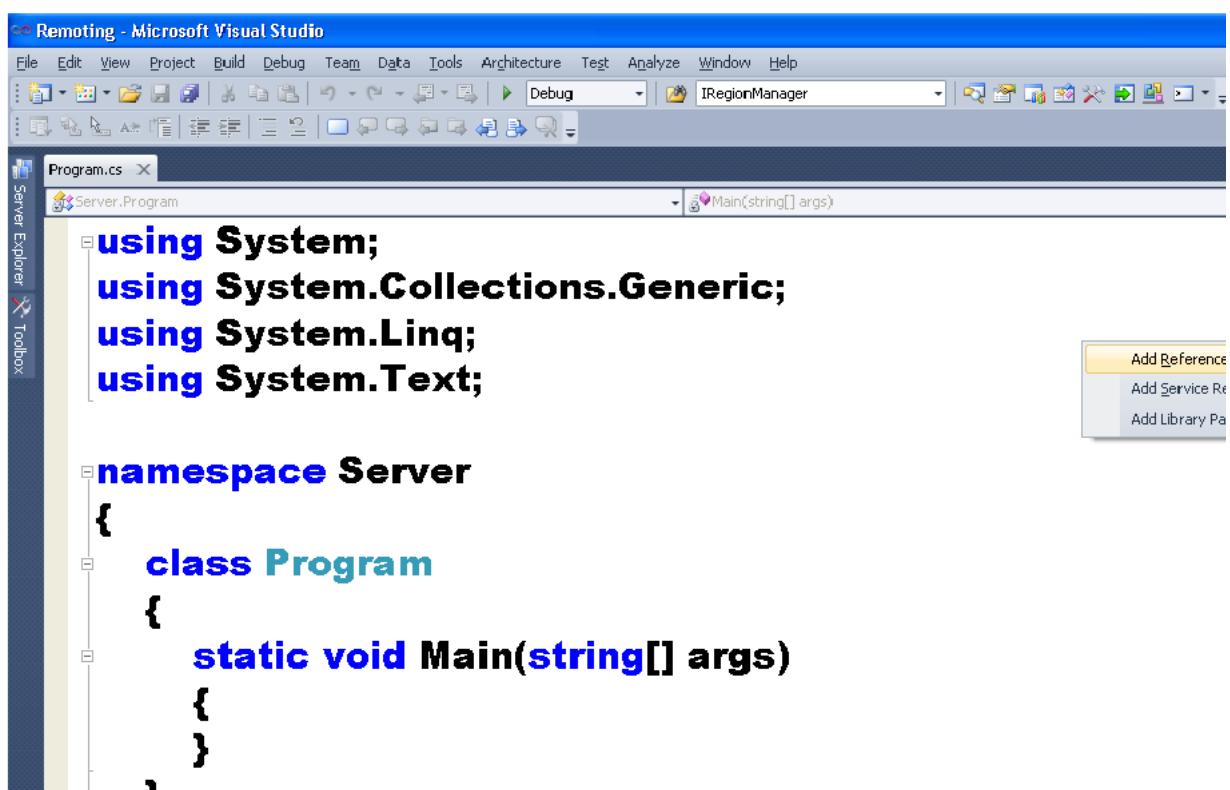
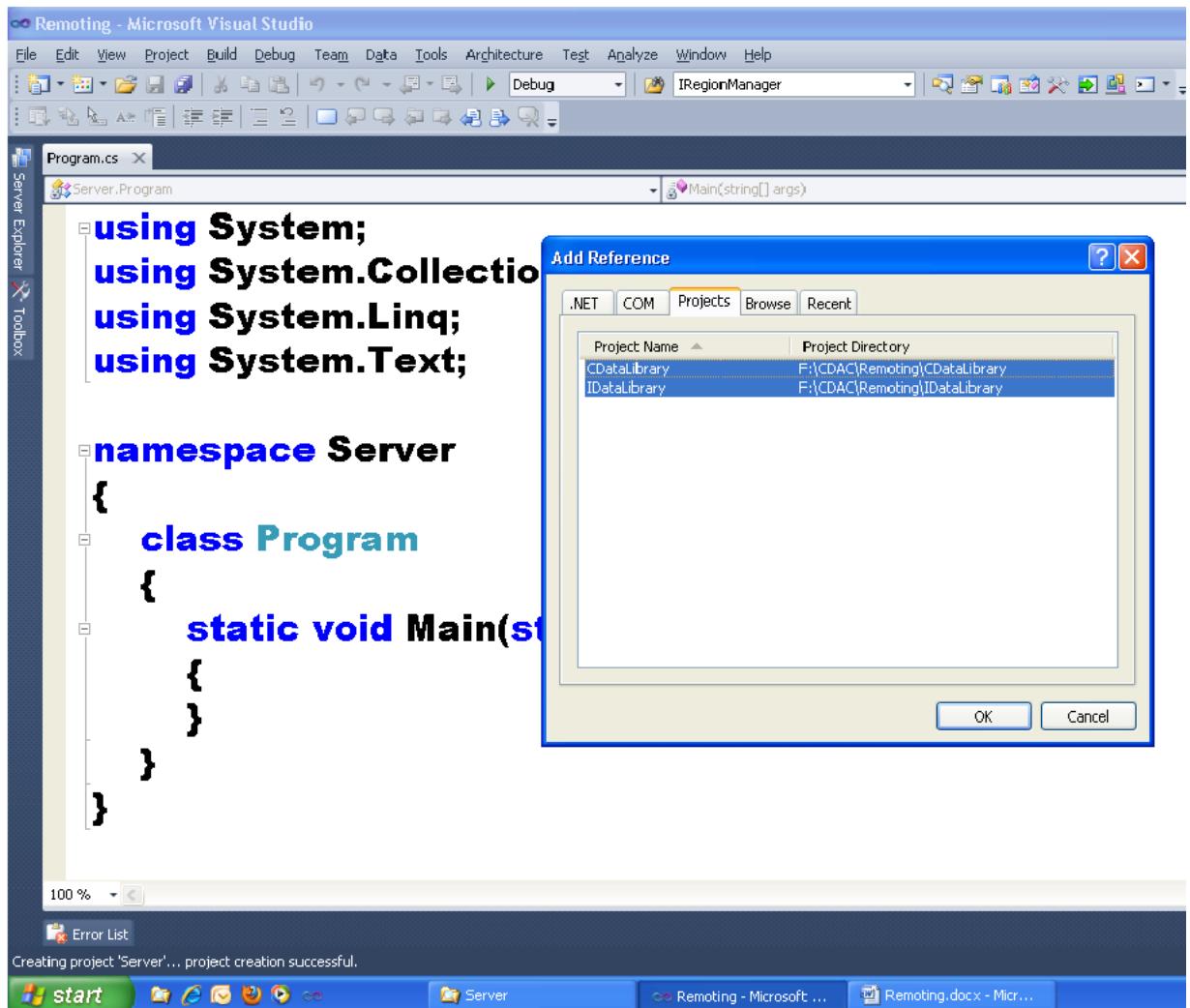


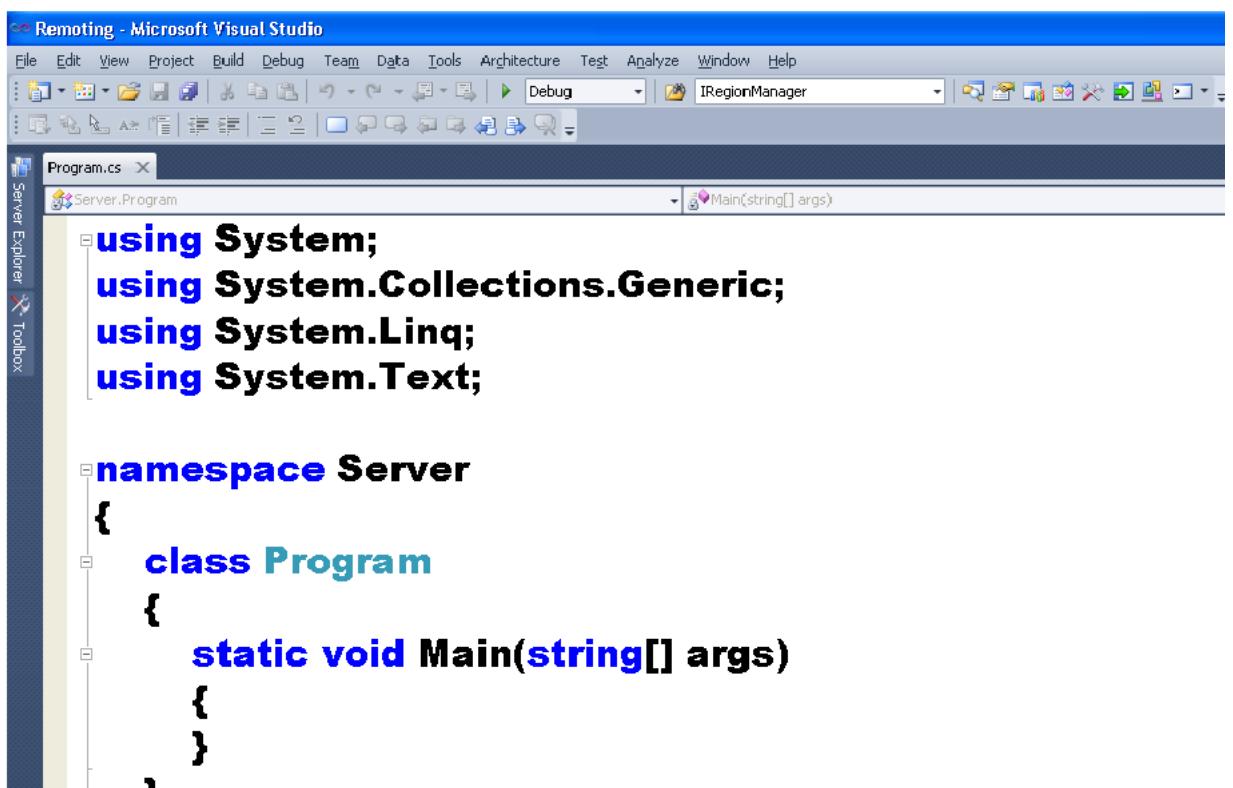
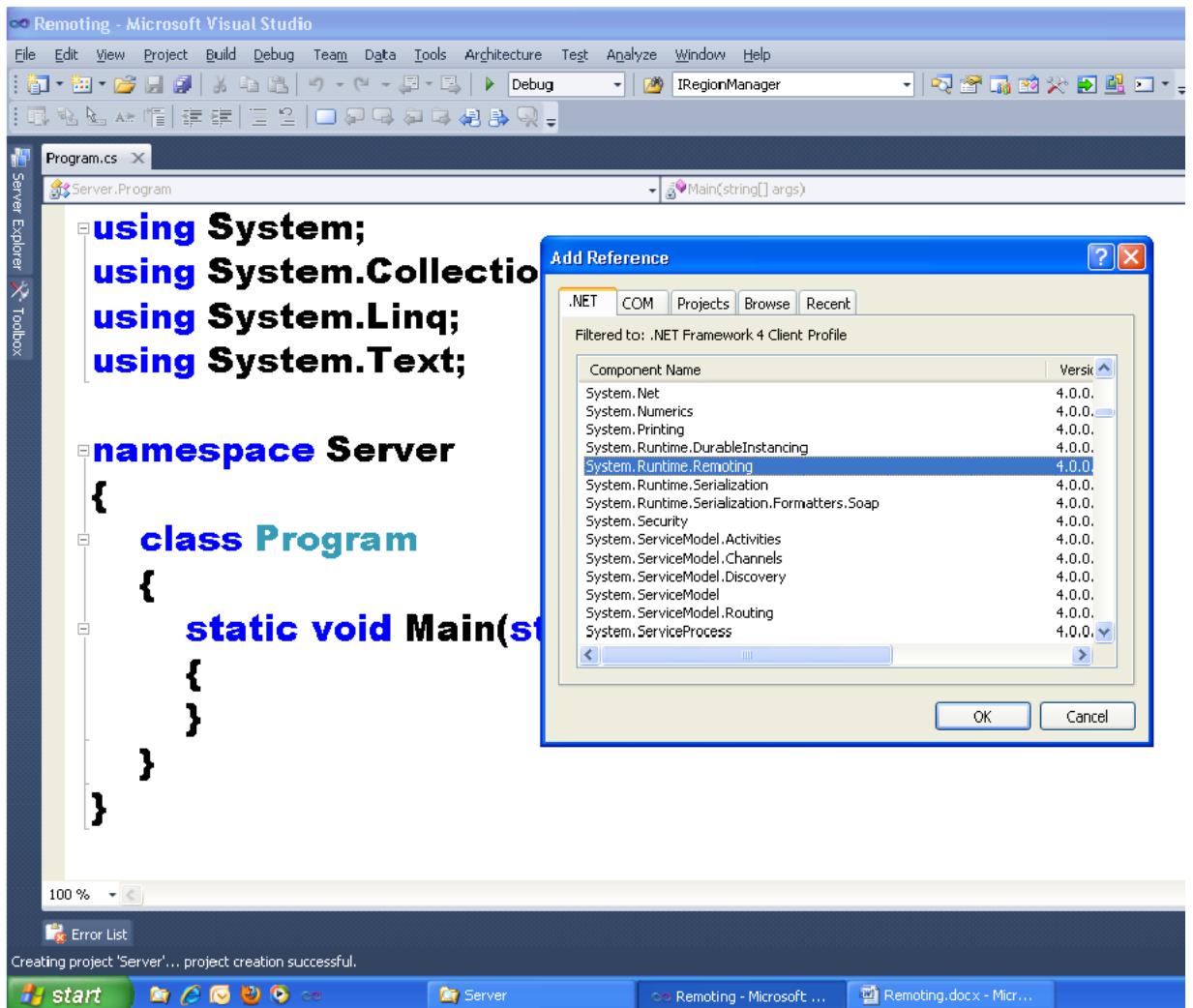
```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Server
{
    class Program
    {
        static void Main(string[] args)
        {
        }
    }
}

```





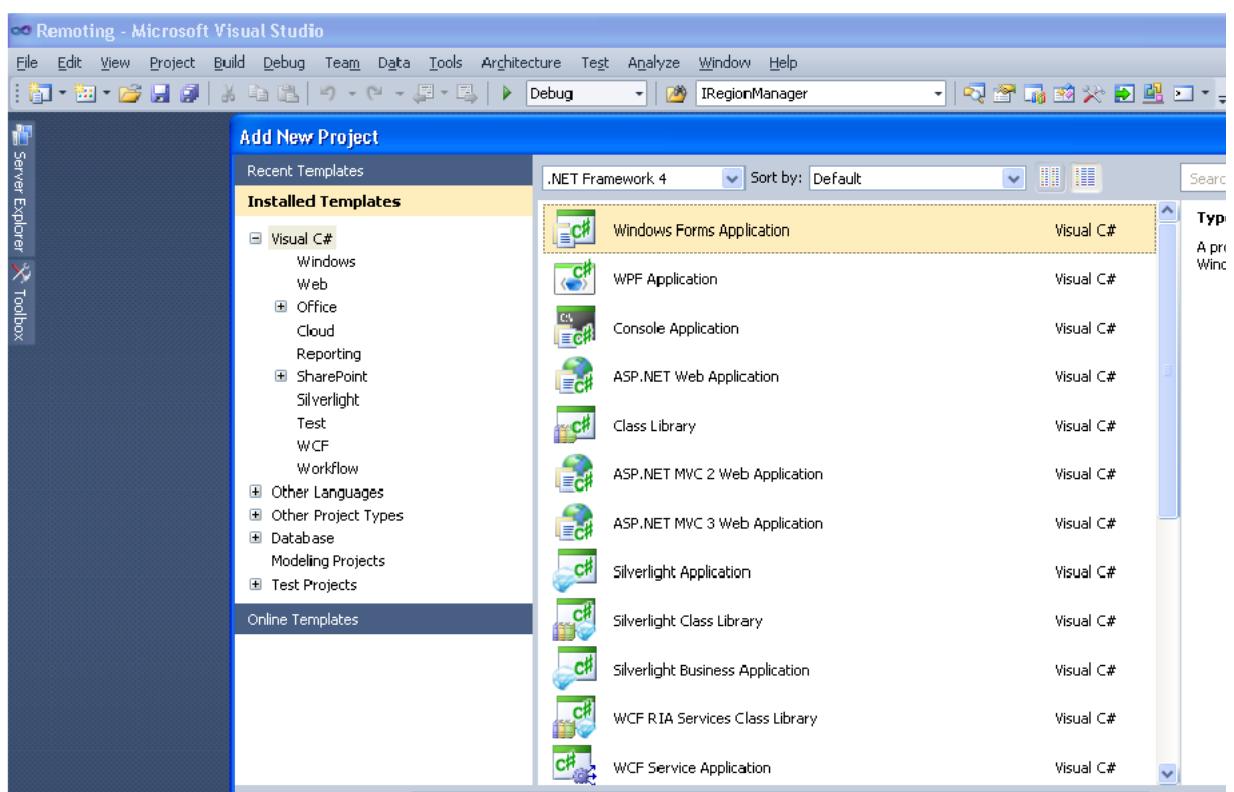
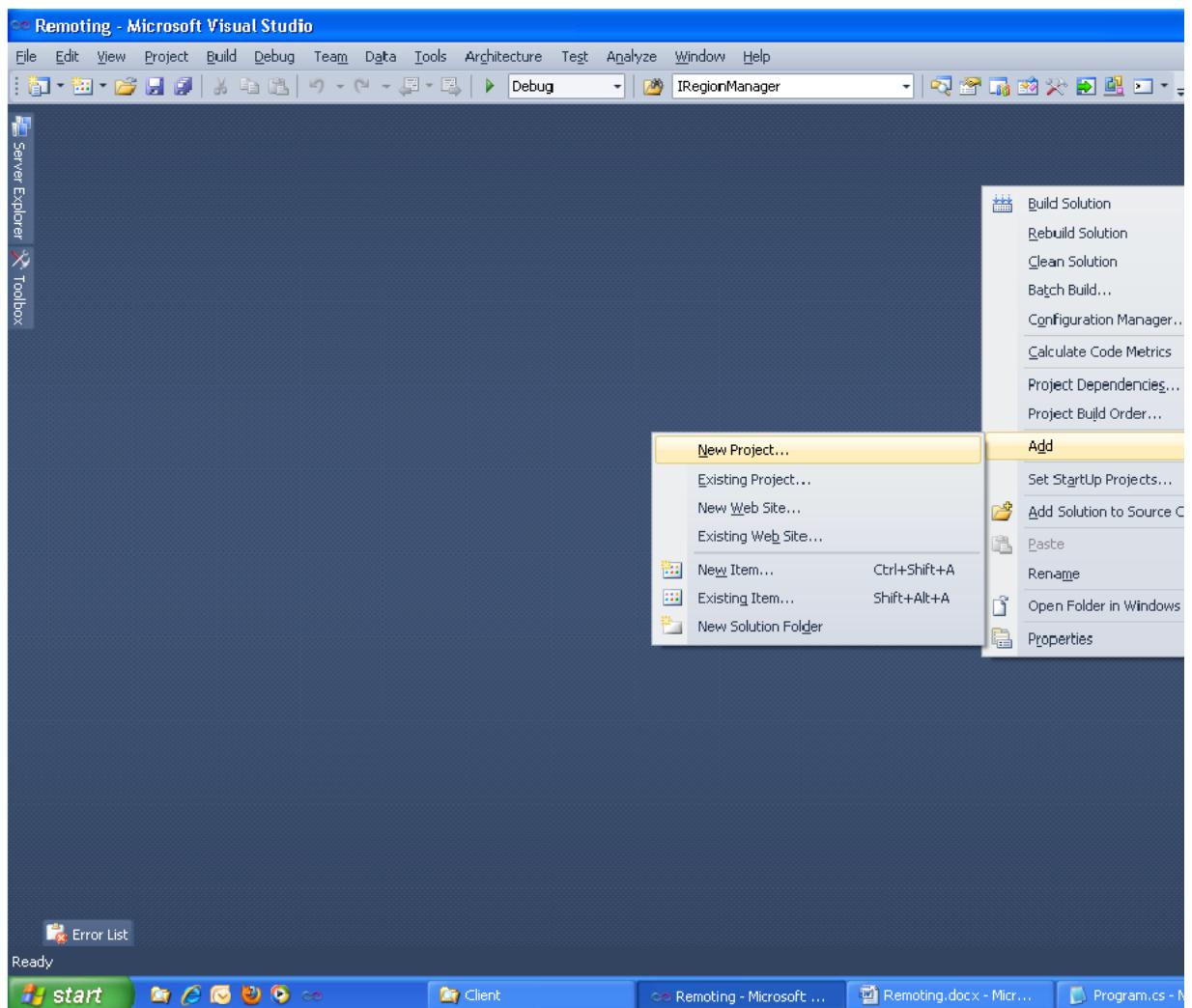
The screenshot shows the Microsoft Visual Studio interface with the title bar "Remoting - Microsoft Visual Studio". The menu bar includes File, Edit, View, Refactor, Project, Build, Debug, Team, Data, Tools, Architecture, Test, Analyze, Window, Help. The toolbar has various icons for file operations like Open, Save, Print, and a "Debug" button. The status bar at the bottom shows "Ready". The main window displays the code for "Program.cs" under the namespace "Server". The code includes several using statements for System, System.Collections.Generic, System.Linq, System.Text, System.Runtime.Remoting, System.Runtime.Remoting.Channels, and System.Runtime.Remoting.Channels.Tcp. It defines a class Program with a static void Main method.

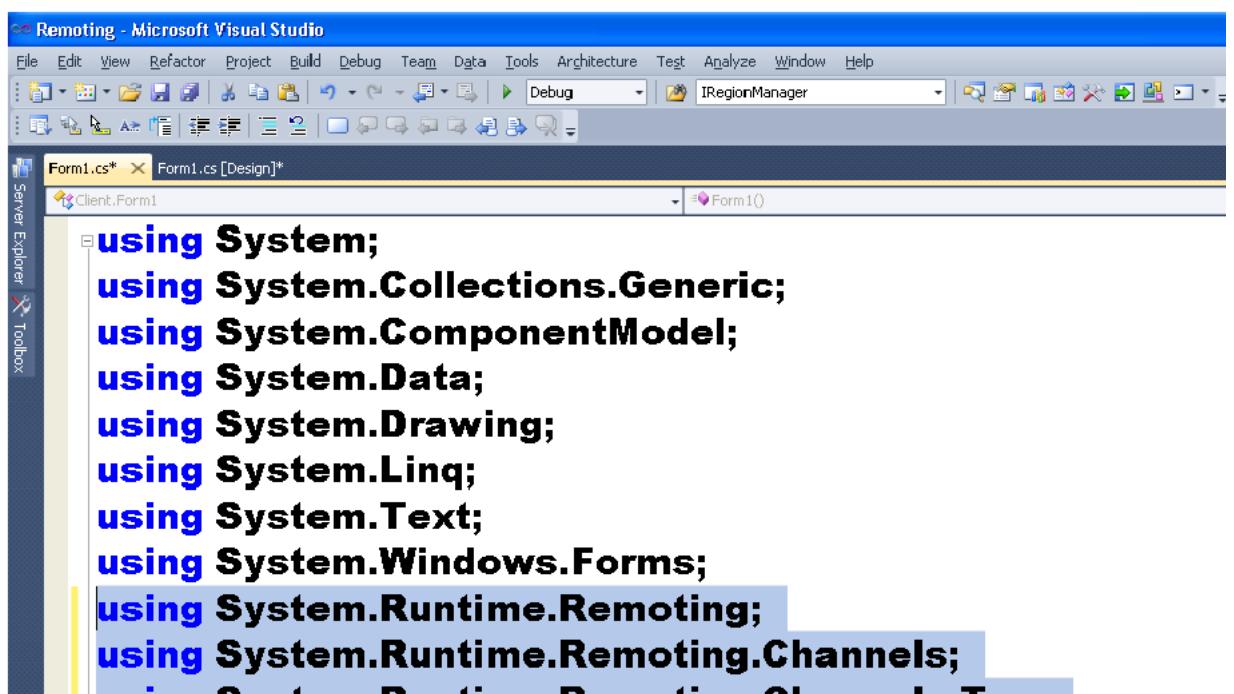
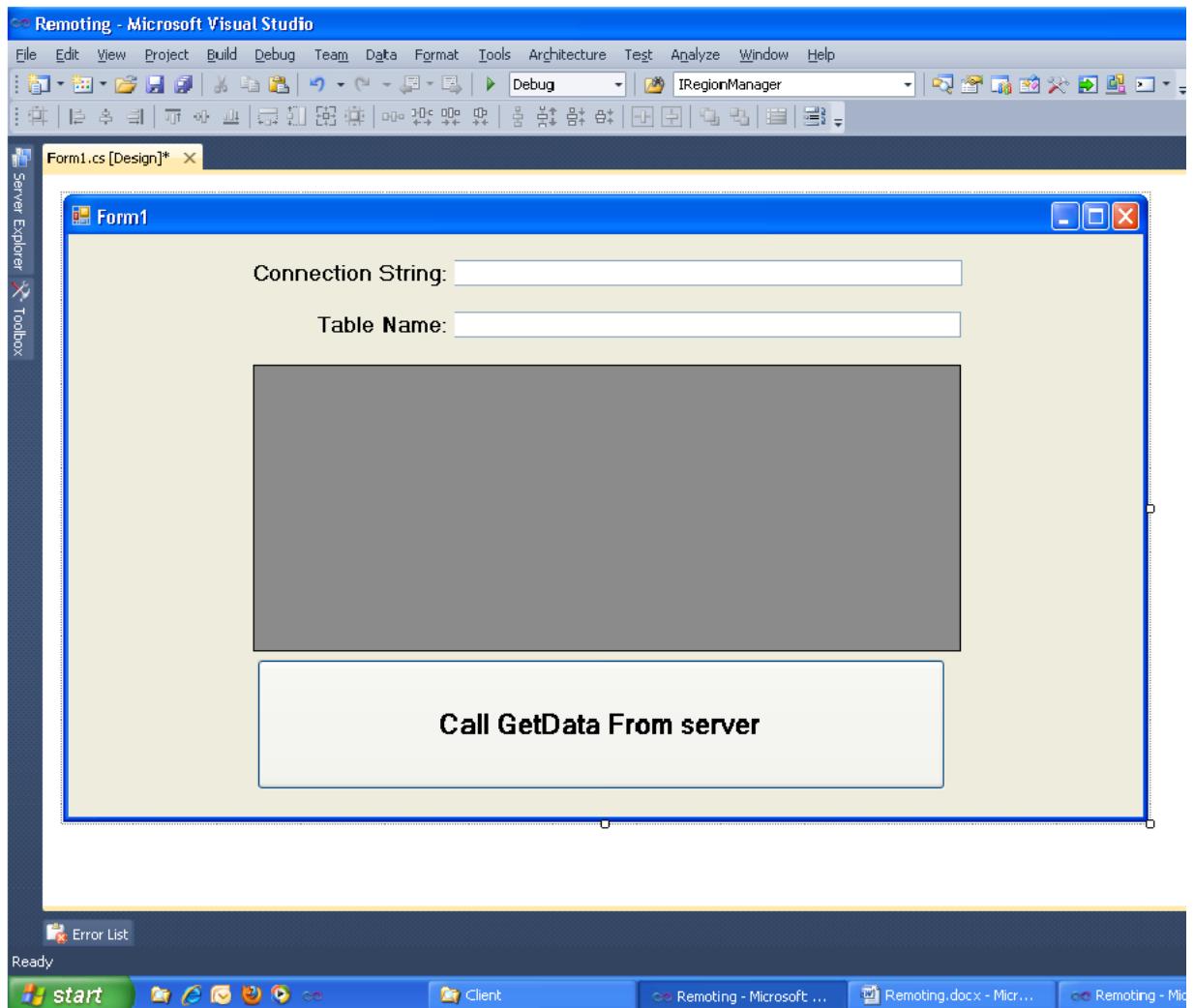
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Runtime.Remoting;
using System.Runtime.Remoting.Channels;
using System.Runtime.Remoting.Channels.Tcp;

namespace Server
{
    class Program
    {
        static void Main(string[] args)
```

This screenshot continues the code from the previous one. It shows the completion of the Main method. The code uses TcpServerChannel to register a service named "MyService" of type CDataLibrary.CData in singleton mode. It then outputs "Server Started ..." to the console and waits for input.

```
TcpServerChannel chnl = new TcpServerChannel();
ChannelServices.RegisterChannel(chnl, false);
RemotingConfiguration.RegisterWellKnownService(
    typeof(CDataLibrary.CData),
    "MyService",
    WellKnownObjectMode.Singleton);
Console.WriteLine("Server Started ...");
Console.ReadLine();
```





The screenshot shows the Microsoft Visual Studio IDE interface. The title bar reads "Remoting - Microsoft Visual Studio". The menu bar includes File, Edit, View, Project, Build, Debug, Team, Data, Tools, Architecture, Test, Analyze, Window, and Help. The toolbar has various icons for file operations like Open, Save, Print, and Build. The status bar at the bottom shows "Ready".

The main code editor window displays the following C# code:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Runtime.Remoting;
using System.Runtime.Remoting.Channels;
using System.Runtime.Remoting.Channels.Tcp;

namespace Client
{
    public partial class Form1 : Form
```

A context menu is open on the right side of the code editor, listing "Add Reference...", "Add Service Refer", and "Add Library Packa".

Remoting - Microsoft Visual Studio

File Edit View Project Build Debug Team Data Tools Architecture Test Analyze Window Help

Form1.cs* Form1.cs [Design]* Client.Form1 Form1()

```

using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows;
using System.Runtime;
using System.Runtime;
using System.Runtime.

namespace Client
{
    public partial class Form1 : Form

```

Add Reference

.NET COM Projects Browse Recent

Project Name	Project Directory
CDataLibrary	F:\CDAC\Remoting\CDataLibrary
IDataLibrary	F:\CDAC\Remoting\IDataLibrary
Server	F:\CDAC\Remoting\Server

OK Cancel

start Remoting - Microsoft ... Remoting.docx - Mic... Remoting - Mic...

File Edit View Refactor Project Build Debug Team Data Tools Architecture Test Analyze Window Help

Form1.cs* Form1.cs [Design]* Client.Form1 Form1()

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Runtime.Remoting;
using System.Runtime.Remoting.Channels;
using System.Runtime.Remoting.Channels.Tcp;

namespace Client
{

```

Add Referer
Add Service
Add Library

