**\*Assignment no-1**

**# Program to show how to accept runtime string in c#.net**

using System;

namespace HelloWorld

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Enter The Word ");

String n1=Console.ReadLine();

Console.WriteLine("How Many Times ");

int no1=Convert.ToInt32(Console.ReadLine());

for(int i=1;i<=no1;i++)

{

Console.WriteLine("\n "+n1);

}

} }

}

**\*Assignment no-2**

**# Program to print “Teach one,Each one,Tree one” for given number of times.**

using System;

public class While

{

public static void Main()

{

int i;

int n;

Console.WriteLine("Enter A Number:- ");

n = Convert.ToInt32(Console.ReadLine());

for(i=1;i<=n;i++)

{

Console.WriteLine("Teach One,Each One,Tree One");

}

Console.ReadLine();

}

}

**Assignment no-3**

**# Program to demonstrate Arithmetic Operators in c#.net**

**using System;**

public class operaters

{

static public void Main()

{

int a,b,c;

Console.WriteLine("Enter value of a=");

a=Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Enter value of b=");

b=Convert.ToInt32(Console.ReadLine());

c=a+b;

Console.WriteLine("Addition of a and b="+c);

Console.ReadLine();

c=a-b;

Console.WriteLine("Substraction of a and b ="+c);

Console.ReadLine();

c=a/b;

Console.WriteLine("Division of a and b ="+c);

Console.ReadLine();

c=a\*b;

Console.WriteLine("Multiplication of a and b="+c);

Console.ReadLine();

c=a%b;

Console.WriteLine("Modulation of a and b="+c);

Console.ReadLine();

}

}

**\*Assignment no-4**

**# Program to demonstrate inheritance in c#.net**

**using system;**

namespace CSharp\_Shell

{

Public class stud

{

Public string Class=”sybca”;

Public string name=”harshali”;

}

Public class exam:stude

{

Public string s1=”C#”;

Public string s2=”audit”;

Public string s3=”rdbms”;

}

Public class result:exam

{

Public int sc=20;

Public int sa=30;

Public int sr=40;

Public static void Main(string [] args)

{

Result r=new result();

Console.WriteLine(r.Class);

Console.WriteLine(r.name);

Console.WriteLine(r.s1);

Console.WriteLine(r.s2);

Console.WriteLine(r.s3);

Console.WriteLine(r.sc);

Console.WriteLine(r.sa);

Console.WriteLine(r.sr);

Console.ReadLine();

}

}

}

**Assignment no-5**

**# Program To demonstrate simple inheritance in c#.net.**

using System;

Namespace CSharp\_Shell

{

Public class Program

{

Public int rollno=107;

Public string name=”harshali”;

}

Class exam:Program

{

Public static void Main()

{

Exam n=new exam();

Console.WriteLine(n.rollno);

Console.WriteLine( n.name);

Console.ReadLine();

}

}

}

**\*Assignment no-6**

**#program to demonstrate multiple inheritance.**

using System;

Namespace inheretance

{

Class student

{

Public void info()

{

String name = “harshali”;

String Class = “SYBCA”;

Console.WriteLine(“student name :-“ +

name);

Console.WriteLine(“Class :-“ + Class);

}

}

Interface exam

{

Void mak();

}

class result:student,exam

{

Public void mak()

{

String subject = “c#.net”;

Int marks = 28;

Console.WriteLine(“subject:-“ + subject);

Console.WriteLine(“marks :-“ + marks);

}

Static void Main(string[] args)

{

Result obj = new result();

Obj.info();

Obj.mak();

Console.ReadLine();

}

}

}

**Assignment no-7**

**# program to demonstrate use of polymorphism in c#.net.**

using System;

Namespace CSharp\_Shell

{ Public class Program

{

Void add(int a,int b)

{

Int c=a+b;

Console.WriteLine(“Addition of a and b=”+c);

}

Void add(int a,int b,int c)

{

Int c1=a+b+c;

Console.WriteLine(“Addition of a.b and c=”+c1);

}

Public static void Main(string [] args)

{ Program p=new Program();

p.add(10,20);

p.add(10,20,30);

Console.ReadLine();

} } }

**\*Assignment no-8**

**# Program to demonstrate function definition and function call in c#.net.**

using System;

Namespace CSharp\_Shell

{

Public class Program

{

Void print()

{

Console.WriteLine(“Harshali”);

}

Public static void Main()

{

Program p=new Program();

p.print();

} } }

**\*Assignment no-9**

**# program to demonstrate parameterize function in c#.net.**

using System;

namespace CSharp\_Shell

{

Public class Program

{

Int x=0;

Void print(int x,int y)

{

x=x+y;

Console.WriteLine(x);

}

Public static void Main()

{

Program p=new Program();

p.print(5,5);

} } }

**\*Assignment no-10**

**# Program to show table of given number using while loop.**

Using System;

Using System.IO;

Using System.Linq;

Using System.Collections.Generic;

Public class PrintWorld

{

Public Static void Main()

{

Int I =1;

Int n;

Console.WriteLine(“Enter A Number”);

n = Convert.ToInt32(Console.ReadLine());

Console.WriteLine(“table of given number”);

While (I <= 10)

{

Console.WriteLine( n \* i);

I++;

} } }

**Assignment no-11**

**# program to demonstrate while loop in c#.net show series of given number.**

Using System;

Namespace CSharp\_Shell

{

Public class Program

{

Public static void Main()

{

Int i=1,n;

Console.WriteLine(“Enter a number:-“);

n=Convert.ToInt32(Console.ReadLine());

Console.WriteLine(“seris of 1 to 10 in for loop:-“);

While(i<=n)

{

Console.WriteLine(i);

I++;

}

Console.ReadLine();

} } }

**Assignment no-12**

**# program to demonstrate for loop in c#.net show series of given number.**

Using System;

Namespace CSharp\_Shell

{ Pu

blic class Program

{

Public static void Main()

{ Int I,n;

Console.WriteLine(“Enter a number:-“);

N=Convert.ToInt32(Console.ReadLine());

Console.WriteLine(“seris of 1 to 10 in for loop:-“);

For(i=1;i<=n;i++)

{

Console.WriteLine(i);

} Console.ReadLine(); } } }

**\*Assignment no-13**

**# Demonstration of continue statement in for loop**

Using System;

Namespace CSharp\_Shell

{

Public class Program

{

Public static void Main()

{

For(int x = 1; x <= 12; x++)

{

If (x == 3)

{

Continue;

}

Console.WriteLine(x);

} } } }

**Assignment no-14**

**# demonstration of continue statement in while loop.**

Using System;

namespace CSharp\_Shell

{

Public static class Program

{

Public static void Main()

{

Int i=1;

While(i<=12)

{

I++;

If(i==5)

{

Continue;

}

Console.WriteLine(i);

}

}

} }

**Assignment no-15**

**# Program to show first 3 even numbers in for loop.**

using System;

namespace CSharp\_Shell

{

public class Program

{

public static void Main()

{

int i;

int n;

Console.WriteLine("Enter a number:-");

n=Convert.ToInt32(Console.ReadLine());

Console.WriteLine("first 3 even numbers of

given number in for loop:-");

for(i=1;i<=6;i++)

{

{

if(i%2==0)

{

Console.WriteLine(i);

}

}

}

Console.ReadLine();

}

}

}

**\*Assignment no-16**

**# Program to demonstrate if else statement in c#.net.**

using System;

namespace CSharp\_Shell

{

public class Program

{

public static void Main()

{

int age;

Console.WriteLine("Entet age of candidate");

age=Convert.ToInt32(Console.ReadLine());

if(age>=18)

{

Console.WriteLine("Eligible for vote");

}

else

{

Console.WriteLine("Not eligible for vote");

}

}

}

}

**\*Assignment no-17**

**# Program to demonstrate default constructor and parameterize constructor in c#.net**

Using System;

namespace CSharp\_Shell

{

Public class Program

{

Int r;

String n;

Program(int rollno,string name)

{

r=rollno;

n=name;

Console.WriteLine(r+”\t“+n);

}

Program()

{

Console.WriteLine(“Constructor called”);

}

Public static void Main(string [] args)

{

Program p=new Program();

Program p1=new Program(107,”harshali”);

}

}

}

**\*Assignment no-18**

**#Program to demonstrate destructor in c#.net.**

Using System;

namespace CSharp\_Shell

{

Public class Program

{

Public Program()

{

Console.WriteLine(“Constructor called”);

}

~Program()

{

Console.WriteLine(“Destructor called”);

}

Public static void Main(string [] args)

}

Program p=new Program();

}

}

}

**\*Assignment no-19**

**# program to show function overriding in c#.net.**

Using System;

Namespace CSharp\_Shell

{

Public class Program

{

Public virtual void showinfo()

{

Console.WriteLine(“base class method”);

}

}

Class b:Program

{

Public override void showinfo()

{

Console.WriteLine(“derived class method”);

}

Public static void Main()

{

B b1=new b();

B1.showinfo();

B1.showinfo();

}

}

}

**\*Assignment no-20**

**# Program to demonstrate array in c#.net.**

Using System;

Namespace array

{

Public class Array

{

Public static void Main()

{

Int num;

Int [] a={1,2,3,4,5,6};

Console.WriteLine(“enter any number:-“);

Num=Convert.ToInt32(Console.ReadLine());

If(num==a[0])

{ Console.WriteLine(“true”);

}

Else if(num==a[1])

{ Console.WriteLine(“true”);

}

Else if(num==a[2])

{ Console.WriteLine(“true”);

}

Else if(num==a[3])

{ Console.WriteLine(“true”);

}

Else if(num==a[4])

{ Console.WriteLine(“true”);

}

Else if(num==a[5])

{ Console.WriteLine(“true”);

}

Else

{

Console.WriteLine(“false”);

}

}

} }

**\*Assignment no-21**

**# Program to show maximum and minimum number in an array.**

Using System;

Namespace CSharp\_Shell

{

Public static class Program

{

Public static void Main()

{

Int [] a={12,34,56,78,89};

Int I,max,min;

Max=a[0];

Min=a[0];

For(i=1;i<5;i++)

{

If(a[i]>max)

{

Max=a[i];

}

If(a[i]<min)

{

Min=a[i];

}

}

Console.WriteLine(“Maximum number:-“+max);

Console.WriteLine(“Minimum number:-“+min);

}

}

}

**\*Assignment no-2**

**# Program to demonstrate call by value in c#.net.**

Using System;

Namespace CSharp\_Shell

{

Public class Program

{

Void get(int x)

{

X=10;

Console.WriteLine(x);

}

Public static void Main()

{

Int x=20;

Program p=new Program();

Console.WriteLine(“before function call:-“+x);

p.get(x);

Console.WriteLine(“after function call:-“+x);

} } }

**\*Assignment no-23**

**# Program to demonstrate call by reference in c#.net.**

Using.system;

Namespace CSharp\_Shell

{

Public class Program

{

Void get(ref int x)

{

X=10;

Console.WriteLine(x);

}

Public static void Main()

{ Int x=20;

Program p=new Program();

Console.WriteLine(“before function call:-“+x);

p.get(ref x);

Console.WriteLine(“after function call:-“+x); } } }

**\*Assignment no-24**

**#Program to demonstrate exception handling in c#.net.**

Using System;

namespace CSharp\_Shell

{

Public class Program

{

Public static void Main()

{

Try

{

Int a=2;

Int b=0;

Int c=a/b;

Console.WriteLine(c);

}

Catch(Exception e)

{

Console.WriteLine(“Exception occurred:-“+e);

}

Finally

{

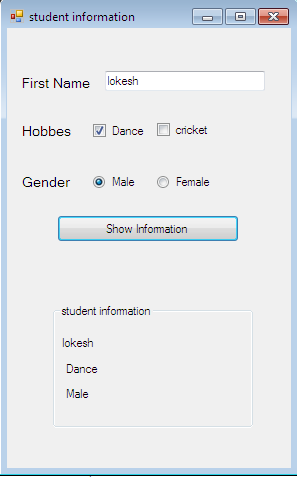
Console.WriteLine(“Exception handled”);

} } } }

**Windows Application Development**

**Assignment no-25**

**//Demonstration of TextBox,CheckBox,RadioButton,Button,Label,GroupBox.**



using System;

using System.Windows.Forms;

namespace studinfo

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void btn1\_Click(object sender, EventArgs e)

{

lblname.Text=txt1.Text;

if(chk1.Checked==true)

{

lblhobb.Text=chk1.Text;

}

if(chk2.Checked==true)

{

lblhobb.Text=chk2.Text;

}

if (rdb1.Checked == true)

{

lblgend.Text = rdb1.Text;

}

if (rdb2.Checked == true)

{

lblgend.Text = rdb2.Text;

}

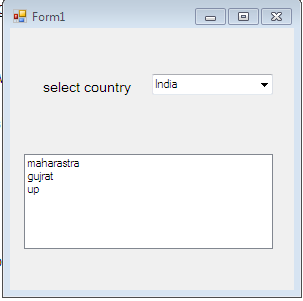
} } }

}

1

**Assignment no-26**

**#Create a C# application using ListBox, ComboBox control**



using System;

using System.Windows.Forms;

namespace WindowsFormsApplication1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void cmbselect\_SelectedIndexChanged(object sender, EventArgs e)

{

lB1.Items.Clear();

if (cmbselect.SelectedItem == "India")

{

lB1.Items.Add("maharastra");

lB1.Items.Add("gujrat");

lB1.Items.Add("up");

}

if (cmbselect.SelectedItem == "USA")

{

lB1.Items.Add("A");

lB1.Items.Add("B");

lB1.Items.Add("C");

}

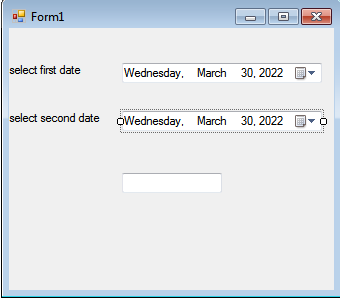
}

}

}

**\*Assignment no-27**

**# Demonstrate use of DateTimePicker**



using System;

using System.Windows.Forms;

namespace WindowsFormsApplication1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void dateTimePicker2\_ValueChanged(object sender, EventArgs e)

{

int d;

DateTime firstdate = Convert.ToDateTime(dateTimePicker1.Text);

DateTime seconddate = Convert.ToDateTime(dateTimePicker2.Text);

d = firstdate.Subtract(seconddate).Days;

textBox1.Text = Math.Abs(d).ToString();

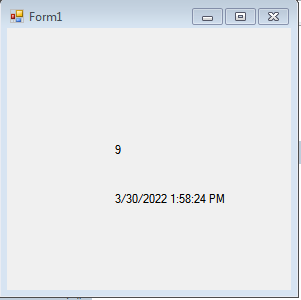
}

}

}

**Assignment no-28**

**//demonstration use of timer control.**



using System;

using System.Windows.Forms;

namespace timer

{

public partial class Form1 : Form

{

int t;

public Form1()

{

InitializeComponent();

timer1.Start();

timer2.Start();

}

private void timer1\_Tick(object sender, EventArgs e)

{

t++;

label1.Text = t.ToString();

if (t == 10)

{

timer1.Stop();

label1.Text = "time out";

}

}

private void timer2\_Tick(object sender, EventArgs e)

{

label2.Text = DateTime.Now.ToString();

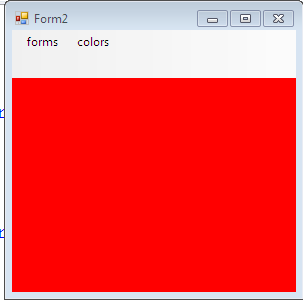
}

}

}

**\*Assignment no-29**

**//demonstration of menustrip control to change background color and open different forms**



using System;

using System.Windows.Forms;

namespace manustrip107.cs

{

public partial class Form2 : Form

{

public Form2()

{

InitializeComponent();

}

private void form1ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Form1 f1 = new Form1();

f1.Show();

}

private void redToolStripMenuItem1\_Click(object sender, EventArgs e)

{

this.BackColor = Color.Red;

}

private void greenToolStripMenuItem1\_Click(object sender, EventArgs e)

{

this.BackColor = Color.Green;

}

private void blueToolStripMenuItem1\_Click(object sender, EventArgs e)

{

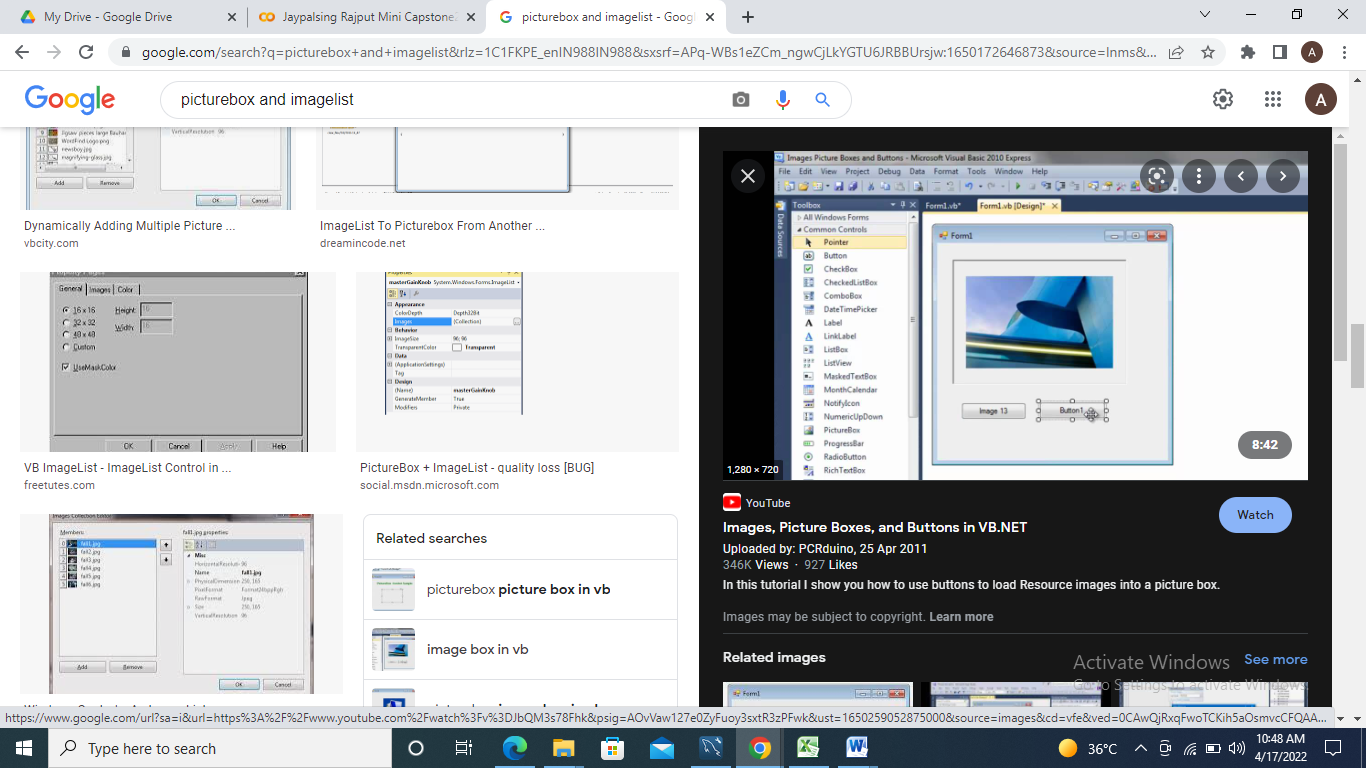
this.BackColor = Color.Blue;

}

} }

**\*Assignment no-30**

**//Demonstration of PictureBox.**



using System;

using System.Windows.Forms;

namespace manustrip107.cs

{

public partial class Form1 : Form

{

int count=-1;

public Form1()

{

InitializeComponent();

}

private void bt2\_Click(object sender, EventArgs e)

{

if (count <= 8)

{

count = count + 1;

}

pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;

pictureBox1.Image=imageList1.Images[count];

}

private void bt1\_Click(object sender, EventArgs e)

{

if (count >= 0)

{

count = count - 1;

}

pictureBox1.Image = imageList1.Images[count]; } } }

**Assignment No: 31**

**#Demonstrate Simple Database Connectivity using wizard.**

**Creating Database and Table:**

**Create the Customers table**

1. Open Visual Studio and create new C# Window Application.
2. In **Server Explorer** ->Select **Data Connections->**Right Click-> Select **Create New SQL server Database.**
3. Specify in wizard **Data Source=.** Or **actual server name and Database Name=HR .**
4. Go to **Server Explore**-> Expand **database HR**->Right-click on **Tables** and select **Add New Table**.

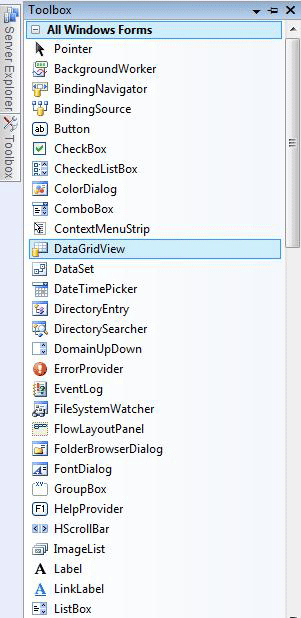
The Table Designer opens and shows a grid with one default row, which represents a single column in the table that you're creating. By adding rows to the grid, you'll add columns in the table.

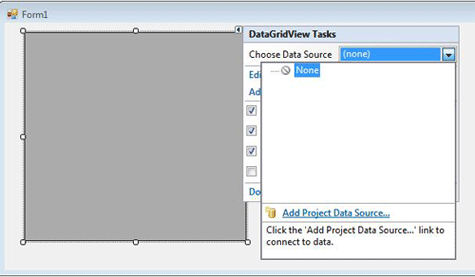
1. In the grid, add a row for each of the following entries:

| **Column name** | **Data type** | **Allow nulls** |
| --- | --- | --- |
| EmpID | nchar(5) | False (cleared) |
| EName | nvarchar(50) | False (cleared) |
| ContactName | nvarchar (50) | True (selected) |
| Phone | nvarchar (24) | True (selected) |

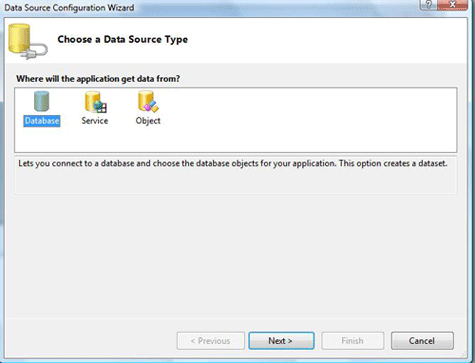
1. Open the shortcut menu for the **Tables** node, select **Refresh**, and then expand the **Tables** node.
2. Open the shortcut menu for the emp table, and then select **Show Table Data** or **View Data**.
3. Add 5 records.

**Retrieving Data from database using DataGridView Control.**

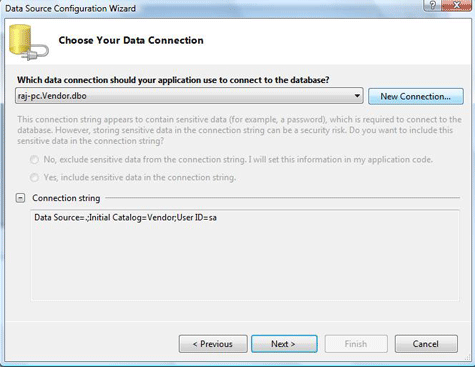
1. Drag and drop DataGridView control from toolbox to form window.  
   
2. Now choose a data source by right clicking on the DataGridView and then click on **Add Project Data Source**. We will be adding a new data source to the project right now.

****  
Figure 3.

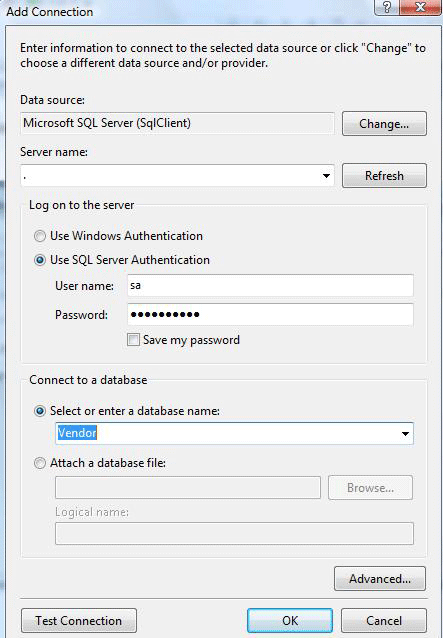
1. Now select data source type, I am choosing Database in this example.  
   After selecting Database, click next.

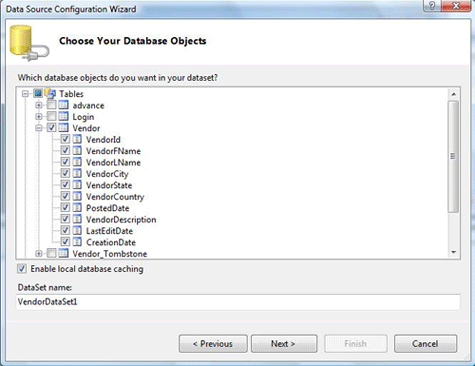
  
Figure 4

1. Choose your data connection, if you already have a connection available select that, otherwise make new connection, and follow the steps and after that click next.



1. Figure 5.
2. Add Connection Wizard.



1. Figure 6
2. Choose you database objects, and click next, if you want local database caching then check enable local database caching check box and click next.
3. 
4. Figure 7.
5. Select table for caching.

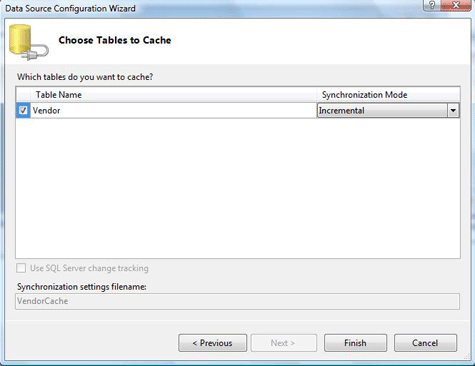
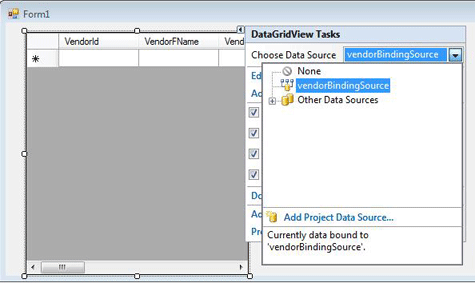


Figure 8.

1. Now your data source is created, right click on DataGridView control and on the option Choose Data Source, select the data source you just created.



1. **Build and Run Application.**