

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



إنشاء مشروع دليل هاتف باستخدام

Sql Server - Linq - Crystal Report



جمع وتصميم

Kslawy

إعداد

Dreamsdotnet

عضوي منتدي فيجوال بيسك لكل العرب

<http://vb4arb.com/vb/forum.php>

مارس ٢٠١٣م

قد سمع الجميع عن هذه التقنية، وربما البعض قد استخدمها أيضاً، ولكن باعتقادي أنها غير مستخدمة بشكل كبير في عالم البرمجة العربية!

Linq تقنية استعلام عن البيانات بأشكال كثيرة، وليست محصورة في التعامل مع قواعد البيانات فقط. تجد فيها السهولة والقوة في آن واحد، تخلو من التعقيدات، وتختصر الكود البرمجي، تصبح جداول وأعمدة قاعدة البيانات فئات وخصائص، يتم التعامل معها من كود لغة البرمجة المستخدمة....

لو نظرنا لهذا الكود، على افتراض أن (Person) فئة داخل المشروع:

```
label1.Text = Person.Name;
textBox1.Text = Person.Age;
```

أو هذا الكود:

```
Person.Name = "Abd Allah";
Person.Age = 40;
```

*أترك لك كتابة كود sql لجلب البيانات من الجدول واستعراضها في الأدوات، أو تغيير قيمها، لتقارن ذلك بنفسك.

البعض يرفض هذه الطريقة في التعامل مع قواعد البيانات، وذلك بأنها لا توحى للشخص أنه يتعامل مع قاعدة بيانات، والبعض يقول: هكذا ماتت sql! لغة التعامل مع قواعد البيانات.

الزبون لا يهمله الكود وشكله، ولا التقنيات المستخدمة أو لغة البرمجة، ما يهمله أن يحصل على طلبه، والمبرمج يهمله الإنتاجية وسرعة البرمجة، كمان أن شكل وترتيب الكود له أهمية كبيرة أيضاً، ناهيك عن التصميم وترتيب الأدوات....

لن أطيل الكلام، ودعنا نبدأ مشروع صغير (دليل هاتف)، سوف نضع فيه ما يلزم لمشروع كبير إن شاء الله، فكل مشاريع قواعد البيانات متشابهة، فقط حجم المشروع وعدد الشاشات هما الفرق.

سوف يحتوي مشروعنا على ما يلي:

١. نموذج لتسجيل الدخول.
٢. نموذج لإضافة مستخدم.
٣. نموذج لحذف / إيقاف مستخدم.
٤. نموذج لتعديل اسم المستخدم وكلمة المرور.
٥. تبويب (أو لسان) لإدخال (اسم ورقم / أرقام هاتف في الدليل).
٦. تبويب لتعديل البيانات (اسم، رقم / أرقام الهاتف).
٧. تبويب لحذف البيانات (اسم، رقم / أرقام الهاتف).
٨. تبويب لاستعراض البيانات.

لنلقي نظرة على قاعدة البيانات، سوف نحتاج للجداول التالية:

- جدول للمستخدمين.
- جدول للأسماء.
- جدول لأرقام الهواتف.

جدول المستخدمين سوف يحتوي على الأعمدة:

- عمود للاسم.
- عمود لكلمة المرور.
- عمود هل المستخدم مدير.
- عمود هل المستخدم مفعّل.

جدول الاسماء سوف يحتوي على الأعمدة:

- الاسم.
- العنوان.
- ملاحظات.

جدول أرقام الهواتف سوف يحتوي على الأعمدة:

- رقم الشخص. (علاقة)
- رقم الهاتف.

* في كل جدول سوف يكون هنالك حقل ID خواصه: مفتاح أساسي وترقيم تلقائي.

نبدأ مشروع جديد باسم (TelephoneBook) ، أول ما سنقوم به هو عمل قاعدة البيانات، نتوجه لـ Menu Bar ونختار:

- Project
- Add New Item

ثم نختار LINQ To SQL Classes ، ليكن اسمه (PhoneBookDB).

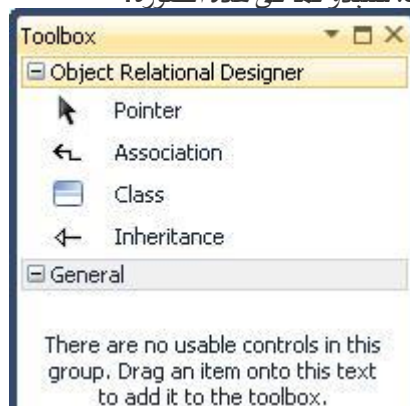
* يمكننا فعل ذلك من Solution Explorer أيضاً، بتحديد المشروع ثم الضغط بالزر الأيمن من الفأرة واختيار:

- Add
- New Item

التبويب الذي ظهر أمامنا بعد إضافة العنصر يبدو فارغاً ولا يحتوي سوى:

The Object Relational Designer allows you to visualize data classes in your code.
Create data classes by dragging items from Server Explorer or Toolbox onto this design surface.

نختار Toolbox لتطالعنا نافذة الأدوات، ستبدو كما في هذه الصورة:



هذه الأدوات الخاصة بالفئة (محتوى البيانات)، نسحب منها أداة Class ونضعها على المصمم:



الآن لو ضغطنا مرتين بالفأرة على كلمة Class1 سيتم تمكيننا من تحرير الاسم:



نغير الاسم لـ Users ثم نضغط Enter من لوحة المفاتيح. نضغط بالزر الأيمن من الفأرة على الـ Users لنضيف خاصية جديدة من:

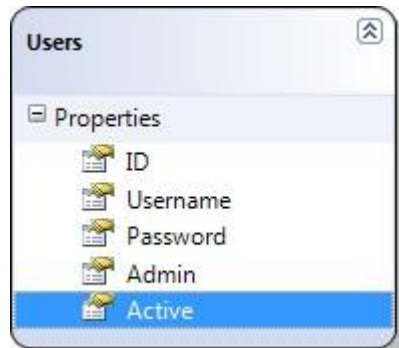
- Add
- Property

الآن نجد الخاصية الجديدة في حالة تحرير الاسم، نكتب ID ، ثم من لوحة المفاتيح نضغط Enter مرتين، ليتم اعتماد الاسم وإنشاء خاصية جديدة اسمها: Username



بعد أن أضفنا خاصية الـ Username إما أن نضغط من لوحة المفاتيح Enter ليتم إنشاء خاصية جديدة، وإما أن نضغط بالزر الأيمن للفأرة ونختار كما فعلنا أول مرة، لنضيف خاصية اسمها Password وخاصيتين أخريين اسمهما

Active: وAdmin



بقي علينا أن نضبط الخصائص، نحدد ID ثم نضغط F4 لننتقل لنافذة Properties أو بالضغط عليها بزر الفأرة الأيمن ثم اختيار Properties ، لنجعل خصائص الخاصية ID كما يلي:

```
Auto Generated Value = True
Auto-Sync = OnInsert
Primary Key = True
Read Only = True
Source = ID
Type = Integer (System.Int32)
```

خاصية Source مهمة جداً، فهي لتحديد العمود المرتبط من قاعدة البيانات بهذه الخاصية، إذا كان فارغاً سيكون هناك

خطأ في إنشاء قاعدة البيانات، ليتم تجاوز إنشاء هذا العمود ، وإكمال الباقي)

خصائص الخاصية: Username

```
Source = Username
```

النوع لن نغير فيه شيء لأنه من نوع نص **String** وهو ما نريد، لكن سيأخذ الحجم الافتراضي من طول النص في قاعدة البيانات وهو ٤٠٠٠ حرف، وهذا حجم كبير، لذلك هناك خاصية اسمها **Server Date Type** تمكنا من تحديد النوع، سنكتب فيها **nvarchar(20)** لنحدد طول النص بـ ٢٠ حرف كحد أقصى.

خصائص الخاصية: **Password:**

```
Source = Password
Server Date Type = nvarchar(20)
```

خصائص الخاصية: **Admin:**

```
Source = Admin
Type = Boolean (System.Boolean)
```

خصائص الخاصية: **Active:**

```
Source = Active
Type = Boolean (System.Boolean)
```

الآن وقد أتمنا هذه الفئة، سننشئ فئة جديدة اسمها **Persons**، فيها هذه الخصائص:

1. ID
2. Name
3. Address
4. Notes

خصائص الخاصية **ID** نجعلها كما في الفئة السابقة (يمكننا نسخ الخاصية **ID** من الفئة **Users** ولصقها في الفئة الجديدة لتحمل جميع الخصائص، دون عناء التغيير)، خصائص الخاصية **Name** كما في الـ **Username** من الفئة السابقة أيضاً، الخاصية **Address** فقط نزيد من قيمه عدد الحروف في خاصية **Server Date Type** لتساوي **nvarchar(50)**، والخاصية **Note** نغير فيها خاصية الـ **Source** لتكون نفس الاسم، كما فعلنا في السابق، وندع **Server Date Type** فارغة لتكون في أعلى قيمة وهي ٤٠٠٠ حرف، ولا ننسى أنها من المحتمل أن تكون فارغة (أي لا يتم اسناد أي قيم لها من المستخدم)، لذلك نضبط الخاصية **Nullable** لتساوي **True**.

الآن ننشئ فئة جديدة اسمها **PhoneNumbers**، فيها من الخصائص ما يلي:

ID وخصائصها كالسابق.

PersonID وخصائصها **Type = Integer (System.Int32)** ولا ننسى أن نضبط خاصية الـ **Source**

(هذه الخاصية للعلاقة ما بين الجدولين).

Phone وخصائصها **Server Date Type = nvarchar(10)**، ولا ننسى أيضاً خاصية الـ **Source**.

الآن نحدد فئة الـ **Users** ونذهب لنافذ الخصائص لنضع قيمة الخاصية **Source** بنفس اسم الفئة، وكذلك باقي الفئات.

الفئة **Users** الخاصية **Source = Users**

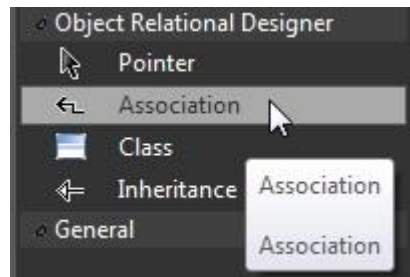
الفئة **Persons** الخاصية **Source = Persons**

الفئة **PhoneNumbers** الخاصية **Source = PhoneNumbers**

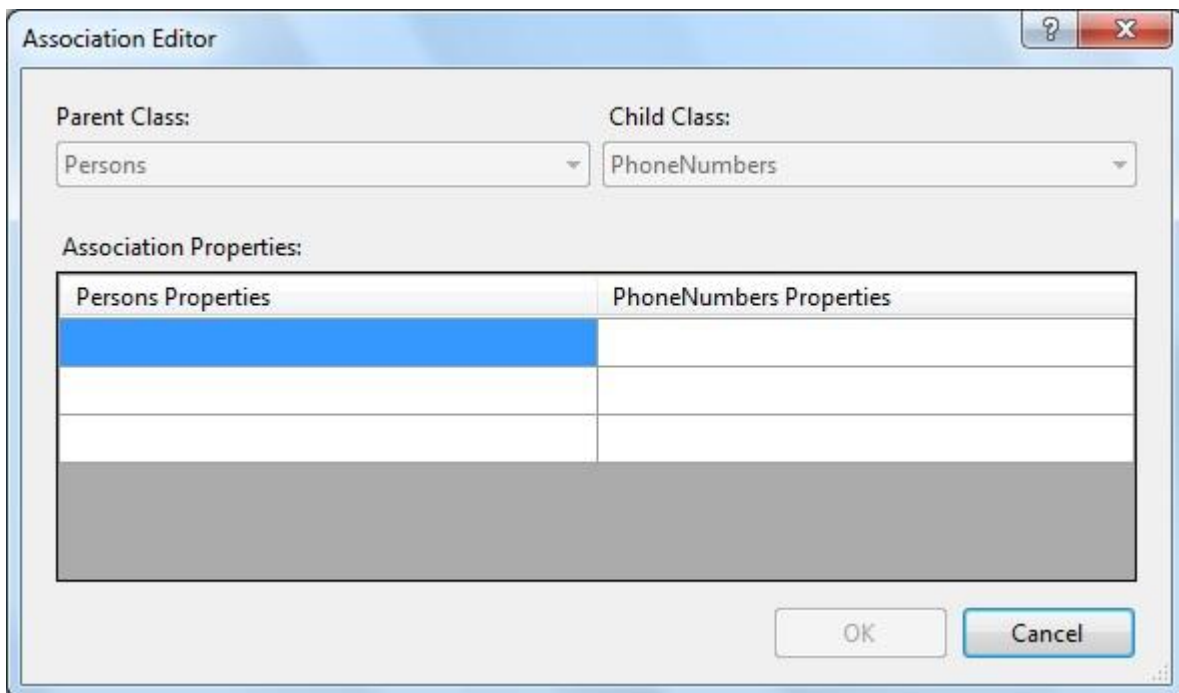
قد شارفنا على الانتهاء من العمل في قاعدة البيانات، بقي أن نربط جدول الأشخاص مع جدول الأرقام بعلاقة.

لننشئ العلاقة بين فئة **Persons** وفئة **PhoneNumbers** ، ولكن قبل ذلك لنعرف ما هي العلاقات. في بعض الأحيان يكون لشخص ما أكثر من رقم هاتف، قد يملك الشخص ثلاث هواتف أو أربعة أو عشرة، فليس هناك حصر، لهذا لا نستطيع أن نضيف خاصية رقم الهاتف في الفئة **Persons** ، لأنها ببساطة سوف تحمل رقم واحد فقط، فالحل يكون بإضافة جدول جديد يتم تخزين الأرقام فيه، والسؤال الذي يطرح نفسه الآن هو: كيف لنا أن نميز أن هذا الرقم هو لفلان وهذا الرقم لآخر؟ الإجابة هي بإنشاء علاقة بين الجدولين الرئيسيين وهو **Persons** والفرعي وهو **PhoneNumbers** ، من خلال ربط العمود **ID** الذي يتم توليده تلقائياً كلما أضفنا اسماً جديداً، بالعمود **PersonID** والذي من نفس النوع **Integer**، بهذا نكون قد عرفنا أن رقم الهاتف يعود لاسم المستخدم والذي رقمه هو مخزن في حقل **PersonID**.

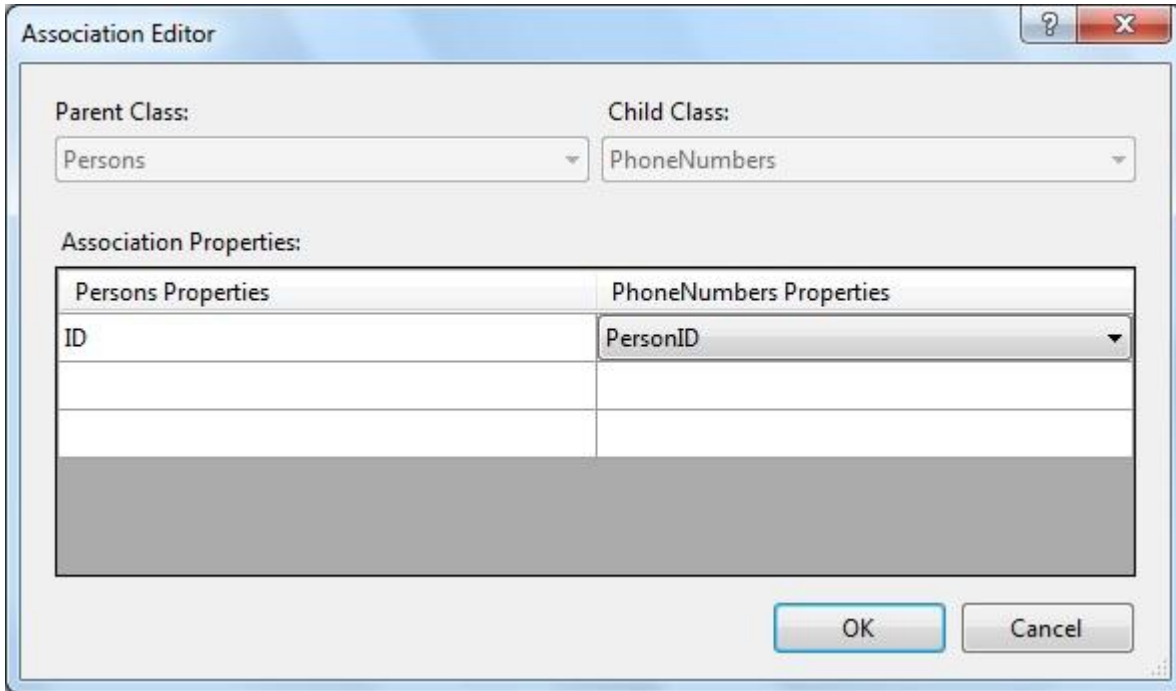
لننشئ العلاقة بالتوجه لنا هذه الأدوات وتحديد الأداة: **Association**:



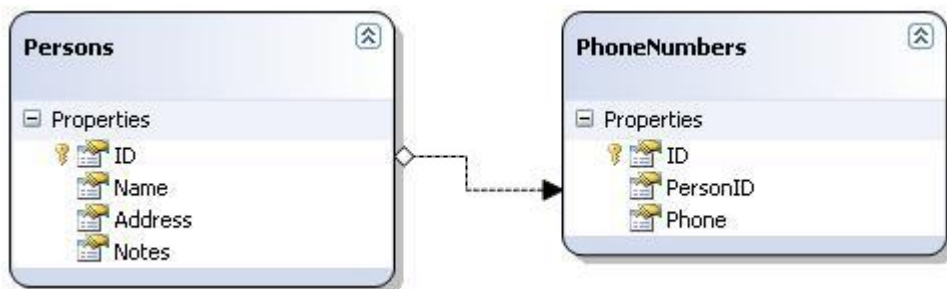
ثم النقر مرة على الفئة **Persons** والتوجه للفئة **PhoneNumbers** والنقر عليها أيضاً مرة واحدة، لنرى هذه النافذة:



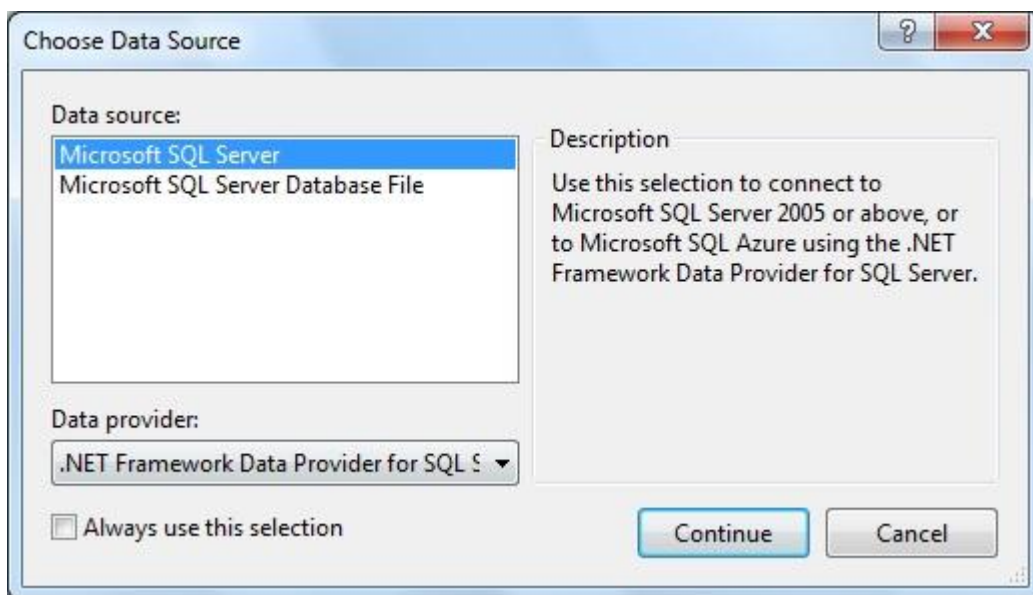
نختار من العمود الأول في الجدول من هذه النافذة، عمود **ID** ومن العمود الثاني نختار عمود **PersonID**:



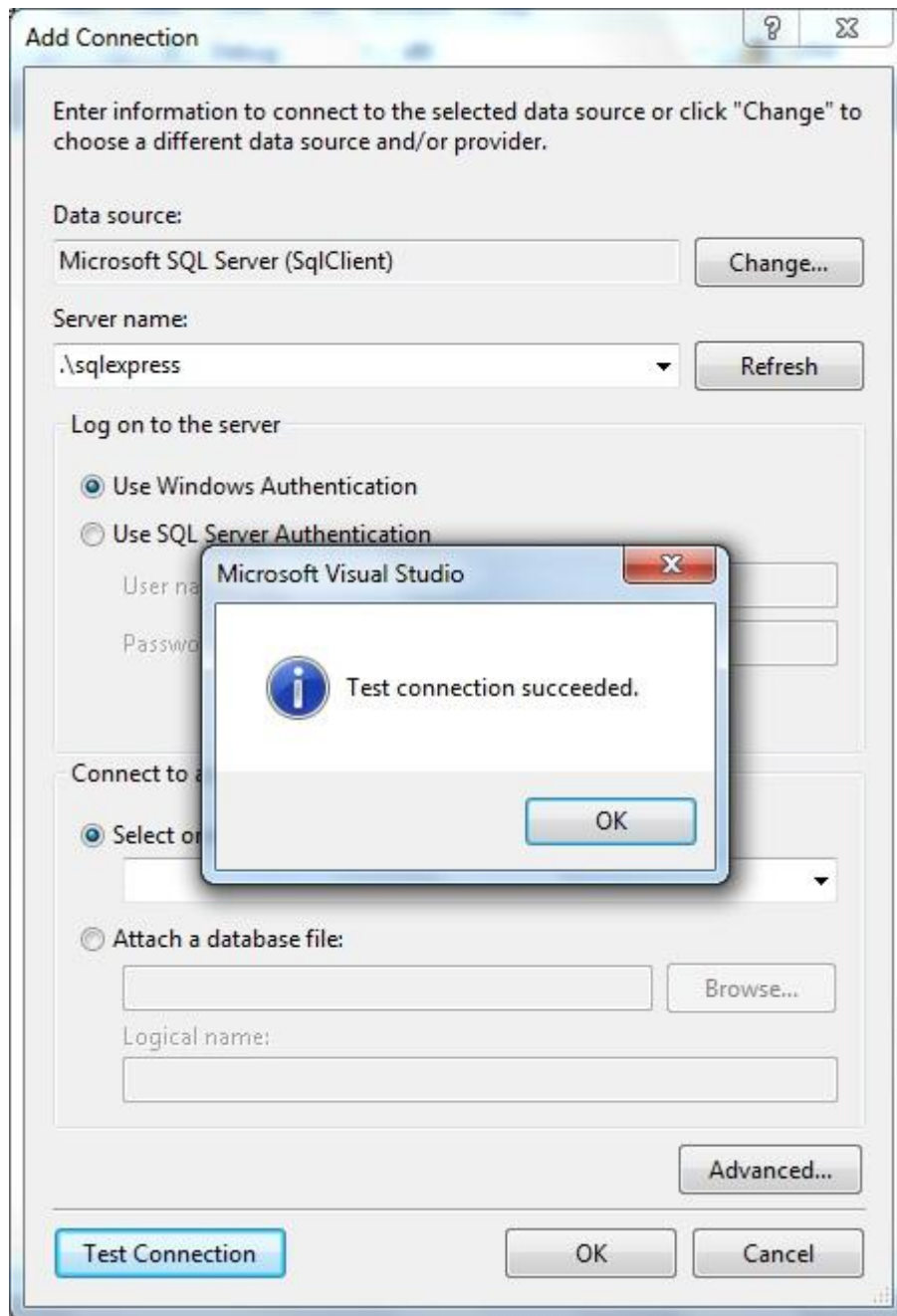
ثم موافق ليبدو الجدولين أو الفتتين كما يلي:



الآن نضغط بزر الفأرة الأيمن على مكان فارغ في المصمم ونختار Properties Context ، في خاصية Namespace نضع القيمة DB ، وفي خاصية Entity Namespace نضع القيمة DB أيضاً، وخاصية Name نجعلها PhoneBookDB ، ثم خاصية Connection نختار منها New لتطالعنا هذه النافذة:



نختار منها الأول وهو Microsoft SQL Server ، ثم Continue لنرى :



نكتب في الـ Combobox التابع لـ Server name: .sqlexpress ، ثم نختبر الإتصال بالضغط على زر Test Connection ، فإن نجح نضغط OK ، وإلا نراجع خدمة الـ SQL Server.

الآن وقد انتهينا من قاعدة البيانات، نغلق التبويب الخاص بها ثم نذهب لـ Solution Explorer ، ونختار منها Form1 لنفتحه في تبويب التصميم، ننقر عليه نقراً مزدوجاً لنذهب لتبويب الكود داخل حدث التحميل، نعرف متغير عام خارج أي حدث:

C#

كود

```
private DB.PhoneBookDB db = new DB.PhoneBookDB ();
```

VB

كود

```
Private db As New DB.PhoneBookDB ()
```


سيكون شكل الكود للآن كمان يلي:

C#

كود

```
public partial class MainForm : Form
{
    public MainForm()
    {
        InitializeComponent();
    }

    private DB.PhoneBookDB db = new DB.PhoneBookDB();

    private void MainForm_Load(object sender, EventArgs e)
    {
    }
}
```

VB

كود

```
Public Class MainForm

    Private db As New DB.PhoneBookDB()

    Private Sub MainForm_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

    End Sub
End Class
```

إلى الآن لم ننشئ قاعدة البيانات، لعمل ذلك نكتب في حدث التحميل:

C#

كود

```
if(!db.DatabaseExists())
try
{
    db.CreateDatabase();
}
catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error); return; }
```

VB

كود

```
If Not db.DatabaseExists() Then
    Try
        db.CreateDatabase()

    Catch ex As Exception
        MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
    Exit Sub
End Try
End If
```

اختبرنا ما إذا لم تكن قاعدة البيانات موجود، وعلى ذلك يتم إنشائها أو الخروج من الإجراء. إن حصل خطأ وقد يكون ذلك بسبب أن خدمة **SQL Server** متوقفة أو أننا لم نكمل خاصية الـ **Source** للفئات وخصائصها، فإن كانت الأخرى، كما قلنا سيتم إنشاء قاعدة البيانات، ولكن بتجاوز الفئة أو الخاصية التي لم يكتب قيمة لها في الـ **Source**، فإن أردنا حذف قاعدة البيانات نكتب:

C#

كود

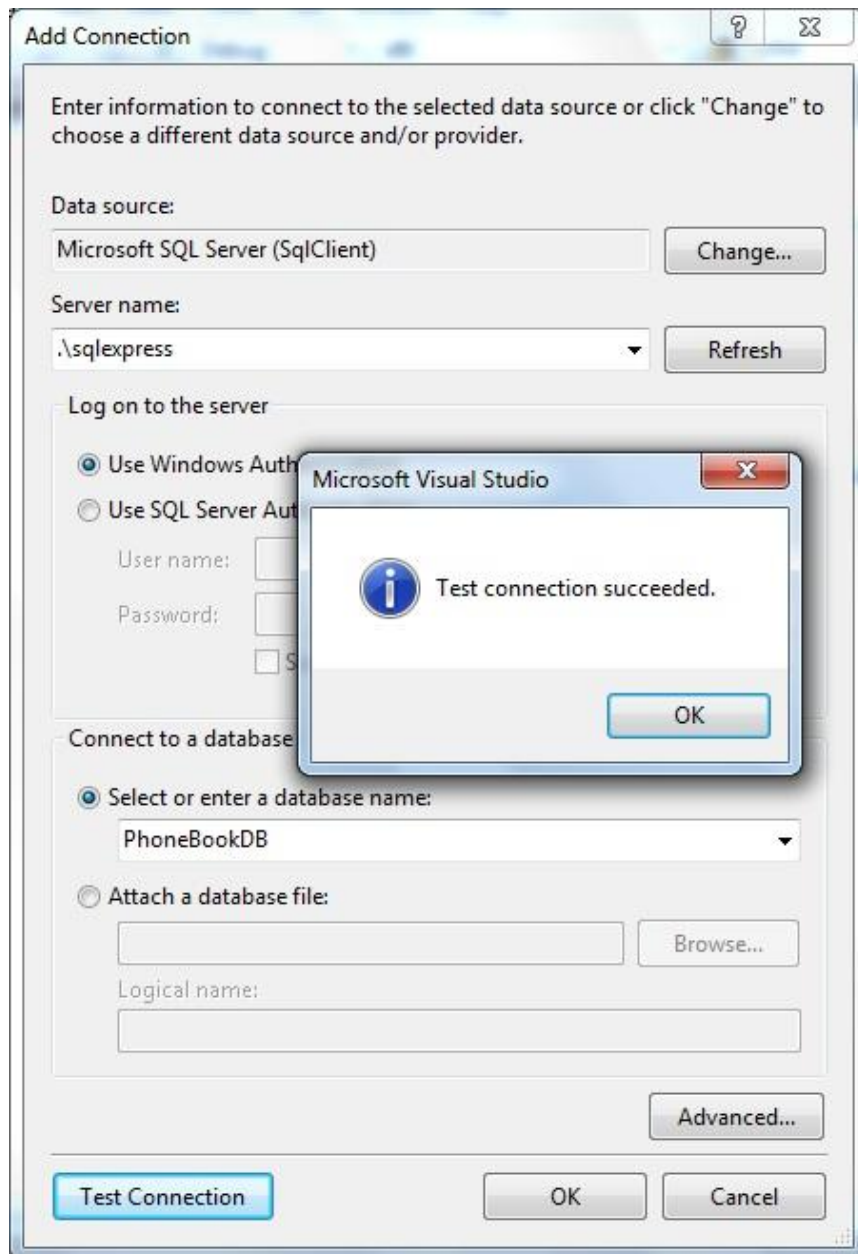
```
db.DeleteDatabase();
```

VB

كود

```
db.DeleteDatabase()
```

نعمل تشغيل للبرنامج ليتم إنشاء قاعدة البيانات، ثم نعود لفتح فئة محتوى البيانات من Solution Explorer والتي اسمها PhoneBookDB.dbml ، لنربط قاعدة البيانات مع محتوى البيانات، في أي مكان فارغ من المصمم نختار بعد الضغط بزر الفأرة الأيمن Properties ، ثم من الخصائص نختار New Connection ثم Microsoft SQL Server ثم Continue وبعد ذلك نكتب في اسم السيرفر \sql express . ، ثم نختار قاعدة البيانات من قائمة Select or enter a database name والتي اسمها: PhoneBookDB



نعود للنموذج بعد أن أتممنا بفضل الله إنشاء قاعدة البيانات وربطها مع الفئة، لنعمل مدير للبرنامج:

C#

كود

```
var users = from d in db.Users
            where d.Admin
            select d;
```

VB

كود

```
Dim users = From d In db.Users
            Where d.Admin
            Select d
```

هذه جملة التحديد في تقنية Linq وصيغتها كما رأينا، بعد كلمة From نكتب اسم متغير يمثل الجدول الذي نكتب اسمه بعد ذلك وهو هنا Users ، والشرط إذا كان Admin يساوي True ، نقوم بجلب جميع المستخدمين في حال كان المستخدم مدير للبرنامج، فإن كان ينتهي الإجراء وإلا سنعمل مدير افتراضي للبرنامج، لنختبر ذلك:

C#

كود

```
if (users.Count() == 0)
```

VB

كود

```
If users.Count() = 0 Then  
End If
```

إذا كان مجموع القيم المحددة من الجدول تساوي 0 ، فهذا يعني أنه لا يوجد مدير، لعمل مدير للبرنامج نكتب بين فكي جملة الشرط السابقة:

C#

كود

```
DB.Users admin = new DB.Users();  
admin.Username = "Admin";  
admin.Password = "Admin";  
admin.Admin = true;  
admin.Active = true;
```

VB

كود

```
Dim admin As New DB.Users  
admin.Username = "Admin"  
admin.Password = "Admin"  
admin.Admin = True  
admin.Active = True
```

في السطر الأول عرفنا متغير من الفئة Admin ، ثم بعد ذلك اسندنا قيم للخصائص المعرفة داخل الفئة، لأن لم يتم حفظ البيانات التي عملناها، لذلك نكمل على الكود السابق:

C#

كود

```
db.Users.InsertOnSubmit(admin);  
db.SubmitChanges();
```

VB

كود

```
db.Users.InsertOnSubmit(admin)  
db.SubmitChanges()
```

في السطر الأول استخدمنا الإجراء InsertOnSubmit التابع للخاصية Users وأرسلنا في الوسيطة المتغير الذي عرفناه فوق وهو admin ليتم إدخاله بقاعدة البيانات بعد أن نرسل أمر حفظ التغييرات، وأرسلنا التغييرات لقاعدة البيانات بالسطر db.SubmitChanges()

ليتم بذلك حفظ التغييرات. هكذا أنشأنا قاعدة البيانات كمان أنشأنا مدير للبرنامج.

C#

كود

```

public partial class MainForm : Form
{
    public MainForm()
    {
        InitializeComponent();
    }

    private DB.PhoneBookDB db = new DB.PhoneBookDB();

    private void MainForm_Load(object sender, EventArgs e)
    {
        if(!db.DatabaseExists())
        {
            try
            {
                db.CreateDatabase();
            }
            catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error); return; }

            try
            {
                var users = from d in db.Users
                    where d.Admin
                    select d;

                if (users.Count() == 0)
                {
                    DB.Users admin = new DB.Users();
                    admin.Username = "Admin";
                    admin.Password = "Admin";
                    admin.Admin = true;
                    admin.Active = true;

                    db.Users.InsertOnSubmit(admin);
                    db.SubmitChanges();
                }
            }
            catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error); }
        }
    }
}

```

VB

كود

```

Public Class MainForm
    Private db As New DB.PhoneBookDB()

    Private Sub MainForm_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
        If Not db.DatabaseExists() Then
            Try
                db.CreateDatabase()

            Catch ex As Exception
                MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
        End If

        Try
            Dim users = From d In db.Users
                Where d.Admin
                Select d

            If users.Count() = 0 Then
                Dim admin As New DB.Users
                admin.Username = "Admin"
                admin.Password = "Admin"
                admin.Admin = True
                admin.Active = True

                db.Users.InsertOnSubmit(admin)
                db.SubmitChanges()
            End If

        Catch ex As Exception
            MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
    End Sub
End Class

```

نعود الآن لتصميم النموذج، ونضع عليه أداة الـ **MenuStrip** وأداة الـ **TabControl** ، نضبط خاصية الـ **Dock** لأداة

الـ **TabControl** على القيمه: **Fill**



MenuStrip ستحتوي ما يلي:

- ملف:
- ١. **Name = newuser** : خاصية الـ **MenuStrip** مستخدم جديد: خاصية الـ **Name = newuser**
- ٢. **Name = deluser** : خاصية الـ **MenuStrip** حذف مستخدم: خاصية الـ **Name = deluser**
- ٣. **Name = edituser** : خاصية الـ **MenuStrip** تعديل البيانات: خاصية الـ **Name = edituser**
- ٤. **Name = signout** : خاصية الـ **MenuStrip** تسجيل الخروج: خاصية الـ **Name = signout**
- ٥. **Name = exitapp** : خاصية الـ **MenuStrip** إنهاء: خاصية الـ **Name = exitapp**

الآن سنعمل نموذج لتسجيل الدخول، من **Solution Explorer** نضغط على المشروع بالزر الأيمن للفأرة، ثم نختار:

- Add
- Windows Form

ليكن اسمه **LoginForm** ، خصائصه كما يلي:

```
FormBorderStyle = FixedDialog
MaximizeBox = False
MinimizeBox = False
ShowIcon = False
ShowInTaskbar = False
Text = تسجيل الدخول
```

نضيف من صندوق الأدوات أدوات **Label** وأداتي **TextBox** و **Button** ، خصائص الأدوات نضبطها كما يلي:

textBox1:

```
Name = unbox
```

textBox2:

```
Name = pasbox
PasswordChar = *
```

button1:

```
Text = "دخول"
```

في الكود الخاص بنموذج تسجيل الدخول، نعرف متغيرين عامين على النموذج:

*في C# نعرف ثلاث متغيرات.

C#

كود

```
private DB.PhoneBookDB db = new DB.PhoneBookDB();
internal bool endapp = true;
internal MainForm mf = null;
```

VB

كود

```
Private db As New DB.PhoneBookDB()
Friend endapp As Boolean = True
```

نضغط على زر الدخول مرتين بالفأرة لنذهب للحدث الافتراضي التابع له **Click** ، سوف نختبر قيم النص لأداتي مربع النص فيما إذا كانت فارغة:

C#

كود

```
if (String.IsNullOrEmpty(unbox.Text.Trim()) ||
String.IsNullOrEmpty(pasbox.Text.Trim()))
{
    MessageBox.Show("يجب إدخال اسم المستخدم وكلمة المرور", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation);
    return;
}
```

VB

كود

```
If String.IsNullOrEmpty(unbox.Text.Trim()) OrElse
String.IsNullOrEmpty(pasbox.Text.Trim()) Then
    MessageBox.Show("يجب إدخال اسم المستخدم وكلمة المرور", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
Exit Sub
End If
```

نتبعه الكود التالي:

C#

كود

```
try
{
    var user = from u in db.Users
                where u.Username == unbox.Text.Trim() && u.Password ==
pasbox.Text.Trim()
                select u;

    if (user.Count() != 0)
    {
        if (!user.SingleOrDefault().Active)
        {
            MessageBox.Show("لقد أدخلت اسم مستخدم موقوف", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation);
            return;
        }

        endapp = false;
        Close();
    }

    else MessageBox.Show("اسم المستخدم أو كلمة المرور خطأ", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation);
}
catch { }
```

```

Try
    Dim user = From u In db.Users
                Where u.Username = unbox.Text.Trim().ToLower() AndAlso u.Password =
pasbox.Text.Trim().ToLower()
                Select u

    If user.Count() <> 0 Then
        If Not user.SingleOrDefault().Active Then
            MessageBox.Show("لقد أدخلت اسم مستخدم موقوف", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
            Exit Sub
        End If

        endapp = False
        Close()

    Else
        MessageBox.Show("اسم المستخدم أو كلمة المرور خطأ", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
    End If

Catch

End Try

```

جملة تحديد تم شرحها سابقاً، وباقي الكود واضح، اللهم المتغير **endapp** سوف يتضح بعد كتابة الكود التالي في حدث

FormClosing

C#

كود

```
if (endapp) Application.Exit();
```

VB

كود

```
If endapp Then End
```

يجب علينا أن ننهي البرنامج في حالة إغلاق النموذج بدون تسجيل الدخول، كي لا يتم الدخول للبرنامج.

نعود للنموذج الرئيسي ونذهب لتبويب الكود التابع له، لنضيف إجراء كما يلي:

C#

كود

```

private void LogIn()
{
    tabControll1.SelectedIndex = 0;

    LoginForm lf = new LoginForm();
    lf.endapp = true;
    lf.mf = this;
    lf.unbox.Text = String.Empty;
    lf.pasbox.Text = String.Empty;
    lf.ShowDialog();
}

```

VB

كود

```

Private Sub LogIn()
    TabControll1.SelectedIndex = 0

    LoginForm.endapp = True
    LoginForm.unbox.Text = String.Empty
    LoginForm.pasbox.Text = String.Empty
    LoginForm.ShowDialog()
End Sub

```

في آخر كود إجراء التحميل للنموذج الرئيسي، تحت جملة الشرط نكتب:

C#

كود

```
LogIn();
```

VB

كود

```
LogIn()
```

C#

كود

```

private void MainForm_Load(object sender, EventArgs e)
{
    if (!db.DatabaseExists())
        try { db.CreateDatabase(); }
        catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ",
        MessageBoxButtons.OK, MessageBoxIcon.Error); return; }

    try
    {
        var users = from d in db.Users
                    where d.Admin
                    select d;

        if (users.Count() == 0)
        {
            DB.Users admin = new DB.Users();
            admin.Username = "Admin";
            admin.Password = "Admin";
            admin.Admin = true;
            admin.Active = true;

            db.Users.InsertOnSubmit(admin);
            db.SubmitChanges();
        }
    }
    catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ",
    MessageBoxButtons.OK, MessageBoxIcon.Error); }

    Login();
}

```

VB

كود

```

Private Sub MainForm_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    If Not db.DatabaseExists() Then
        Try
            db.CreateDatabase()

            Catch ex As Exception
                MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
                MessageBoxIcon.Error)
                Exit Sub
            End Try
        End If

        Try
            Dim users = From d In db.Users
                        Where d.Admin
                        Select d

            If users.Count() = 0 Then
                Dim admin As New DB.Users
                admin.Username = "Admin"
                admin.Password = "Admin"
                admin.Admin = True
                admin.Active = True

                db.Users.InsertOnSubmit(admin)
                db.SubmitChanges()
            End If

            Catch ex As Exception
                MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
                MessageBoxIcon.Error)
            End Try

            Login()
        End Sub

```

لنرى مشروعنا كيف يعمل الآن، نجرب تسجيل الدخول دون كتابة شيء في مربعات النصوص، نجرب أيضاً كتابة اسم مستخدم وكلمة مرور خطأ، وأيضاً نغلق النموذج لنرى أنه لا يعود للنموذج الرئيسي بل يخرج من البرنامج كامل، ولا ننسى أن ندخل معلومات صحيحة لنرى أنه يعمل بشكل جيد.

سوف ننهي خيارات المستخدم أولاً، لذا قبل أن نبدأ بها يجب علينا تحديد الصلاحيات، سوف نمنع المستخدمين من إضافة مستخدم جديد للبرنامج وحذف مستخدم من البرنامج، باستثناء المدير، نعود للكود داخل حدث ضغط زر تسجيل الدخول ونكتب داخل جملة الشرط الرئيسية وتحت جملة الشرط الفرعية:

C#

كود

```
mf.newuser.Enabled = user.SingleOrDefault().Admin;
mf.deluser.Enabled = mf.newuser.Enabled;
```

VB

كود

```
MainForm.newuser.Enabled = user.SingleOrDefault().Admin
MainForm.deluser.Enabled = MainForm.newuser.Enabled
```

SingleOrDefault دالة تعود بصف واحد من جملة التحديد، والصف يحتوي خاصية **Admin** التي أدرجناها سابقاً في فئة **Users**، أسندنا قيمتها لخاصية التمكين في خيار إضافة مستخدم جديد، كما أسندنا قيمة التمكين في خيار حذف المستخدم لتساوي قيمة التمكين في خيار إضافة مستخدم جديد، بقي أن نخزن اسم المستخدم لكي نعدل بياناته، دون الحاجة لكتابة اسمه الحالي وكلمة المرور الخاصة به الحالية في النموذج الخاص بذلك، سوف نضيف في النموذج الرئيسي خاصية نخزن بها اسم المستخدم:

C#

كود

```
private string un = String.Empty;
internal string Username
{
    get { return un; }
    set { un = value; }
}
```

VB

كود

```
Private un As String = String.Empty
Friend Property Username As String
    Get
        Return un
    End Get
    Set(ByVal value As String)
        un = value
    End Set
End Property
```

نذهب لنموذج تسجيل الدخول، لنكتب تحت سطري الصلاحيات:

C#

كود

```
mf.Username = unbox.Text.Trim();
```

VB

كود

```
MainForm.Username = unbox.Text.Trim()
```

C#

كود

```

public partial class LoginForm : Form
{
    public LoginForm()
    {
        InitializeComponent();
    }

    private DB.PhoneBookDB db = new DB.PhoneBookDB();
    internal bool endapp = true;
    internal MainForm mf = null;

    private void Button1_Click(object sender, EventArgs e)
    {
        if (String.IsNullOrEmpty(unbox.Text.Trim()) ||
String.IsNullOrEmpty(pasbox.Text.Trim()))
        {
            MessageBox.Show("يجب إدخال اسم المستخدم وكلمة المرور", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation);
            return;
        }

        try
        {
            var user = from u in db.Users
                where u.Username == unbox.Text.Trim() && u.Password ==
pasbox.Text.Trim()
                select u;

            if (user.Count() != 0)
            {
                if (!user.SingleOrDefault().Active)
                {
                    MessageBox.Show("لقد أدخلت اسم مستخدم موقوف", Text,
MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
                    return;
                }

                mf.newuser.Enabled = user.SingleOrDefault().Admin;
                mf.deluser.Enabled = mf.newuser.Enabled;
                mf.Username = unbox.Text.Trim();

                endapp = false;
                Close();
            }

            else MessageBox.Show("اسم المستخدم أو كلمة المرور خطأ", Text,
MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        }
        catch { }
    }

    private void LoginForm_FormClosing(object sender, FormClosingEventArgs e)
    {
        if (endapp) Application.Exit();
    }
}

```

```

Public Class LoginForm

    Private db As New DB.PhoneBookDB()
    Friend endapp As Boolean = True

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        If String.IsNullOrEmpty(unbox.Text.Trim()) OrElse
String.IsNullOrEmpty(pasbox.Text.Trim()) Then
            MessageBox.Show("يجب إدخال اسم المستخدم وكلمة المرور", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
            Exit Sub
        End If

        Try
            Dim user = From u In db.Users
                Where u.Username = unbox.Text.Trim().ToLower() AndAlso u.Password =
pasbox.Text.Trim().ToLower()
                Select u

            If user.Count() <> 0 Then
                If Not user.SingleOrDefault().Active Then
                    MessageBox.Show("لقد أدخلت اسم مستخدم موقوف", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
                    Exit Sub
                End If

                MainForm.newuser.Enabled = user.SingleOrDefault().Admin
                MainForm.deluser.Enabled = MainForm.newuser.Enabled
                MainForm.Username = unbox.Text.Trim()

                endapp = False
                Close()

            Else
                MessageBox.Show("اسم المستخدم أو كلمة المرور خطأ", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
            End If
        End Try
    End Sub
End Class

```

الآن لننشئ نموذج إضافة مستخدم جديد، وليكن اسمه **AddNewUser**، وخصائصه كما خصائص نموذج تسجيل الدخول، الأدوات التي سندرجها عليه هي: أدوات **Label** وأداتي **TextBox** وأداتي **CheckBox** و **Button**، نضبط مواقعهن، والخصائص كما يلي:

الأداة	خاصية الاسم Name	خاصية النص Text
Label1	-	اسم المستخدم
Label2	-	كلمة المرور
textBox1	unbox	-
textBox2	pasbox	-
checkBox1	admin	مدير
checkBox1	active	مفعّل
button1	-	حفظ

C#

كود

```

if (String.IsNullOrEmpty(unbox.Text.Trim()) || String.IsNullOrEmpty(pasbox.Text.Trim()))
{
    MessageBox.Show("يجب إكمال البيانات", Text, MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
    return;
}

DB.PhoneBookDB db = new DB.PhoneBookDB();

try
{
    var users = from u in db.Users
                where u.Username == unbox.Text.Trim()
                select u;

    if (users.Count() > 0)
    {
        MessageBox.Show("لقد أدخلت اسم موجود، قم باختيار اسم آخر", Text, MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
        return;
    }
}
catch { return; }

DB.Users user = new DB.Users();
user.Username = unbox.Text.Trim();
user.Password = pasbox.Text.Trim();
user.Admin = admin.Checked;
user.Active = active.Checked;

db.Users.InsertOnSubmit(user);

try
{
    db.SubmitChanges();
    MessageBox.Show("تم إضافة المستخدم", Text, MessageBoxButtons.OK, MessageBoxIcon.Information);

    unbox.Text = String.Empty;
    pasbox.Text = String.Empty;
    unbox.Focus();
}
catch (Exception ex) {
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
    MessageBoxIcon.Error);
}

```

VB

كود

```

If String.IsNullOrEmpty(unbox.Text.Trim()) OrElse String.IsNullOrEmpty(pasbox.Text.Trim())
Then
    MessageBox.Show("يجب إكمال البيانات", Text, MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
    Return
End If

Dim db As New DB.PhoneBookDB()

Try
    Dim users = From u In db.Users
                Where u.Username = unbox.Text.Trim()
                Select u

    If users.Count() > 0 Then
        MessageBox.Show("لقد أدخلت اسم موجود، قم باختيار اسم آخر", Text, MessageBoxButtons.OK, MessageBoxIcon.Warning)
        Return
    End If
Catch
    Exit Sub
End Try

Dim user As New DB.Users()
user.Username = unbox.Text.Trim()
user.Password = pasbox.Text.Trim()
user.Admin = admin.Checked
user.Active = active.Checked

db.Users.InsertOnSubmit(user)

Try
    db.SubmitChanges()
    MessageBox.Show("تم إضافة المستخدم", Text, MessageBoxButtons.OK, MessageBoxIcon.Information)
    unbox.Text = String.Empty
    pasbox.Text = String.Empty
    unbox.Focus()
Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try

```

الآن نضيف نموذج جديد للمشروع ونسميه `EditUser` ، نضبط خصائصه كما نموذج تسجيل الدخول، ونلقي عليه أداة `Label` وأداة `ComboBox` وأداتي `CheckBox` وأداتي `Button` ، نضبط مواقع الأدوات، والخصائص نجعلها كما يلي:

الأداة	خاصية الاسم Name	خاصية النص Text
Label1	-	اسم المستخدم
comboBox1	unlist	-
checkBox1	admin	مدير
checkBox1	active	مفعّل
button1	-	حفظ

أداة الـ `comboBox1` نعدّل فيها خاصيتي:

`AutoCompleteMode = SuggestAppend`
`AutoCompleteSource = ListItems`

المتغير المعتاد: (عام خارج أي إجراء)

C#

كود

```
private DB.PhoneBookDB db = new DB.PhoneBookDB();
```

VB

كود

```
Private db As New DB.PhoneBookDB()
```

ثم ننتقل لحدث تحميل النموذج، لنكتب:

C#

كود

```
try
{
    var users = from u in db.Users
                select u;

    unlist.DataSource = users;
    unlist.DisplayMember = "Username";

    unlist_SelectedIndexChanged(sender, e);
}
catch { }
```

VB

كود

```
Try
    Dim users = From u In db.Users
                Select u

    unlist.DataSource = users
    unlist.DisplayMember = "Username"

    unlist_SelectedIndexChanged(sender, e)

Catch

End Try
```

قمنا بجلب بيانات المستخدمين، من خلال جملة التحديد (والتي تم شرحها فيما سبق)، ثم اسندنا البيانات للقائمة من خلال خاصية `DataSource`، ثم استعرضنا فيها اسماء المستخدمين (لأنها تحتوي على أكثر من صف، نريد استعراض الاسم فقط)، من خلال الخاصية `DisplayMember`.

في حدث `SelectedIndexChanged` التابع للأداة `unlist`

C#

كود

```
try
{
    DB.Users user = (DB.Users)unlist.SelectedValue;
    admin.Checked = user.Admin;
    active.Checked = user.Active;
}
catch { }
```

VB

كود

```
Try
    Dim user As DB.Users = CType(unlist.SelectedValue, DB.Users)
    admin.Checked = user.Admin
    active.Checked = user.Active

Catch

End Try
```

وفي حدث زر الحفظ:

C#

كود

```
try
{
    DB.Users user = (DB.Users)unlist.SelectedValue;
    user.Admin = admin.Checked;
    user.Active = active.Checked;

    db.SubmitChanges();
    MessageBox.Show("تم تحديث البيانات", Text, MessageBoxButtons.OK,
    MessageBoxIcon.Information);
}
catch { }
```

VB

كود

```
Try
    Dim user As DB.Users = CType(unlist.SelectedValue, DB.Users)
    user.Admin = admin.Checked
    user.Active = active.Checked

    db.SubmitChanges()
    MessageBox.Show("تم تحديث البيانات", Text, MessageBoxButtons.OK,
    MessageBoxIcon.Information)

Catch

End Try
```

السطر الأول عرفنا متغير من الفئة `Users`، وقد أخذنا القيمة من قائمة الأسماء، والتي تحتوي على البيانات التي أدرجناها في حدث التحميل، وخاصية الـ `SelectedValue` التابعة للأداة، ما هي إلا صف يمثل البيانات.

C#

كود

```

        if (MessageBox.Show("هل أنت متأكد", Text, MessageBoxButtons.YesNo,
        MessageBoxIcon.Question) == DialogResult.No) return;

        try
        {
            db.Users.DeleteOnSubmit((DB.Users)unlist.SelectedValue);
            db.SubmitChanges();

            MessageBox.Show("تم حذف المستخدم", Text, MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        }
        catch { }
    
```

VB

كود

```

        If MessageBox.Show("هل أنت متأكد", Text, MessageBoxButtons.YesNo,
        MessageBoxIcon.Question) = DialogResult.No Then Exit Sub

        Try
            db.Users.DeleteOnSubmit(CType(unlist.SelectedValue, DB.Users))
            db.SubmitChanges()

            MessageBox.Show("تم حذف المستخدم", Text, MessageBoxButtons.OK,
        MessageBoxIcon.Information)

        Catch ex As Exception

        End Try
    
```

في النموذج الرئيسي، ادرجنا قائمة ووضعنا فيها بعض الخيارات، منها خيار (حذف مستخدم)، غير النص ب: تعديل مستخدم.

اضغط مرتين على خيار (مستخدم جديد) في القائمة، لننتقل للكود ونكتب:

C#

كود

```

AddNewUser anu = new AddNewUser();
anu.ShowDialog();
    
```

VB

كود

```

AddNewUser.ShowDialog()
    
```

ثم اضغط على خيار (تعديل مستخدم) من القائمة، سنكتب:

C#

كود

```

EditUsers eu = new EditUsers();
eu.ShowDialog();
    
```

VB

كود

```

EditUsers.ShowDialog()
    
```

نضيف نموذج جديد للمشروع اسمه EditData ، نضع عليه أدوات: ٣ Label و ٣ TextBox و Button ، خصائص

الأدوات:

الأداة	خاصية الاسم Name	خاصية النص Text
label1	-	كلمة المرور القديمة
label1	-	كلمة المرور الجديدة
label1	-	تأكيد كلمة المرور
textBox1	opbox	-
textBox2	npbox	-
textBox3	vnbox	-
button1	-	حفظ

نضغط مرتين على زر الحفظ، ثم نكتب:

C#

كود

```

DB.PhoneBookDB db = new DB.PhoneBookDB();

if (String.IsNullOrEmpty(opbox.Text.Trim()) ||
String.IsNullOrEmpty(npbox.Text.Trim()) ||
String.IsNullOrEmpty(vnbox.Text.Trim()))
{ MessageBox.Show("يجب إكمال البيانات", Text, MessageBoxButtons.OK, MessageBoxIcon.Warning);
return; }

if (npbox.Text.Trim() != vnbox.Text.Trim())
{ MessageBox.Show("كلمة المرور غير متطابقة", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation); return; }

try
{
var user = (from u in db.Users
where u.Username == mf.Username
select u).SingleOrDefault();

if (user.Password != opbox.Text.Trim())
{ MessageBox.Show("كلمة السر خطأ، أعد الكتابة", Text, MessageBoxButtons.OK,
MessageBoxIcon.Exclamation); return; }

user.Password = npbox.Text.Trim();

db.SubmitChanges();

MessageBox.Show("البيانات تم تعديل", Text, MessageBoxButtons.OK,
MessageBoxIcon.Information);
}
catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error); }

```


VB

كود

```

Dim db As New DB.PhoneBookDB()
If String.IsNullOrEmpty(opbox.Text.Trim()) OrElse String.IsNullOrEmpty(npbox.Text.Trim())
OrElse String.IsNullOrEmpty(vnpbox.Text.Trim()) Then
    MessageBox.Show("يجب إكمال البيانات", Text, MessageBoxButtons.OK, MessageBoxIcon.Warning)
    Exit Sub
End If
If npbox.Text.Trim() <> vnpbox.Text.Trim() Then
    MessageBox.Show("كلمة المرور غير متطابقة", Text, MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
    Exit Sub
End If
Try
    Dim user = (From u In db.Users
                Where u.Username = MainForm.Username
                Select u).SingleOrDefault()

    If user.Password <> opbox.Text.Trim() Then
        MessageBox.Show("كلمة السر خطأ، أعد الكتابة", Text, MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
    End If

    user.Password = npbox.Text.Trim()
    db.SubmitChanges()
    MessageBox.Show("تم تعديل البيانات", Text, MessageBoxButtons.OK, MessageBoxIcon.Information)
Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try

```

نعود للنموذج الرئيسي، ثم من القائمة التي أنشأناها، نضغط مرتين على خيار تعديل البيانات:

C#

كود

```

EditData ed = new EditData();
ed.mf = this;
ed.ShowDialog();

```

VB

كود

```

EditUsers.ShowDialog()

```

مستخدمي #C في نموذج تعديل البيانات قم بتعريف متغير عام في التصريحات العامة:

C#

كود

```

internal MainForm mf = null;

```

من القائمة في النموذج الرئيسي، نضغط على تسجيل الخروج أيضا، لنكتب:

C#

كود

```

LogIn();

```

VB

كود

```

LogIn()

```

وفي (إنهاء) من القائمة، نكتب:

C#

كود

```

Close();

```

VB

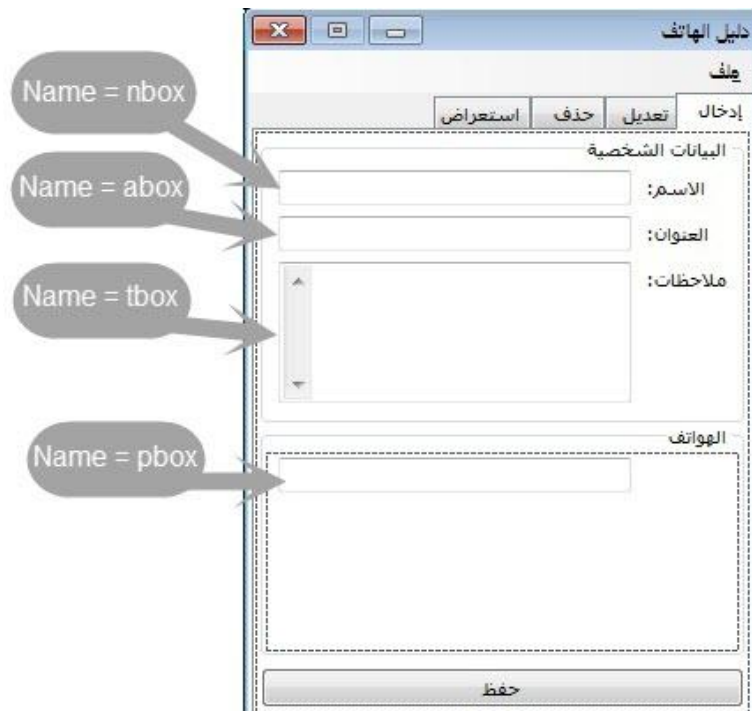
كود

```

Close()

```

نضع في تبويب (إدخال)، أدواتي **GroupBox** وأداة **Button** ، داخل الـ **groupBox1** نضع **Label** ٢ و **textBox** ٢ ، داخل الـ **groupBox2** نضع أداة **Panel** وداخل الـ **Panel** نضع **TextBox**. نضبطه ليكون هكذا :



بما أن رقم الهاتف قد يكون غير محدود، سوف نجعل الـ **textbox** المخصص لإدخال الرقم والذي اسمناه **pbox** يتضاعف عند كتابة رقم جديد، ومن ثم نحفظ البيانات دفعة واحدة في القاعدة، نذهب لحدث **TextChanged** التابع لمربع نص الرقم لنكتب فيه :

C#

كود

```
if (plist.Controls[plist.Controls.Count - 1].Focused && plist.Controls[plist.Controls.Count - 1].Text.Length == 1)
{
    TextBox tb = new TextBox();
    tb.Width = pbox.Width;
    tb.Location = new Point(pbox.Location.X, (pbox.Top + pbox.Height + 5) *
plist.Controls.Count);
    tb.TextChanged += new EventHandler(pbox_TextChanged);
    tb.KeyPress += new KeyPressEventHandler(pbox_KeyPress);
    plist.Controls.Add(tb);
}
```

VB

كود

```
If plist.Controls(plist.Controls.Count - 1).Focused AndAlso
plist.Controls(plist.Controls.Count - 1).Text.Length = 1 Then
    Dim tb As New TextBox()
    tb.Width = pbox.Width
    tb.Location = New Point(pbox.Location.X, (pbox.Top + pbox.Height + 5) *
plist.Controls.Count)
    AddHandler tb.TextChanged, AddressOf pbox_TextChanged
    AddHandler tb.KeyPress, AddressOf pbox_KeyPress
    plist.Controls.Add(tb)
End If
```

قمنا بإنشاء مربع نص جديد، ووضعناه في الـ **plist** ، بعد ضبط خصائصه ومكانه، ثم أضفناه له أحداث كي نستطيع إضافة مربع نص آخر عند كل إضافة رقم جديد. نريد تحديد الإدخال في مربع نص الرقم، ليقبل أرقام فقط، ولذلك نذهب لحدث الـ **KeyPress** التابع له لنكتب:

C#

كود

```
e.Handled = !char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar);
```

VB

كود

```
e.Handled = Not Char.IsDigit(e.KeyChar) AndAlso Not Char.IsControl(e.KeyChar)
```

C#

كود

```

if (string.IsNullOrEmpty(nbox.Text.Trim()) || string.IsNullOrEmpty(abox.Text.Trim()))
{ MessageBox.Show("يجب اكتمال البيانات", Text, MessageBoxButtons.OK, MessageBoxIcon.Warning); return; }

DB.Persons person = new DB.Persons();
person.Name = nbox.Text.Trim();
person.Address = abox.Text.Trim();
person.Note = tbox.Text.Trim();

try { db.Persons.InsertOnSubmit(person); db.SubmitChanges(); }
catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error); return; }

foreach (Control tb in plist.Controls)
    if (tb is TextBox && !string.IsNullOrEmpty(tb.Text))
    {
        DB.PhoneNumbers pn = new DB.PhoneNumbers();
        pn.PersonID = person.ID;
        pn.Phone = tb.Text;

        db.PhoneNumbers.InsertOnSubmit(pn);
    }

try { db.SubmitChanges(); } catch { }

MessageBox.Show("تمت الاضافة بنجاح", Text, MessageBoxButtons.OK, MessageBoxIcon.Information);

nbox.Text = string.Empty;
abox.Text = string.Empty;
tbox.Text = string.Empty;
pbox.Text = string.Empty;

for (int i = 1; i <= 2; ++i)
    foreach (Control ctrl in plist.Controls)
        if (ctrl.Name != "pbox") plist.Controls.Remove(ctrl);

GetData();

```

VB

كود

```

If String.IsNullOrEmpty(nbox.Text.Trim()) OrElse String.IsNullOrEmpty(abox.Text.Trim()) Then
    MessageBox.Show("يجب اكتمال البيانات", Text, MessageBoxButtons.OK, MessageBoxIcon.Warning)
    Return
End If
Dim person As New DB.Persons()
person.Name = nbox.Text.Trim()
person.Address = abox.Text.Trim()
person.Note = tbox.Text.Trim()

Try
    db.Persons.InsertOnSubmit(person)
    db.SubmitChanges()
Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
    Exit Sub
End Try

For Each tb As Control In plist.Controls
    If TypeOf (tb) Is TextBox AndAlso Not String.IsNullOrEmpty(tb.Text) Then

        Dim pn As New DB.PhoneNumbers()
        pn.PersonID = person.ID
        pn.Phone = tb.Text

        db.PhoneNumbers.InsertOnSubmit(pn)
    End If
Next
Try
    db.SubmitChanges()

Catch
End Try

MessageBox.Show("تمت الاضافة بنجاح", Text, MessageBoxButtons.OK, MessageBoxIcon.Information)

nbox.Text = String.Empty
abox.Text = String.Empty
tbox.Text = String.Empty
pbox.Text = String.Empty

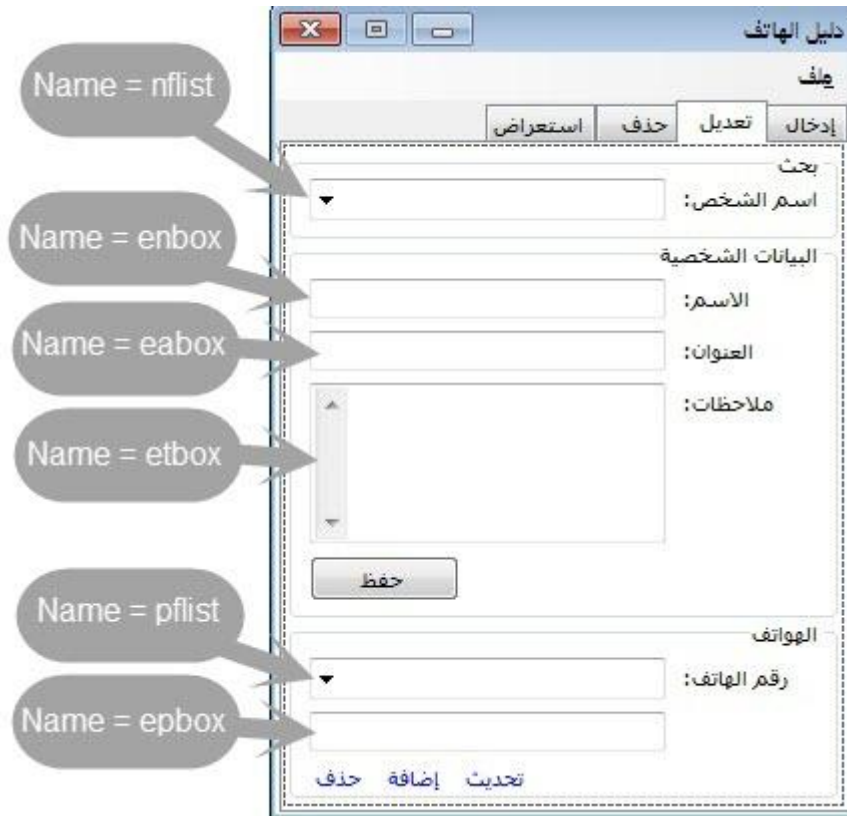
For i As Integer = 1 To 2
    For Each ctrl As Control In plist.Controls
        If (ctrl.Name <> "pbox") Then plist.Controls.Remove(ctrl)
    Next
Next

GetData()

```

GetData() إجراء سوف نضيفه فيما بعد إن شاء الله.

نذهب لتبويب (تعديل)، ونجعله كهذا الشكل:



في حدث الـ `SelectedIndexChanged` التابع للأداة `nflist` نكتب:

C#

كود

```
try
{
    DB.Persons person = (DB.Persons)nflist.SelectedValue;
    enbox.Text = person.Name;
    eabox.Text = person.Address;
    etbox.Text = person.Note;
    pflist.SelectedIndexChanged(sender, e);
}
catch { }
```

كود

VB

```
Try
    Dim person As DB.Persons = CType(nflist.SelectedValue, DB.Persons)
    enbox.Text = person.Name
    eabox.Text = person.Address
    etbox.Text = person.Note
    pflist.SelectedIndexChanged(sender, e)
Catch
End Try
```

في حدث زر (حفظ)، نكتب:

C#

كود

```
try
{
    DB.Persons person = (DB.Persons)nflist.SelectedValue;
    person.Name = enbox.Text;
    person.Address = eabox.Text;
    person.Note = etbox.Text;

    db.SubmitChanges();

    MessageBox.Show("تم تحديث البيانات", Text, MessageBoxButtons.OK, MessageBoxIcon.Information);
}
catch (Exception ex) { MessageBox.Show(ex.Message, "خطا", MessageBoxButtons.OK, MessageBoxIcon.Error); }
```

VB

كود

```

Try
    Dim person As DB.Persons = CType(nflist.SelectedValue, DB.Persons)
    person.Name = enbox.Text
    person.Address = eabox.Text
    person.Note = etbox.Text

    db.SubmitChanges()

    MessageBox.Show("تم تحديث البيانات", Text, MessageBoxButtons.OK,
    MessageBoxIcon.Information)

Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try

```

في حدث الـ SelectedIndexChanged التابع للأداة pflist، نكتب:

C#

كود

```
epbox.Text = pflist.Text;
```

VB

كود

```
epbox.Text = pflist.Text
```

الآن نأتي للـ LinkLabel، في (تحديث) نكتب:

C#

كود

```

try
{
    DB.PhoneNumbers pn = (DB.PhoneNumbers)pflist.SelectedValue;
    pn.Phone = epbox.Text.Trim();

    db.SubmitChanges();

    MessageBox.Show("تم تحديث الرقم", Text, MessageBoxButtons.OK,
    MessageBoxIcon.Information);
}
catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
    MessageBoxIcon.Error); }

```

VB

كود

```

Try
    Dim pn As DB.PhoneNumbers = CType(pflist.SelectedValue, DB.PhoneNumbers)
    pn.Phone = epbox.Text.Trim()

    db.SubmitChanges()

    MessageBox.Show("تم تحديث الرقم", Text, MessageBoxButtons.OK, MessageBoxIcon.Information)

Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try

```

في (إضافة)، نكتب:

C#

كود

```

try
{
    DB.PhoneNumbers pn = new DB.PhoneNumbers();
    pn.PersonID = ((DB.Persons)nflist.SelectedValue).ID;
    pn.Phone = epbox.Text.Trim();

    db.PhoneNumbers.InsertOnSubmit(pn);
    db.SubmitChanges();

    MessageBox.Show("تمت إضافة الرقم", Text, MessageBoxButtons.OK, MessageBoxIcon.Information);
}
catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
    MessageBoxIcon.Error); }

```

```

Try
    Dim pn As New DB.PhoneNumbers()
    pn.PersonID = CType(nflist.SelectedValue, DB.Persons).ID
    pn.Phone = epbox.Text.Trim()

    db.PhoneNumbers.InsertOnSubmit(pn)
    db.SubmitChanges()

    MessageBox.Show("تمت إضافة الرقم", Text, MessageBoxButtons.OK,
    MessageBoxIcon.Information)

Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)

End Try

```

في (حذف) نكتب:

C#

كود

```

if (MessageBox.Show("هل أنت متأكد", Text, MessageBoxButtons.YesNo, MessageBoxIcon.Question) ==
DialogResult.No) return;

try
{
    db.PhoneNumbers.DeleteOnSubmit((DB.PhoneNumbers)pflist.SelectedValue);
    db.SubmitChanges();

    MessageBox.Show("تم حذف الرقم", Text, MessageBoxButtons.OK, MessageBoxIcon.Information);
}
catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error); }

```

VB

كود

```

If MessageBox.Show("هل أنت متأكد", Text, MessageBoxButtons.YesNo, MessageBoxIcon.Question) =
DialogResult.No Then Exit Sub

Try
    db.PhoneNumbers.DeleteOnSubmit(CType(pflist.SelectedValue, DB.PhoneNumbers))
    db.SubmitChanges()

    MessageBox.Show("تم حذف الرقم", Text, MessageBoxButtons.OK, MessageBoxIcon.Information)

Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)

End Try

```

هكذا انتهينا من (إدخال وتعديل)، نتوجه لتبويب (حذف)، سوف يكون كهذا:



C#

كود

```

if (MessageBox.Show("هل أنت متأكد", Text, MessageBoxButtons.YesNo, MessageBoxIcon.Question) ==
DialogResult.No) return;
try
{
    var pn = from d in db.PhoneNumbers
              where d.PersonID == ((DB.Persons)dnflist.SelectedValue).ID
              select d;

    foreach (var p in pn) db.PhoneNumbers.DeleteOnSubmit(p);
    db.Persons.DeleteOnSubmit((DB.Persons)dnflist.SelectedValue);
    db.SubmitChanges();

    MessageBox.Show("تم حذف البيانات", Text, MessageBoxButtons.OK, MessageBoxIcon.Information);

    GetData();
}
catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error); }

```

VB

كود

```

If MessageBox.Show("هل أنت متأكد", Text, MessageBoxButtons.YesNo, MessageBoxIcon.Question) =
DialogResult.No Then Exit Sub
Try
    Dim pn = From d In db.PhoneNumbers
              Where d.PersonID = CType(nflist.SelectedValue, DB.Persons).ID
              Select d

    For Each p In pn
        db.PhoneNumbers.DeleteOnSubmit(p)
    Next

    db.Persons.DeleteOnSubmit(CType(nflist.SelectedValue, DB.Persons))
    db.SubmitChanges()

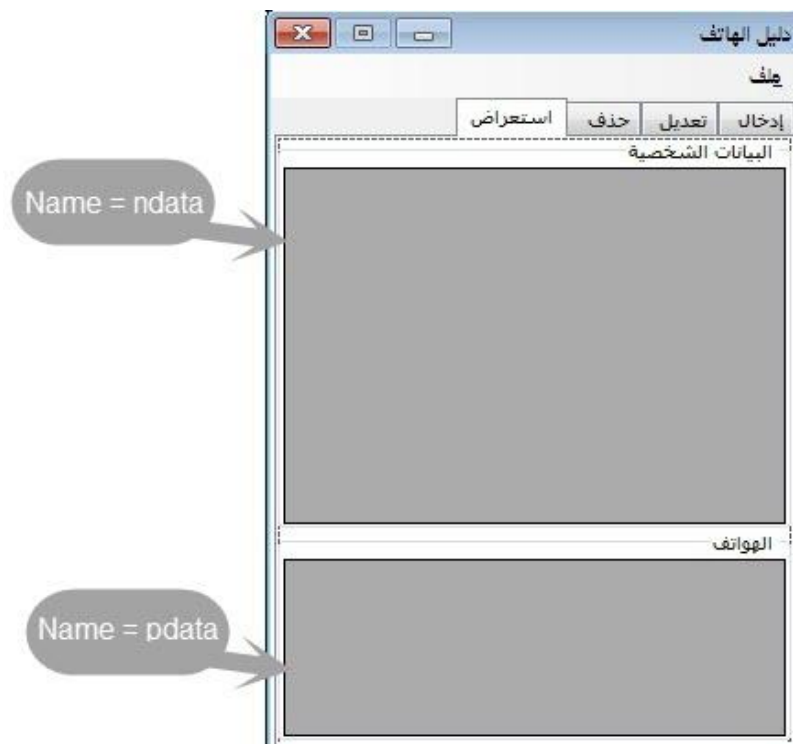
    MessageBox.Show("تم حذف البيانات", Text, MessageBoxButtons.OK, MessageBoxIcon.Information)

    GetData()

Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try

```

نذهب لتبويب (استعراض)، سيكون كما يلي:



C#

كود

```

private void GetData()
{
    try
    {
        var persons = from d in db.Persons
                      select d;

        BindingSource bsn = new BindingSource();
        bsn.DataSource = persons;

        BindingSource bsp = new BindingSource();
        bsp.DataSource = bsn;
        bsp.DataMember = "PhoneNumbers";
        //
        // تعديل البيانات
        //
        nflist.DataSource = bsn;
        nflist.DisplayMember = "Name";

        pflist.DataSource = bsp;
        pflist.DisplayMember = "Phone";
        //
        // حذف البيانات
        //
        dnflist.DataSource = bsn;
        dnflist.DisplayMember = "Name";
        //
        // استعراض البيانات
        //
        ndata.DataSource = null;
        ndata.Columns.Clear();
        ndata.AutoGenerateColumns = false;
        ndata.ColumnCount = 3;
        ndata.DataSource = bsn;

        ndata.Columns[0].HeaderText = "الاسم";

        ndata.Columns[1].HeaderText = "العنوان";

        ndata.Columns[2].HeaderText = "ملاحظات";

        ndata.Columns[0].DataPropertyName = "Name";
        ndata.Columns[1].DataPropertyName = "Address";
        ndata.Columns[2].DataPropertyName = "Note";

        pdata.DataSource = null;
        pdata.Columns.Clear();
        pdata.AutoGenerateColumns = false;
        pdata.ColumnCount = 1;
        pdata.DataSource = bsp;

        pdata.Columns[0].HeaderText = "رقم الهاتف";

        pdata.Columns[0].DataPropertyName = "Phone";
    }
    catch { }
}

```



```
Private Sub GetData()  
    Try  
        Dim persons = From d In db.Persons  
                        Select d  
  
        Dim bsn As New BindingSource()  
        bsn.DataSource = persons  
  
        Dim bsp As New BindingSource()  
        bsp.DataSource = bsn  
        bsp.DataMember = "PhoneNumbers"  
        '  
        ' تعديل البيانات  
        '  
        nflist.DataSource = bsn  
        nflist.DisplayMember = "Name"  
  
        pflist.DataSource = bsp  
        pflist.DisplayMember = "Phone"  
        '  
        '  
        ' حذف البيانات  
        '  
        dnflist.DataSource = bsn  
        dnflist.DisplayMember = "Name"  
        '  
        '  
        ' استعراض البيانات  
        '  
        ndata.DataSource = Nothing  
        ndata.Columns.Clear()  
        ndata.AutoGenerateColumns = False  
        ndata.ColumnCount = 3  
        ndata.DataSource = bsn  
  
        ndata.Columns(0).HeaderText = "الاسم"  
  
        ndata.Columns(1).HeaderText = "العنوان"  
  
        ndata.Columns(2).HeaderText = "ملاحظات"  
  
        ndata.Columns(0).DataPropertyName = "Name"  
        ndata.Columns(1).DataPropertyName = "Address"  
        ndata.Columns(2).DataPropertyName = "Note"  
  
        pdata.DataSource = Nothing  
        pdata.Columns.Clear()  
        pdata.AutoGenerateColumns = False  
        pdata.ColumnCount = 1  
        pdata.DataSource = bsp  
  
        pdata.Columns(0).HeaderText = "رقم الهاتف"  
        pdata.Columns(0).DataPropertyName = "Phone"  
  
    Catch  
    End Try  
End Sub
```

في حدث التحميل للنموذج، في آخر الكود تحت سطر Login() نكتب:

C#

كود

```
GetData();
```

VB

كود

```
GetData
```

سوف نضيف خيارات جديدة على القائمة MenuStip كالتالي:

تقرير

• كل الأسماء

• محدد

قاعدة البيانات

• نسخ احتياطي

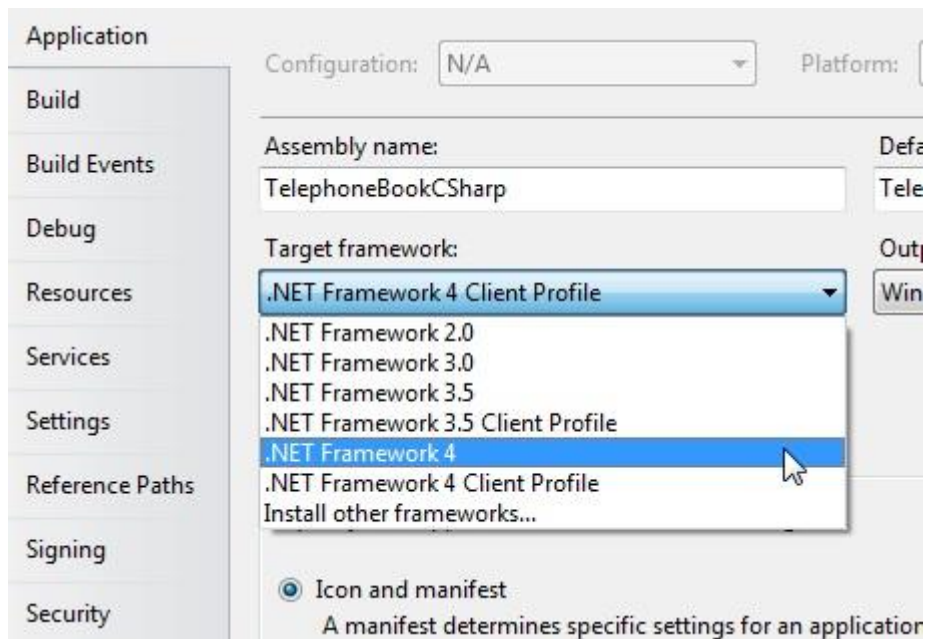
• استرجاع

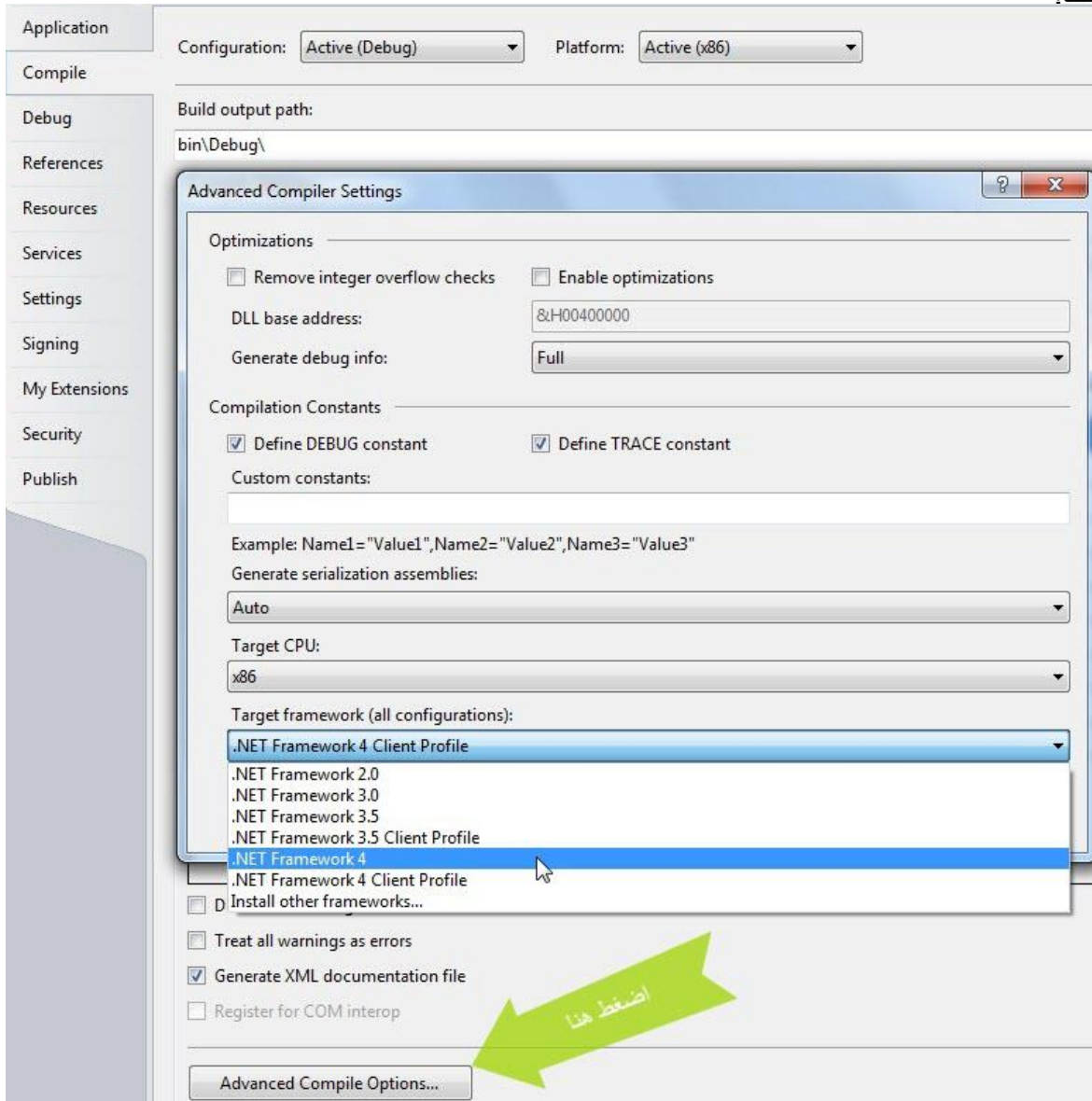
• تفسير قاعدة البيانات

قبل أن نشرع في عمل التقرير، هنالك إعدادات للمشروع يجب أن نعدلها.

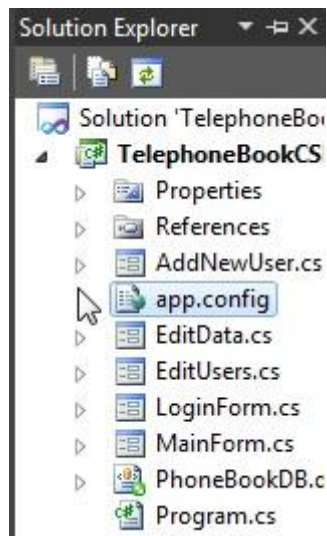
نذهب لخصائص المشروع، ثم نتبع الصورة:

C#





ثم نذهب لملف app.config ، من Solution Explorer ، سوف نضيف هذا الكود عليه :



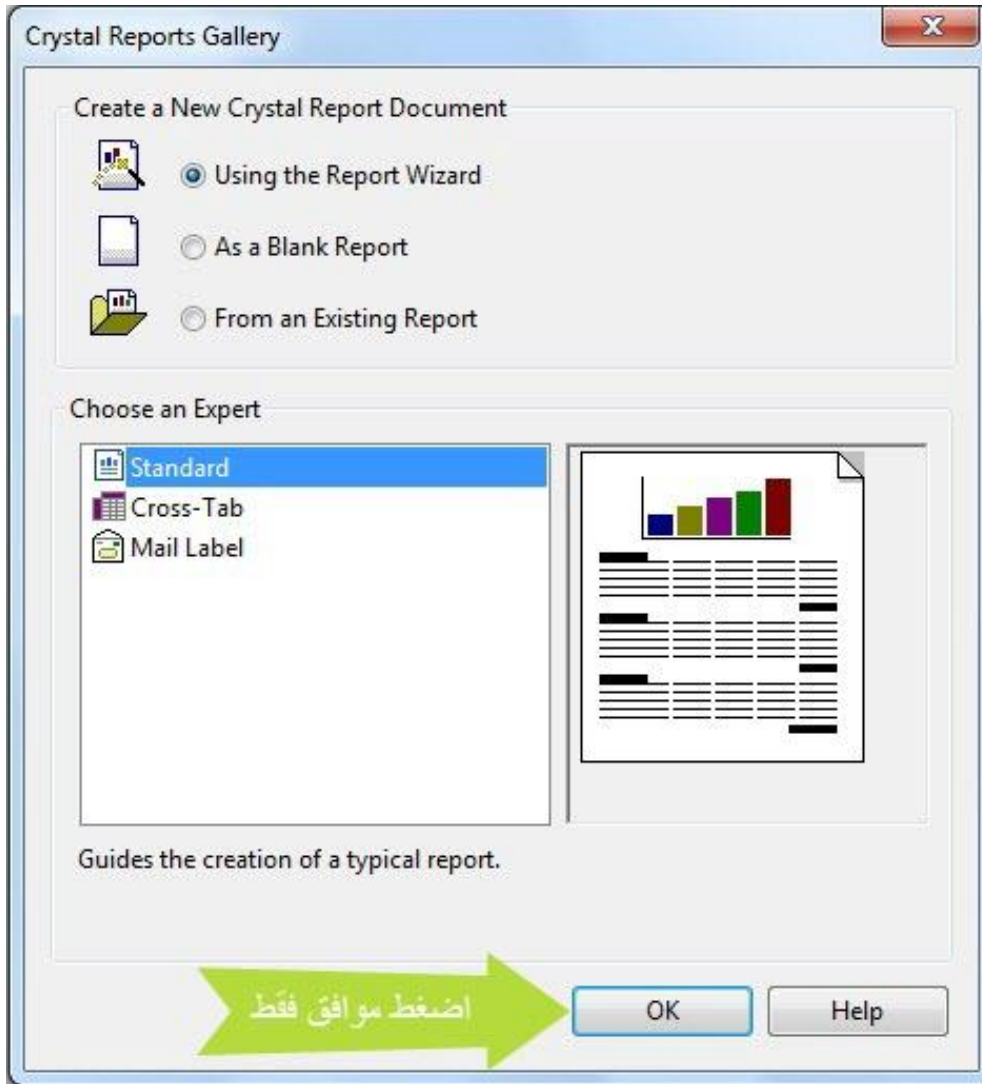
*إذا وجدنا الكود نقوم بالتعديل عليه فقط.

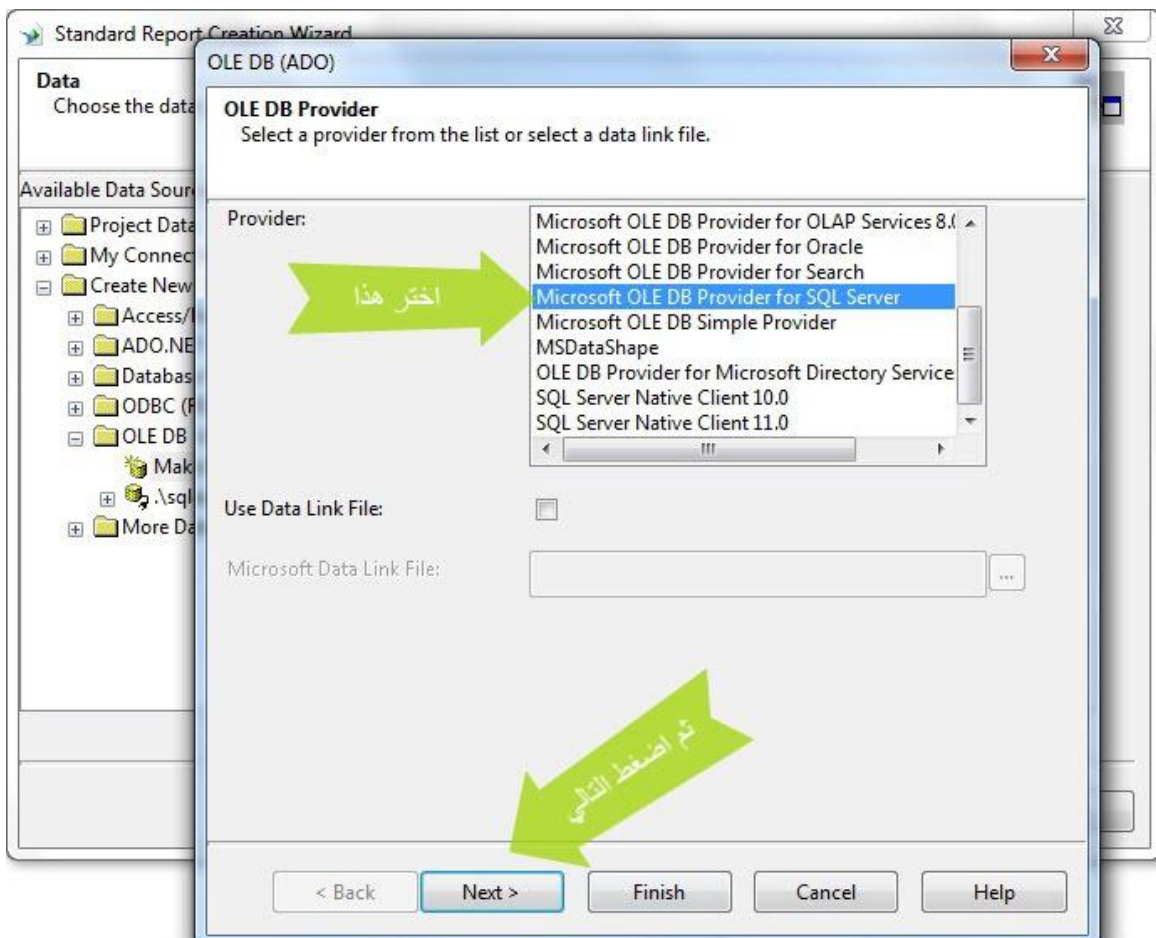
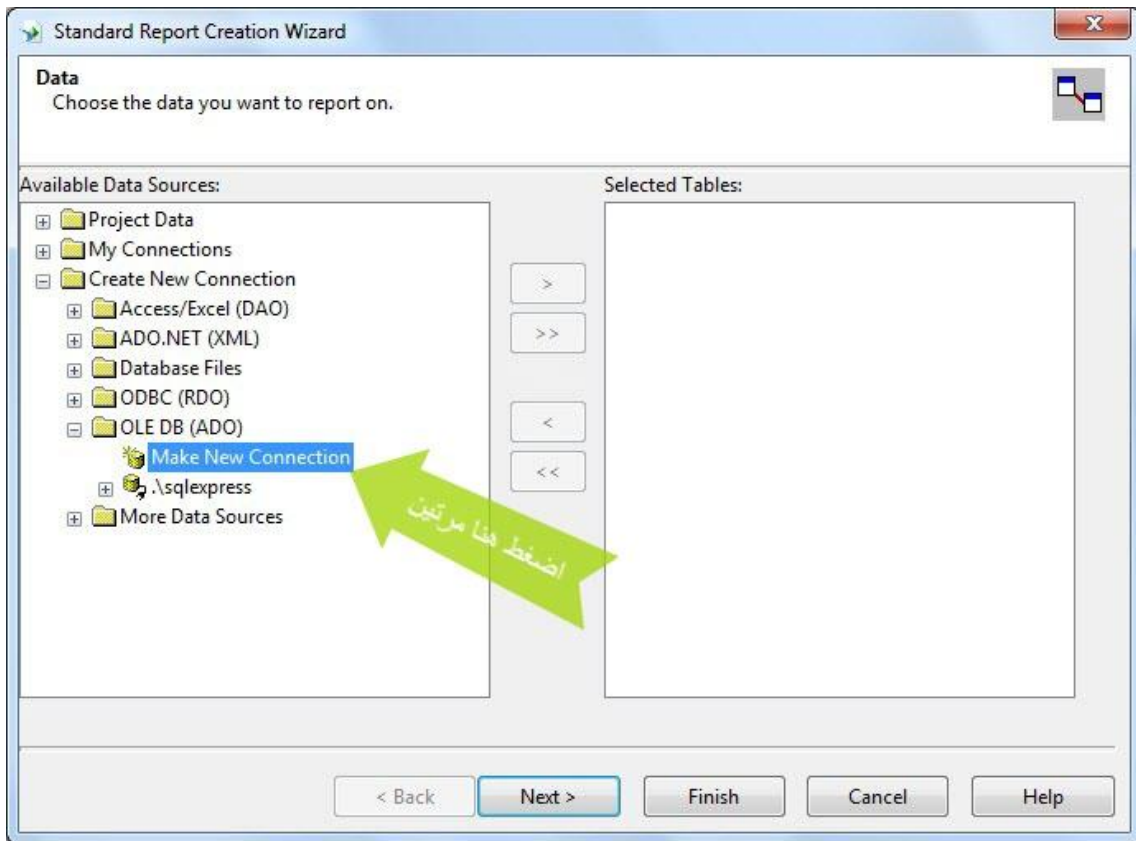
```
<startup useLegacyV2RuntimeActivationPolicy="true">  
  <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.0"/>  
</startup>
```

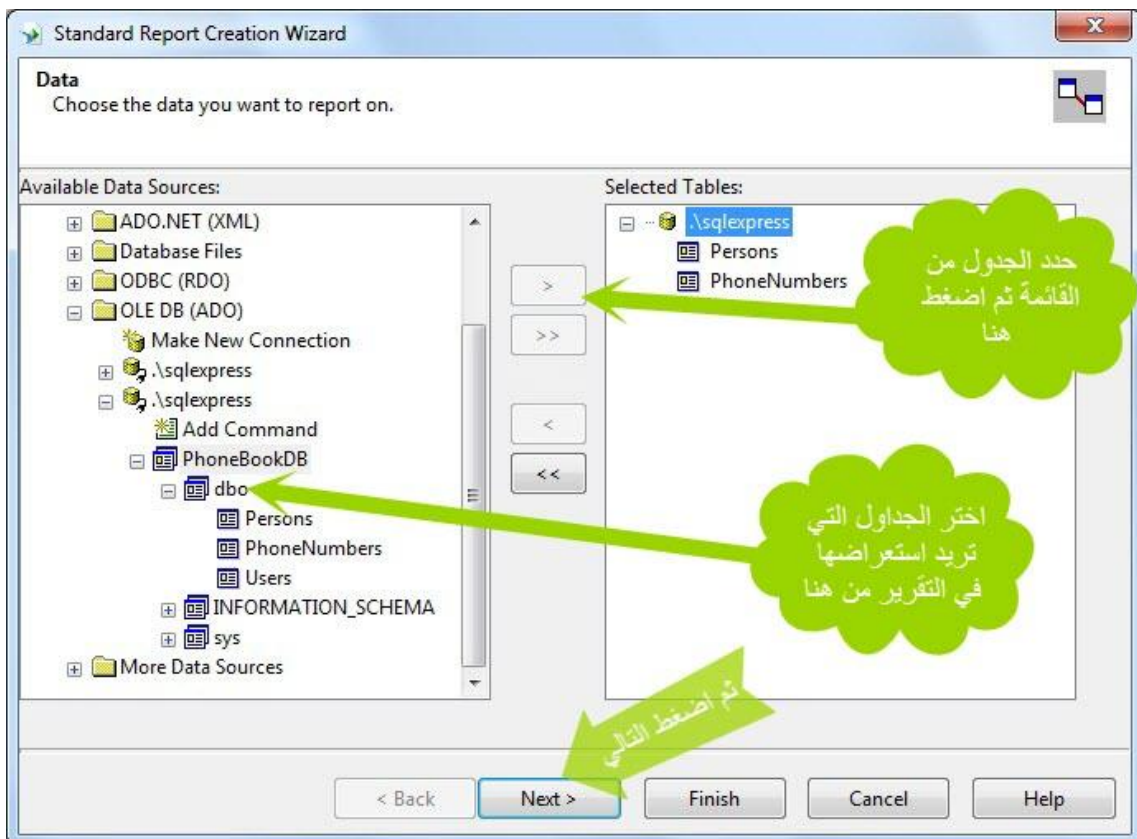
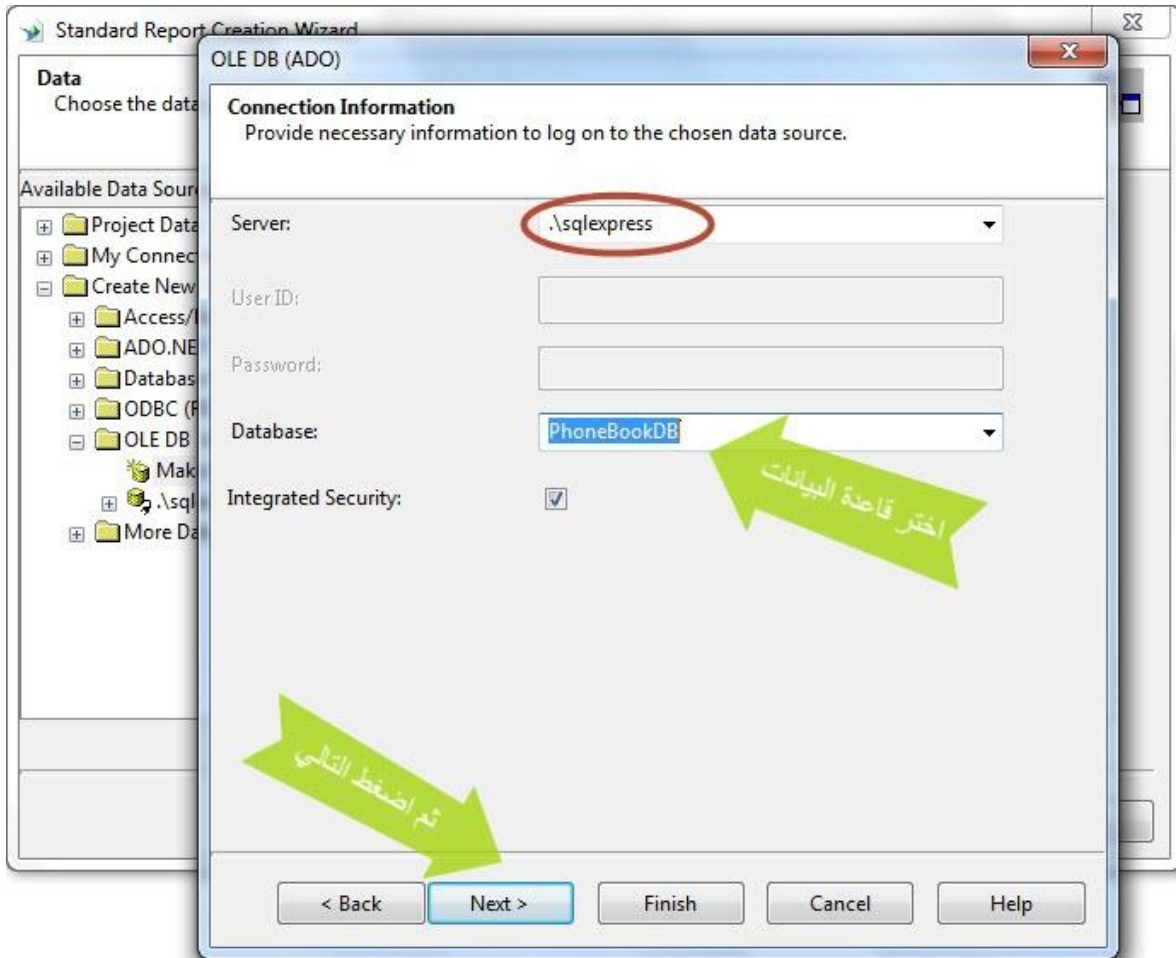
الآن نبدأ بإضافة التقرير للمشروع، من Solution Explorer نضغط على المشروع، ثم نختار:

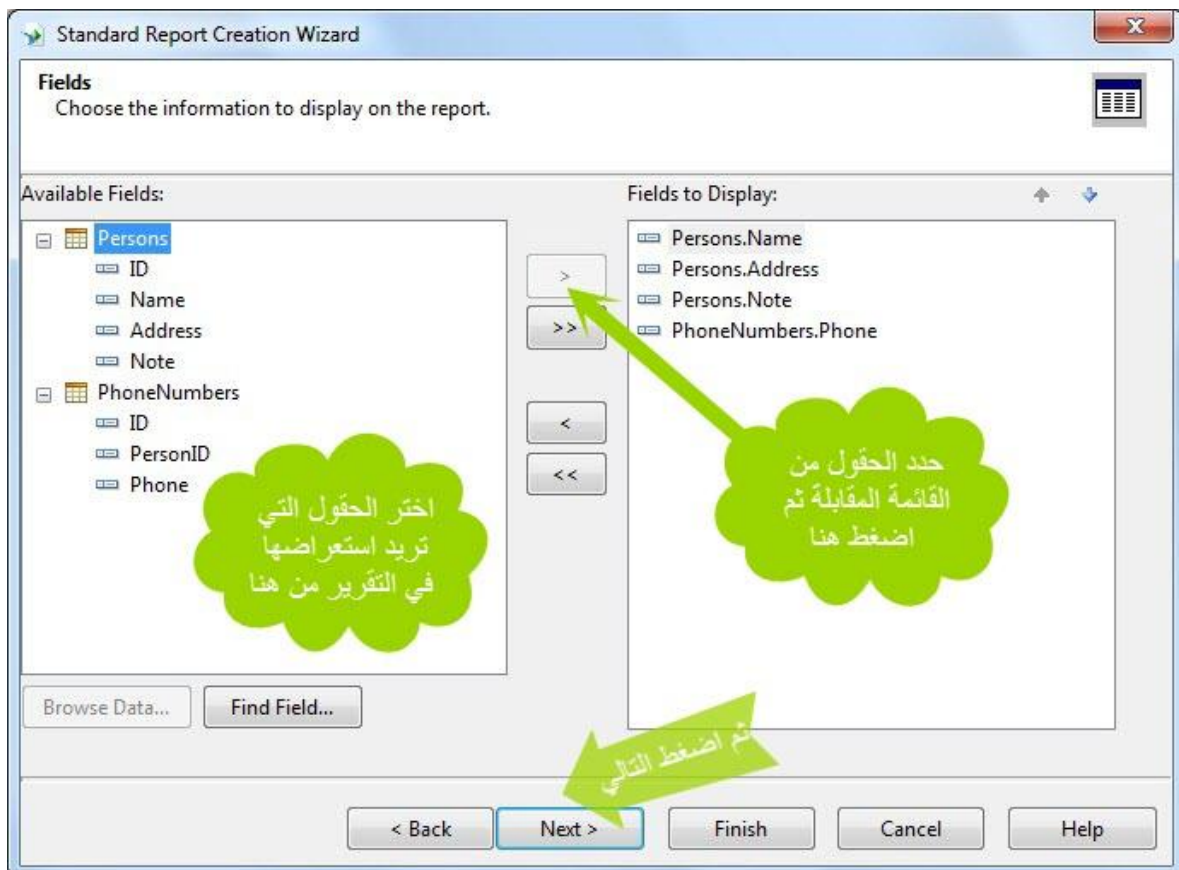
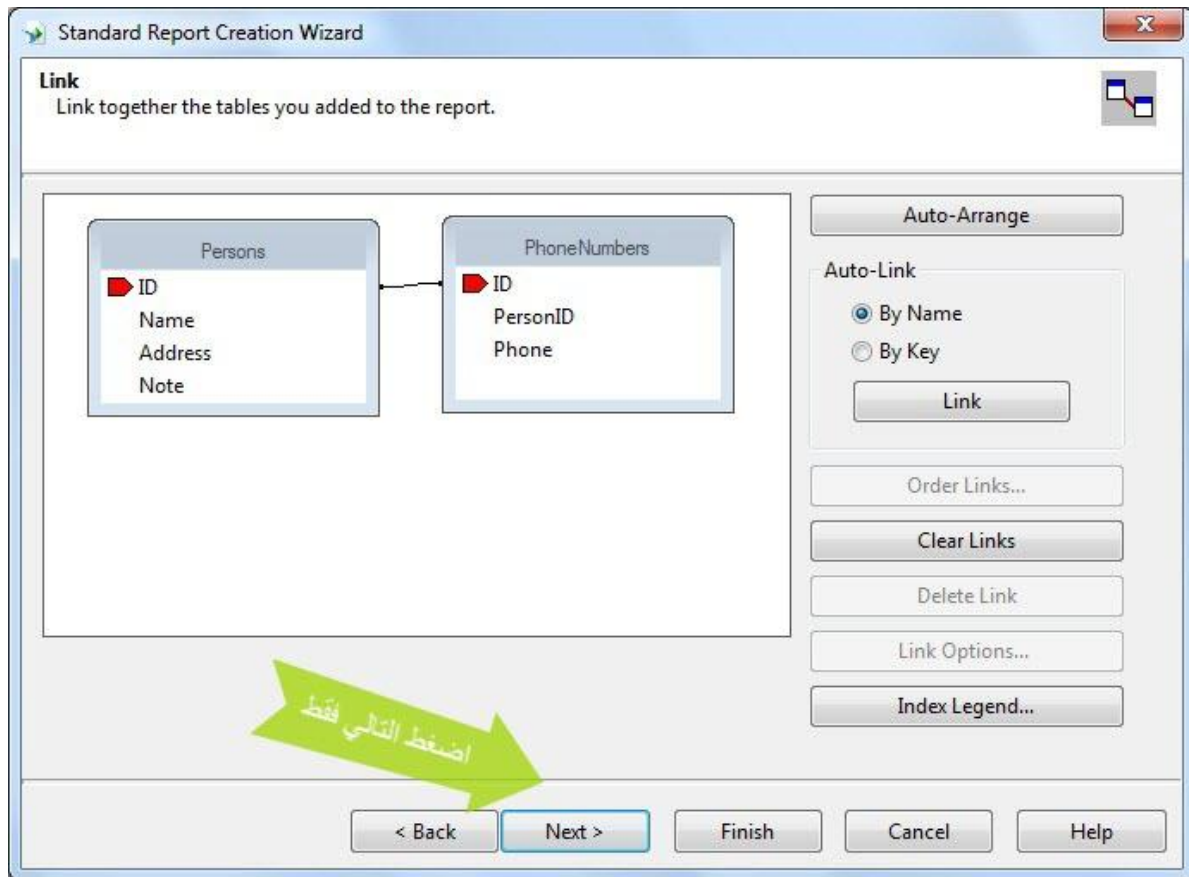
- Add
- New Item

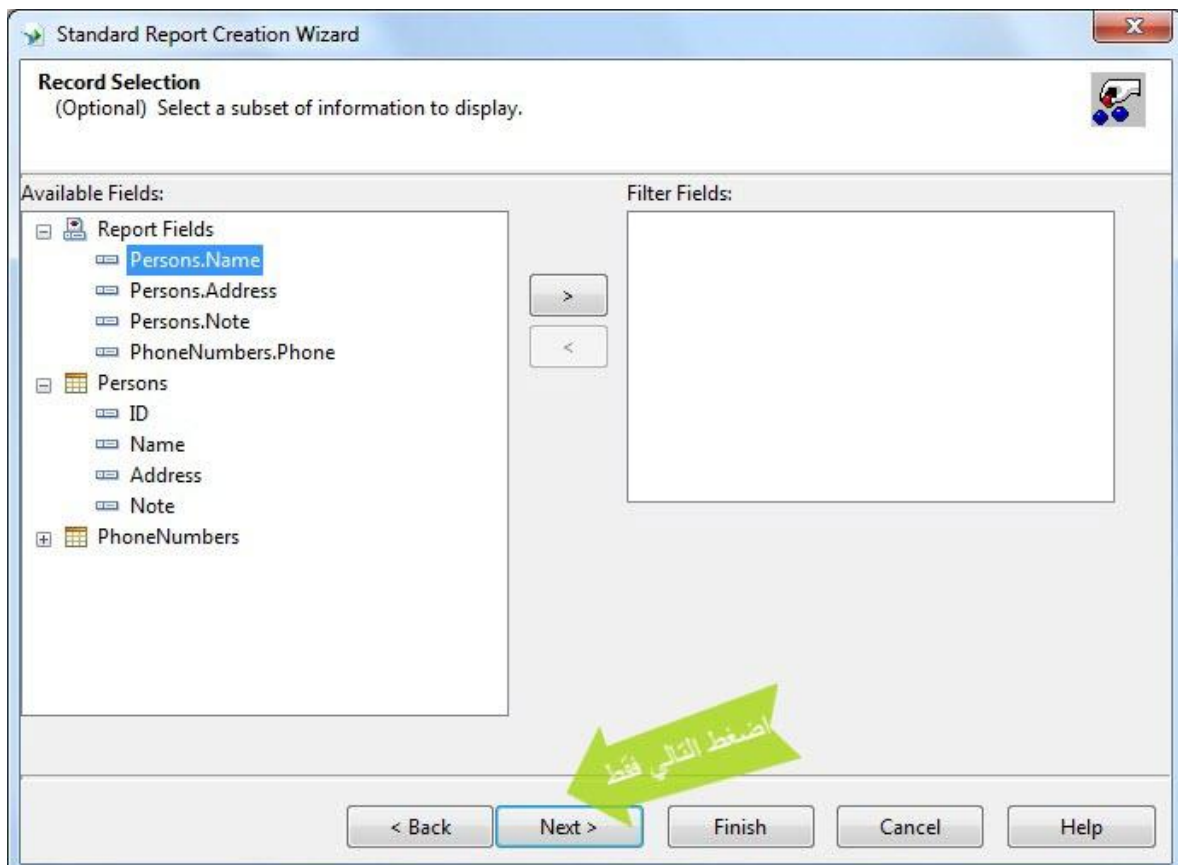
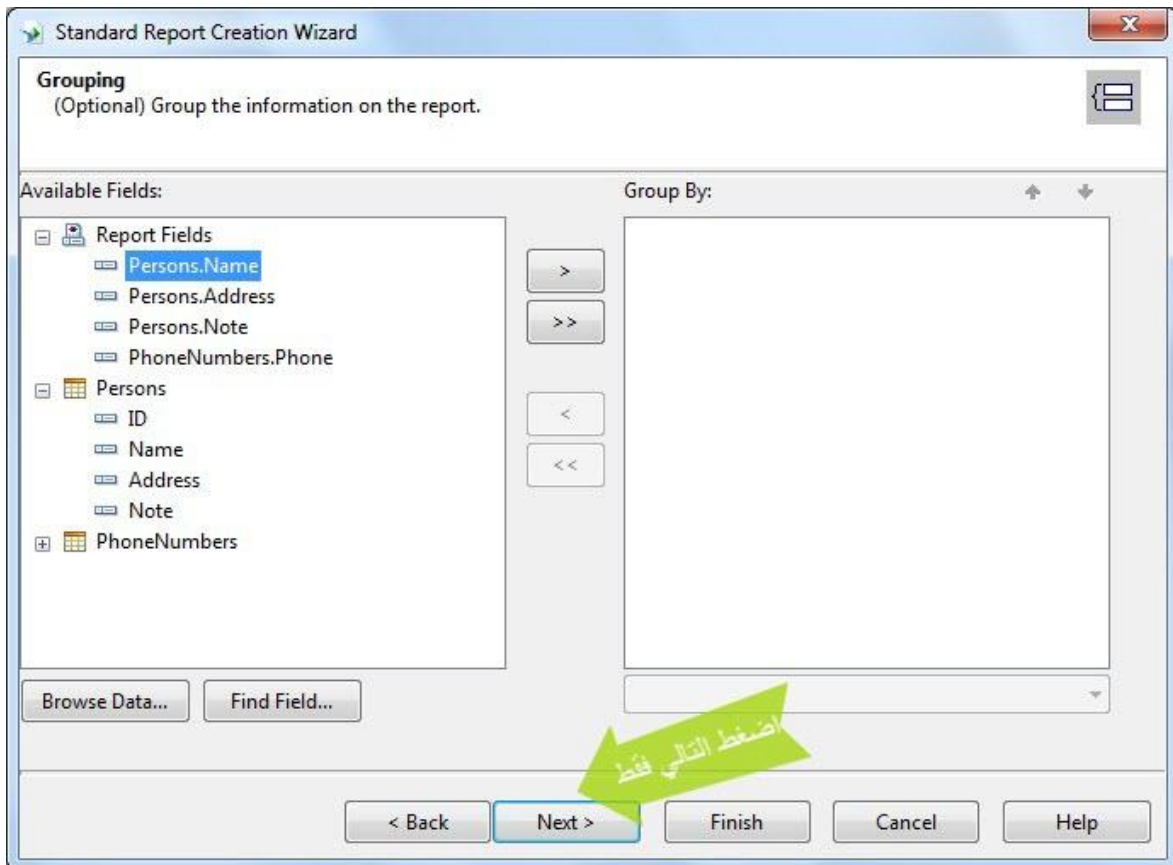
نضيف Crystal Report، ليكن اسمه Report، ثم نتبع الصور التالية:

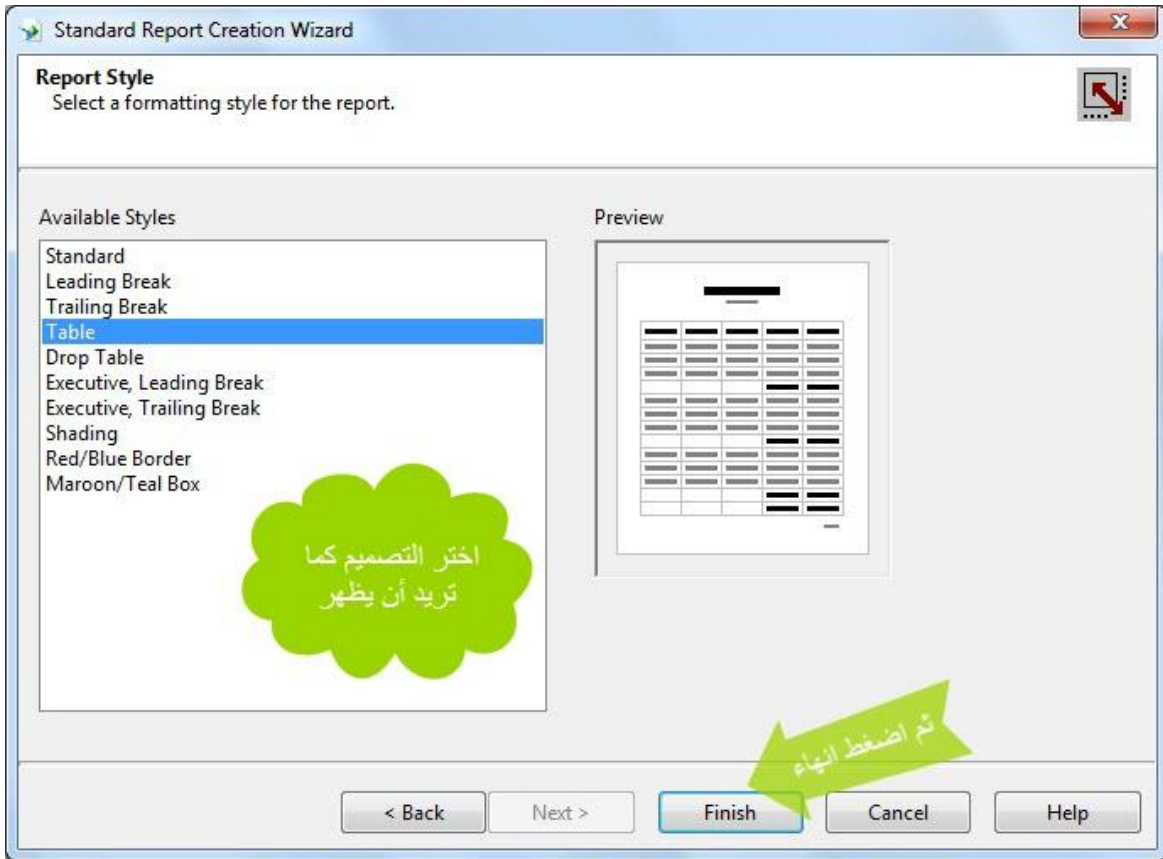










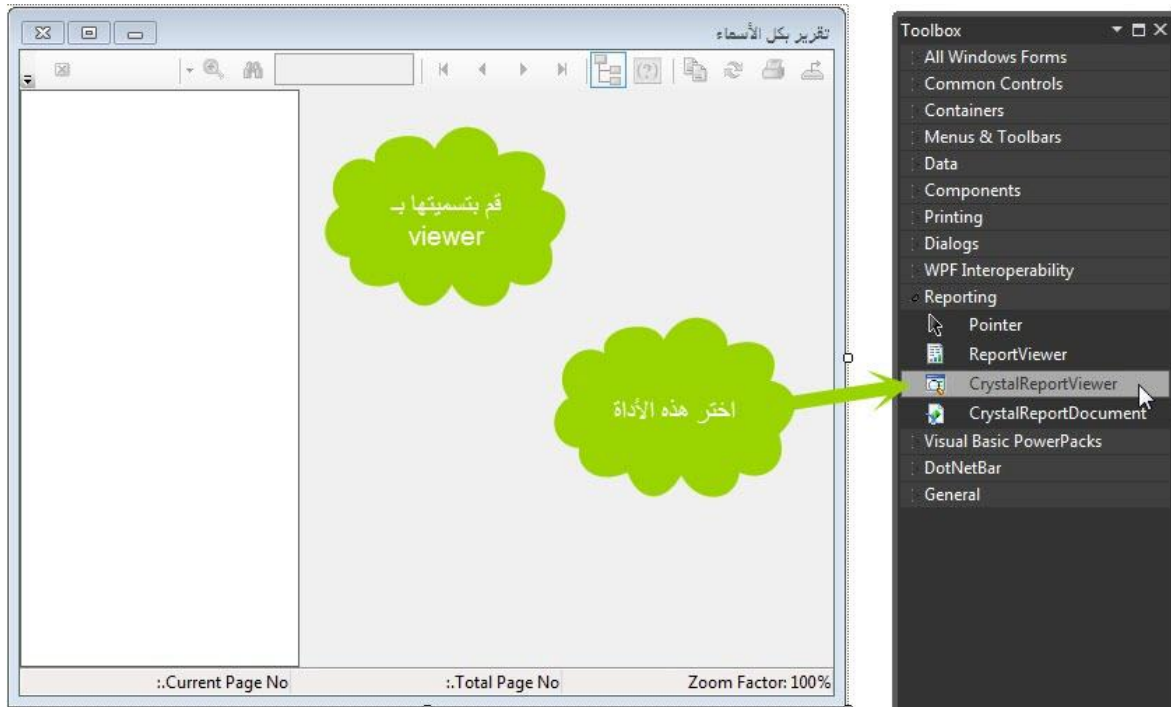


هكذا يظهر التقرير، بعد القليل من التنسيقات:

Section1 (Report Header)			
			
Section2 (Page Header)			
[Print Date]			
ملاحظات	رقم الهاتف	العنوان	الاسم
Section3 (Details)			
Note	Phone	Address	Name
Section4 (Report Footer)			
Section5 (Page Footer)			

Page Number

نضيف نموذج جديد للمشروع، سيكون اسمه: All



في حدث التحميل للنموذج، نكتب:

C#

كود

```
try
{
    DB.PhoneBookDB db = new DB.PhoneBookDB();
    var data = from d in db.Persons select d;

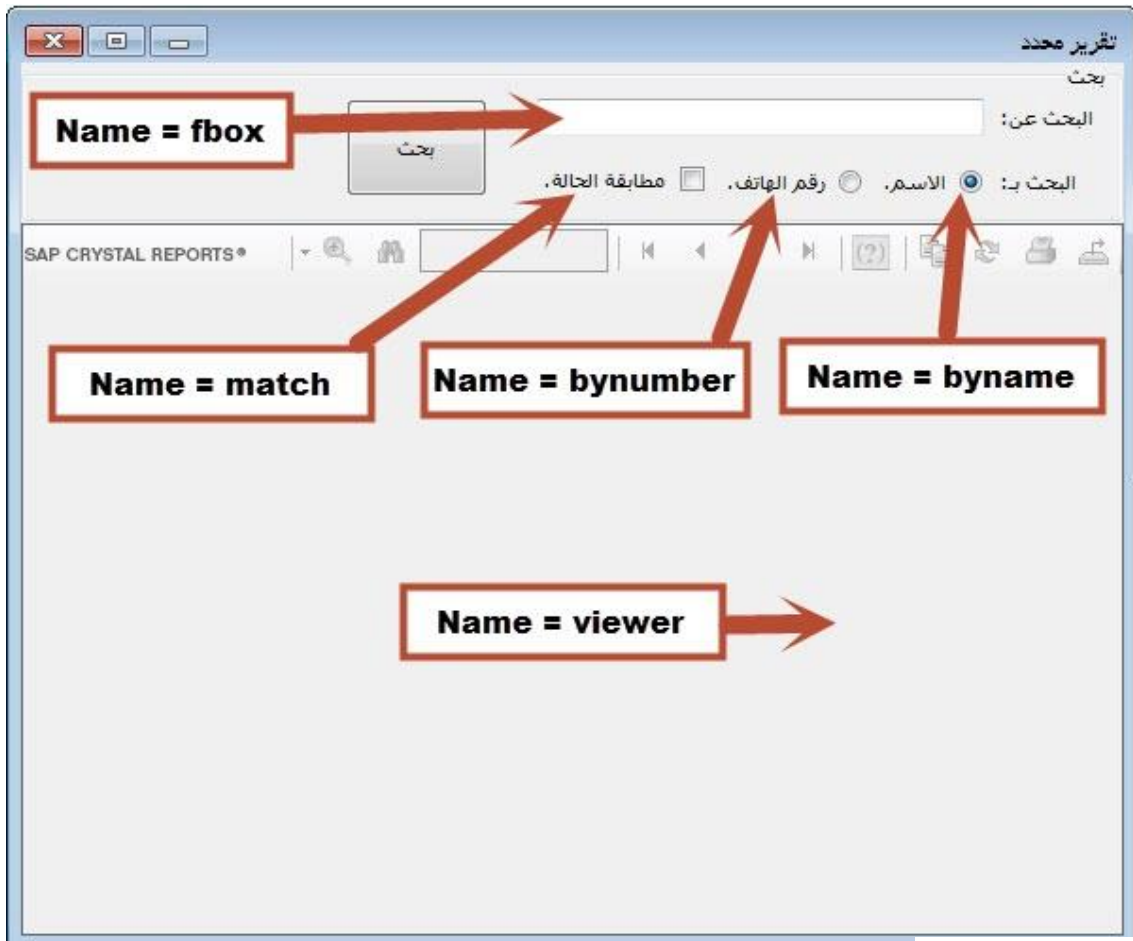
    Report report = new Report();
    report.SetDataSource(data);

    viewer.ReportSource = report;
}
catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
    MessageBoxIcon.Error); }
```

VB

كود

```
Try
    Dim db As New DB.PhoneBookDB()
    Dim data = From d In db.Persons Select d
    Dim report As New Report()
    report.SetDataSource(data)
    viewer.ReportSource = report
Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try
```



في حدث ضغط زر البحث:

C#

كود

```
DB.PhoneBookDB db = new DB.PhoneBookDB();
Report report = new Report();

try
{
    if (byname.Checked)
    {
        var data = from d in db.Persons
                    where match.Checked ? d.Name == fbox.Text.Trim() :
                    d.Name.Contains(fbox.Text.Trim())
                    select d;

        report.SetDataSource(data);
    }
    else
    {
        var data = from d in db.Persons
                    where d.PhoneNumbers.Any(p => match.Checked ? p.Phone ==
                    fbox.Text.Trim() : p.Phone.Contains(fbox.Text.Trim()))
                    select d;

        report.SetDataSource(data);
    }

    viewer.ReportSource = report;
}
catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
    MessageBoxIcon.Error); }
```

```

Dim db As New DB.PhoneBookDB()
Dim report As New Report()

Try
    If byname.Checked Then
        Dim data = From d In db.Persons
                    Where IIf(match.Checked, d.Name = fbox.Text.Trim(),
d.Name.Contains(fbox.Text.Trim()))
                    Select d

        report.SetDataSource(data)

    Else
        Dim data = From d In db.Persons
                    Where d.PhoneNumbers.Any(Function(p) IIf(match.Checked, p.Phone
= fbox.Text.Trim(), p.Phone.Contains(fbox.Text.Trim())))
                    Select d

        report.SetDataSource(data)
    End If

    viewer.ReportSource = report

Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try

```

نعود للنموذج الرئيسي، نختار من قائمة تقرير (كل الأسماء) ثم نكتب:

C#

كود

```

All all = new All();
all.Show(this);

```

VB

كود

```

All.Show(Me)

```

ثم نختار (محدد)، لنكتب:

C#

كود

```

Custom custom = new Custom();
custom.Show(this);

```

VB

كود

```

Custom.Show(Me)

```

C#

كود

```

SaveFileDialog sfd = new SaveFileDialog();
sfd.Title = "نسخة احتياطية";
sfd.FileName = string.Empty;
sfd.Filter = "نسخة احتياطية|.bak";

if (sfd.ShowDialog() == DialogResult.OK)
    try
    {
        string dbname = string.Format("{0}{1} {2}{3}",
            System.IO.Path.GetDirectoryName(sfd.FileName),
            System.IO.Path.GetFileNameWithoutExtension(sfd.FileName),
            DateTime.Today.ToShortDateString().Replace('/', '-'),
            System.IO.Path.GetExtension(sfd.FileName));

        db.ExecuteNonQuery("backup database PhoneBookDB to disk ='" + dbname + "'");

        MessageBox.Show("تم إنشاء النسخة الإحتياطية", Text, MessageBoxButtons.OK,
            MessageBoxIcon.Information);
    }
    catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
        MessageBoxIcon.Error); }

sfd.Dispose();
sfd = null;

```

VB

كود

```

Dim sfd As New SaveFileDialog()
sfd.Title = "نسخة احتياطية"
sfd.FileName = String.Empty
sfd.Filter = "نسخة احتياطية|.bak"

If sfd.ShowDialog() = DialogResult.OK Then
    Try
        Dim dbname As String = String.Format("{0}{1} {2}{3}", _
            System.IO.Path.GetDirectoryName(sfd.FileName), _
            System.IO.Path.GetFileNameWithoutExtension(sfd.FileName), _
            DateTime.Today.ToShortDateString().Replace("/"c, "-"c), _
            System.IO.Path.GetExtension(sfd.FileName))

        db.ExecuteNonQuery("backup database PhoneBookDB to disk ='" & dbname & "'")

        MessageBox.Show("تم إنشاء النسخة الإحتياطية", Text, MessageBoxButtons.OK,
            MessageBoxIcon.Information)

        Catch ex As Exception
            MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
                MessageBoxIcon.Error)
        End Try
    End If

sfd.Dispose()
sfd = Nothing

```

C#

كود

```

if (MessageBox.Show("هل تريد الاستمرار\nسوف يتم استبدال جميع البيانات", Text, MessageBoxButtons.YesNo,
MessageBoxIcon.Question) == DialogResult.No) return;

OpenFileDialog ofd = new OpenFileDialog();
ofd.Title = "استرجاع";
ofd.FileName = string.Empty;
ofd.Filter = "نسخة احتياطية*.bak";

if (ofd.ShowDialog() == DialogResult.OK)
    try
    {
        db.ExecuteNonQuery("restore database PhoneBookDB from disk ='" +
ofd.FileName + "'");

        MessageBox.Show("تم استرجاع النسخة", Text, MessageBoxButtons.OK,
MessageBoxIcon.Information);
    }
    catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error); }

ofd.Dispose();
ofd = null;

```

VB

كود

```

If MessageBox.Show("سوف يتم استبدال جميع البيانات" & vbCrLf & "هل تريد الاستمرار", Text,
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = DialogResult.No Then Exit Sub

Dim ofd As New OpenFileDialog()
ofd.Title = "استرجاع"
ofd.FileName = String.Empty
ofd.Filter = "نسخة احتياطية*.bak"

If ofd.ShowDialog() = DialogResult.OK Then
    Try
        db.ExecuteNonQuery("restore database PhoneBookDB from disk ='" &
ofd.FileName & "'")

        MessageBox.Show("تم استرجاع النسخة", Text, MessageBoxButtons.OK,
MessageBoxIcon.Information)

    Catch ex As Exception
        MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error)
    End Try
End If

ofd.Dispose()
ofd = Nothing

```

C#

كود

```
if (MessageBox.Show("هل تريد الاستمرار\nسوف تفقد جميع البيانات", Text, MessageBoxButtons.YesNo,
MessageBoxIcon.Question) == DialogResult.No) return;
try { db.DeleteDatabase(); db.CreateDatabase(); GetData(); }
catch (Exception ex) { MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK,
MessageBoxIcon.Error); }
```

VB

كود

```
If MessageBox.Show("هل تريد الاستمرار" & vbCrLf & "سوف تفقد جميع البيانات", Text,
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = DialogResult.No Then Exit Sub
Try
    db.DeleteDatabase()
    db.CreateDatabase()
    GetData()
Catch ex As Exception
    MessageBox.Show(ex.Message, "خطأ", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try
```

هذا والله تعالى أعلم وأجل

وصلى الله على سيدنا محمد وعلى آله وصحبه أجمعين