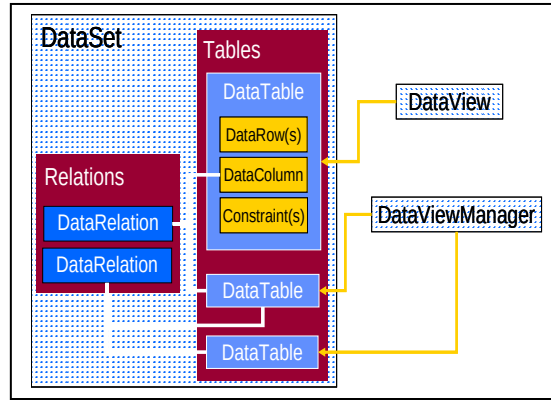


العمل مع مجموعة البيانات DATASET

في الفصل الذي تكلمنا فيه عن البيئة غير المتصلة قمنا بتعريفها كما يلي : مجموعة من البيانات يتم نسخها و تعديلها من مصدر البيانات ومن ثم إرجاع هذه التعديلات إلى مصدر البيانات. لذلك تعتبر البيئة غير المتصلة هي البيئة المثالية عندما يكون أداء النظام هو المحك.

تزودنا ADO.NET بمجموعة من الكائنات للعمل في البيئة غير المتصلة. يوجد كئنان أساسيان للعمل في البيئة غير المتصلة و هما :

١ - مجموعة البيانات Dataset : هي عبارة عن قاعدة بيانات صغيرة موجودة في الذاكرة يمكنك من إنشاء جداول و ربط هذه الجداول بعلاقات كما يتوفر فيها مجموعة من الوظائف المساعدة على تسهيل التعامل مع هذه البيانات. ومع أن مجموعة البيانات تعتبر خيارا ممتازا في كثير من الحالات إلا أنها قد لا تكون حلا مثاليا في حالات أخرى مثل أن تريد قراءة بيانات من قاعدة البيانات فقط.



حسب ما يظهر من الشكل يمكن تقسيم مجموعة البيانات إلى :

- جدول بيانات DataTable : عبارة عن تمثيل لجدول إما أن تقوم ببناء هيكله أو تأخذه من مصدر بيانات. يمكن أن تحتوي مجموعة البيانات على عدد غير محدود من الجداول.
- أعمدة DataColumn : عبارة عن تمثيل لعמוד داخل جدول هذا العمود له نوع و اسم و طول و غير ذلك من مواصفات الأعمدة كما يوجد في قواعد البيانات.

- c. صفوف DataRow : وهي صفوف البيانات.
- d. القيود DataConstarint : قيود توضع على الأعمدة مثل Unique Constraint قيد عدم التكرار أو قيد عدم قابلية قيمة Null.
- e. العلاقات DataRelation : بعد إنشاء الجداول تستطيع من خلال هذا الكائن إنشاء علاقات بين هذه الجداول.
- f. المستعرضات DataView : تستطيع إنشاء مستعرض على جدول أو أكثر من خلال هذا الكائن.

ملاحظات عامة هامة حول مجموعة البيانات

- يمكن أن تحتوي مجموعة البيانات عدد غير محدود من الجداول.
- أقرب ما يمكن تشبيهاً بـ RecordSet
- تتصل بقواعد البيانات من خلال DataAdaptor
- تستطيع إضافة الجداول إلى مجموعة البيانات الواحدة من مصادر متنوعة MySQL, Oracle.. ثم ربطها مع بعض
- مجموعة البيانات ليس لها مزود بيانات محدد بل هي عامة لجميع مزودي البيانات
- مجموعة البيانات متسلسلة Serializable : تحول إلى اكس ام ال و العكس

٢- مهية البيانات Data Adaptor : يقوم هذا الكائن باستخدام ثلاثة كائنات باطنيا وهي قارئ البيانات و كائن الأمر و كائن الاتصال. يقوم بفتح الاتصال بمصدر البيانات عن طريق كائن الاتصال ثم استخدام كائن الأمر لتنفيذ عملية ما ثم قارئ البيانات لقراءة البيانات و تعبئتها في مجموعة البيانات

دورة الحياة في مجموعة البيانات

تصور دورة الحياة في مجموعة البيانات يساعد على فهمها فهما دقيقا ومن ثم يسهل التعامل معها. يمكن تقسيم مراحل التعامل مع مجموعة البيانات إلى الخطوات التالية :

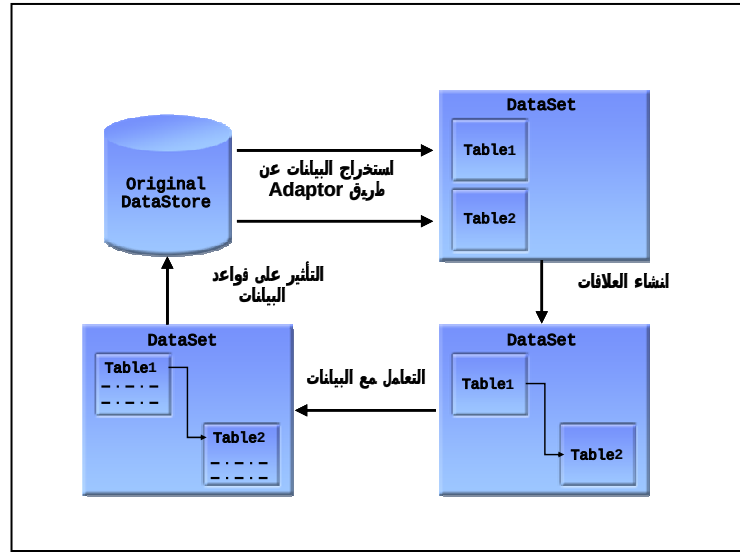
١ - تعبئة مجموعة البيانات بواسطة مهية البيانات.

٢ - ربط الجداول بعلاقات إن كان هناك احتياج.

٣ - التعامل مع البيانات.

٤ - التأثير على قواعد

البيانات بواسطة مهية البيانات.



يوجد هناك طريقتان للتعامل مع مجموعة البيانات :

١ - إنشاء هيكل البيانات برمجيا : أن تقوم بإنشاء مجموعة البيانات و إنشاء الجداول و

الأعمدة و العلاقات بنفسك ومن ثم إضافتها إلى مجموعة البيانات.

٢ - بناء مجموعة البيانات من هيكل قاعدة البيانات : أن تحضر هيكل البيانات من قواعد

البيانات و تبني مجموعة البيانات بناء على هذا الهيكل.

إنشاء هيكل البيانات برمجيا

تعتبر هذه الطريقة قليلة الاستخدام لأن أغلب التطبيقات تقوم بملء مجموعة البيانات من قاعدة بيانات وهو ما يعني إحضار هيكل البيانات. سوف تساعدك هذه الطريقة على فهم قريب لمجموعة البيانات. سوف ننشئ هيكل البيانات لمجموعة البيانات في الخطوات التالية :

١ - الخطوة الأولى هي إنشاء مجموعة البيانات. لإنشاء مجموعة بيانات برمجيا قم بكتابة الأمر التالي :

```
Dim dsTelDir As New DataSet()
```

يقوم الأمر السابق بإنشاء مجموعة بيانات اسمها dsTelDir.

٢ - الخطوة الثانية إنشاء جدول ثم إضافته إلى مجموعة البيانات التي سبق إنشائها. لإنشاء جدول و إضافته إلى مجموعة البيانات قم بما يلي:

```
Dim dtPerson As DataTable = dsTelDir.Tables.Add()
```

في العبارة السابقة قمنا بإضافة جدول إلى مجموعة البيانات التي أنشأناها حديثا ثم أسندنا هذا الجدول إلى جدول أنشأناه و اسمه dtPerson. عند إضافة جدول إلى مجموعة بيانات و عدم تسميته فإنه يأخذ اسم TableX و العدد اكس هو تصاعدي حسب عدد الجداول الموجودة في مجموعة البيانات.

٣ - الخطوة الثالثة هي إضافة عامود إلى جدول داخل مجموعة البيانات. الأمر التالي يقوم بإنشاء عامود ومن ثم إضافته إلى جدول :

```
Dim colName As DataColumn = dtPerson.Columns.Add("Name", GetType(System.String))
colName.AllowDBNull = False
colName.Unique = True
```

في الخطوة السابقة قمنا بما يلي :

a. إضافة عامود -حقل- وعينا له اسم Name و نوع String.

b. وضعنا قيد منع إدخال قيم Null

c. في السطر الثالث استخدمنا قيد Unique لمنع تكرار نفس القيمة.



سوف نحاول تطبيق كل الخطوات السابقة من خلال المثال التالي :

```
Dim dsTelDir As New DataSet()
Dim dtPerson As DataTable = dsTelDir.Tables.Add()
Dim colName As DataColumn = dtPerson.Columns.Add("Name",
GetType(System.String))
```

```
colName.AllowDBNull = False
colName.Unique = True
MessageBox.Show(dtPerson.TableName)
MessageBox.Show(dsTelDir.Tables.Count.ToString())
```

في هذا المثال طبقنا كل الخطوات السابقة ثم أظهرنا رسالتان :

- ١ - الرسالة الأولى نقوم بعرض اسم الجدول.
- ٢ - الرسالة الثانية نقوم فيها بعرض عدد الجداول في مجموعة البيانات.

إنشاء قيد المفتاح الرئيسي

تحتوي الجداول في قواعد البيانات على عامود أو أكثر لا يمكن أن تتكرر قيمته في

أي صف من صفوف الجدول. من أمثلة المفتاح الرئيسي :

١ - رقم الموظف في جدول الموظفين

٢ - رمز الصنف في جدول الأصناف



المثال التالي يوضح كيفية إنشاء قيد المفتاح الرئيسي :

```
Dim myTable As DataTable = New DataTable()
Dim myColumn As DataColumn
myColumn = New DataColumn()
myColumn.DataType = System.Type.GetType("System.String")
myColumn.ColumnName = "FirstName"
myTable.Columns.Add(myColumn)
myTable.PrimaryKey = New DataColumn() {myTable.Columns("FirstName")}
```

يمكن شرح المثال السابق بالخطوات التالية :

١ - أنشأنا جدول ثم أنشأنا عامود.

٢ - حددنا نوع البيانات و اسم العامود

٣ - قمنا بإضافة العامود إلى الجدول

٤ - قمنا بتحديد المفتاح الرئيسي من خلال العبارة :

```
myTable.PrimaryKey = New DataColumn() {myTable.Columns("FirstName")}
```

إنشاء عبارات رياضية

قد تحتاج أن تعرض حقل يكون نتيجة حسابات لحقول أخرى، مثل أن يكون الحقل الأول يمثل

درجة الاختبار الأول و الحقل الثاني يمثل درجة الاختبار الثاني ثم هناك حقل يمثل مجموع

الدرجات. هذا الحقل يمثل بعמוד يأخذ قيمه من الحسابات و لا يكون له عامود مماثل في قواعد البيانات غالبا. يمكن إضافته باستخدام خاصية Expression وهي أحد خصائص الكائن DataColumn.



لننظر إلى المثال التالي الذي سوف يبرز الفكرة :

```
Dim ColUnitPrice As New DataColumn("Price", GetType(System.Decimal))
Dim ColQuantity As New DataColumn("Quantity", GetType(System.Decimal))
Dim TotalCost As New DataColumn("Total", GetType(System.Decimal))
Dim mytable As New DataTable()
TotalCost.Expression = "Price * Quantity"
mytable.Columns.Add(ColUnitPrice)
mytable.Columns.Add(ColQuantity)
mytable.Columns.Add(TotalCost)
Dim myRow As DataRow
myRow = mytable.NewRow()
myRow("Price") = 5
myRow("Quantity") = 1
mytable.Rows.Add(myRow)
Label2.Text = mytable.Rows(0).Item(2)
```

يمكن توضيح المثال السابق بالخطوات التالية :

- ١ - تعريف ثلاثة أعمد وهي ColUnitPrice و ColQuantity و TotalCost
- ٢ - تعريف جدول
- ٣ - قمنا بتعريف العמוד TotalCost بأنه عبارة رياضية و عينا له قيمة $Price * Quantity$ وهو حاصل ضرب العמודين الآخرين.
- ٤ - أضفنا الأعمدة إلى الجدول
- ٥ - قمنا بتعريف صف جديد
- ٦ - عينا لهذا الصف قيمة وهي صف جديد من الجدول الذي يحوي ثلاثة حقول و بالتالي فإن هذا الصف سوف يتكون من ثلاث قيم هي ٥ و ١ ونتيجة ضربهما.
- ٧ - أضفنا قيمة ٥ للسعر و ١ للكمية.
- ٨ - قمنا بإضافة هذا الصف للجدول
- ٩ - قمنا بعرض العنصر الثالث -مجموع التكلفة- على أداة عنوان و سوف تكون قيمته ٥ لأنه تم حسابها بناء على المعادلة التي أنشأناها.

بناء مجموعة البيانات من هيكل قاعدة البيانات

لبناء مجموعة بيانات بناء على قاعدة بيانات يوجد هناك طريقتان :

١ - استخدام واجهة تطوير الدوت نت.

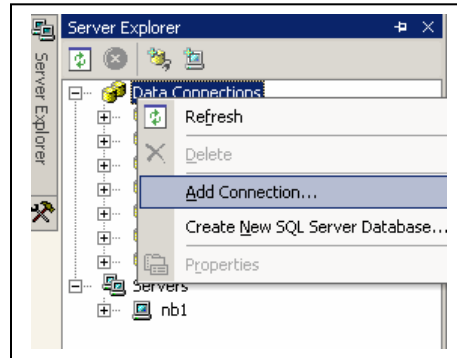
٢ - بناء ذلك برمجيا.

استخدام واجهة تطوير الدوت نت

تعتبر هذه الطريقة من الطرق السريعة للوصول إلى قواعد البيانات ومن ثم بناء واجهات تتعامل مع قواعد البيانات. استخدام مساعد إنشاء النماذج يعتبر طريقة جيدة لما يسمى بالنمذجة -Prototype-. في البداية لابد من إنشاء مجموعة البيانات ويتم كالتالي :

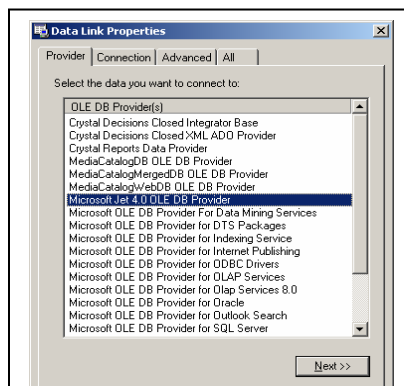
١ - قم بإنشاء الاتصال بقاعدة البيانات TelDir. من خلال نافذة مستكشف الخادم كما يظهر

في الشكل

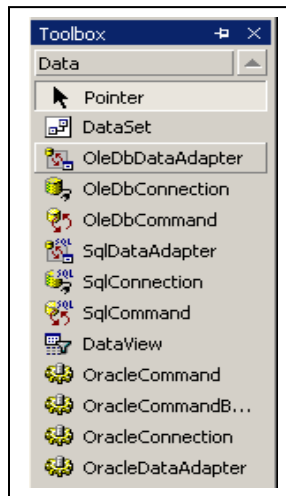


٢ - قم بنقر زر الفأرة الأيمن على Data Connection ثم اختر Add Connection. سيظهر لك الشكل اختر خانة مزود البيانات Provider ثم اختر المزود الخاص بأكسس وهو Microsoft.Jet.OLEDB.4 اختر خانة Connection ثم اختر

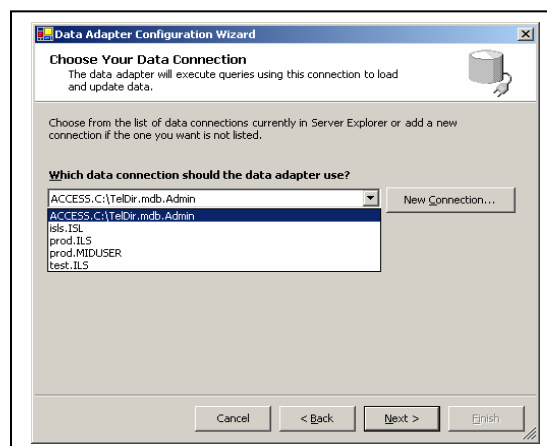
قاعدة البيانات TelDir



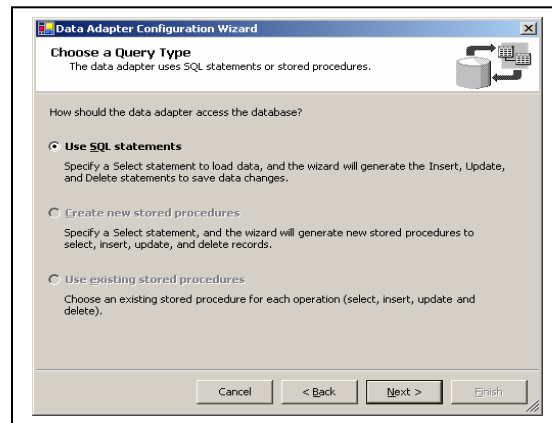
٣- أنشء مهية البيانات (والذي يوجد في خانة Data في مربع الأدوات كما يظهر من الشكل) وذلك بإضافته إلى النموذج. سوف تظهر لك رسالة ترحيب انقر التالي.



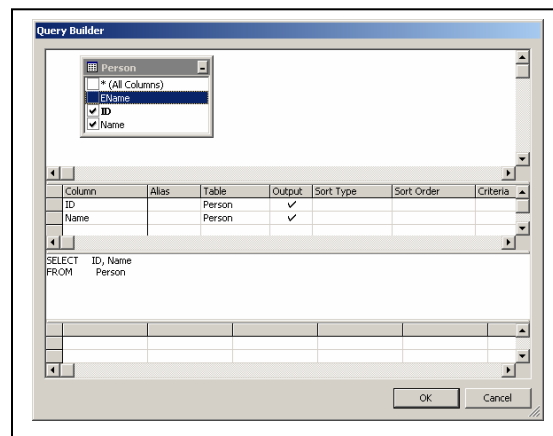
٤- سوف تظهر لك شاشة اختيار الاتصال الذي تريد استخدامه قم باختيار الاتصال الذي تم إنشائه كما يظهر من الشكل. ثم انقر على التالي.



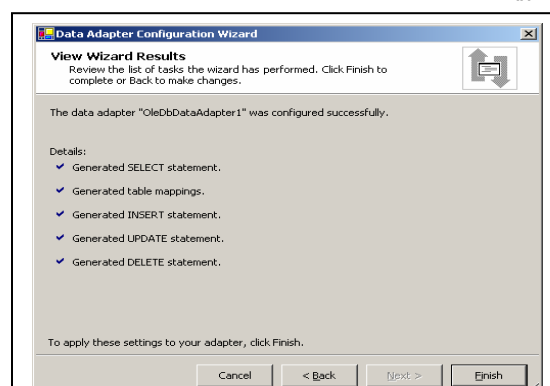
٤- سوف تظهر لك شاشة تطلب منك اختيار طريقة الوصول إلى قواعد البيانات. في حالة مايكروسوفت أكسس لا يوجد إلا الاختيار الأول كما يظهر من الشكل، انقر التالي،



- ٥ - سوف تظهر لك شاشة تسألك عن جملة SQL التي تريد تنفيذها. انقر على زر QueryBuilder لإنشاء الجملة بطريقة سهلة. سوف تظهر لك قائمة بالجدول الموجودة في قاعدة البيانات. قم باختيار الجدول Person ثم انقر موافق. سوف تظهر لك حقول هذا الجدول كما ترى في الشكل اختر ID, Name ثم انقر موافق. سوف ترجع إلى الشاشة الأولى انقر التالي.

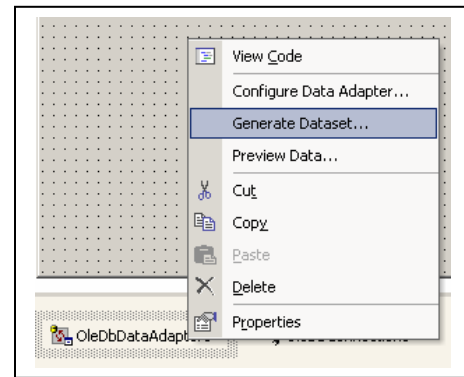


- ٦ - سوف يظهر لك الشكل و الذي يخبرك بأنه تم إنتاج جميع أنواع جمل الاستفسار البنائية للجدول الذي اخترت انقر على زر إنهاء. لاحظ أنه تم إنشاء كائني اتصال و مهية بيانات.



٧- لإنشاء مجموعة البيانات قم بالنقر بالزر الأيمن على مهية البيانات ليظهر لك الشكل. ثم

اختر Generate Data Set

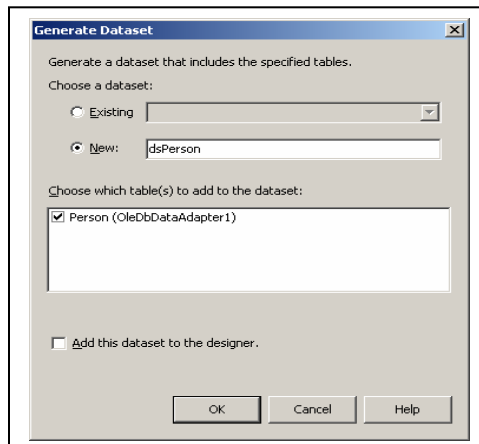


٨- سوف يظهر لك الشكل و الذي يطلب منك أن تزوده باسم مجموعة البيانات. قم تسميتها

dsPerson كما يظهر لك من الشكل. قم بإزالة علامة الاختيار عن Add This

Dataset to The Designer انقر على زر موافق سوف تلاحظ أنه تم إضافة

ملف dsPerson.xsd إلى مستكشف الحلول.



٩- لتحميل البيانات من قاعدة البيانات إلى مجموعة البيانات أضف زر أمر على النموذج و

أضف شبكة بيانات Datagrid من مربع الأدوات ثم أضف الكود التالي داخل حدث

النقر لزر أمر

```
Dim ds As New DataSet()
Me.OleDbConnection1.Open()
Try
    Me.OleDbDataAdapter1.Fill(ds)
    DataGrid1.DataSource = ds.Tables(0)
Catch Ex As OleDb.OleDbException
    MessageBox.Show(Ex.Message)
End Try
Me.OleDbConnection1.Close()
```

في الكود السابق قم بما يلي :

- ١ - عرفنا مجموعة بيانات
- ٢ - قمنا بفتح الاتصال الذي سبق وأن أنشأناه
- ٣ - قمنا بملء مجموعة البيانات بواسطة مهية البيانات الذي سبق إنشائه.
- ٤ - أرجعنا الجدول رقم صفر -الجدول الأول- إلى Datagrid ليتم عرضه



بناء مجموعة البيانات برمجيا

تعتبر هذه الطريقة الأكثر شيوعاً عند استخدام مجموعة البيانات. يمكن ملئ مجموعة البيانات دون أن تحتاج إلى بناء هيكل البيانات لأن هيكل البيانات يتم تكوينه بعد استرجاع البيانات من قاعدة البيانات. قم بإضافة الأدوات التالية و أعطها الخصائص التالية :

اسم الأداة	الخاصية	القيمة
Button1	Name	cmdFillGrid
	Text	إحضار معلومات الأشخاص
Label1	Name	Label1
	Text	
dataGrid	Name	DataGrid1
Form1	Text	مجموعة البيانات برمجيا
	RightToLeft	Yes

قم بإضافة الكود التالي في حدث النقر الخاص بزر الأمر :

```
Dim con As New OleDb.OleDbConnection()
Dim dsperson As New DataSet()
Dim sql As String = "select id,name from person"
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=C:\TelDir.mdb;Persist Security Info=False"
Try
    Dim da As New OleDb.OleDbDataAdapter(sql, con)
    da.Fill(dsperson, "Person")
    DataGrid1.DataSource = dsperson.Tables("Person")
Catch e1 As OleDb.OleDbException
    Label1.Text = e1.Message
End Try
```

من الكود السابق يمكن توضيح النقاط التالية :

- ١- عرفنا اتصال بقاعدة البيانات ثم عرفنا مجموعة بيانات ثم عرفنا جملة sql التي نريد أن نسترجعها، ثم عرفنا سلسلة الاتصال.
- ٢- قمنا بتعريف مهية البيانات و أعطيناها عاملان هما : الاتصال الذي سيستخدمه لاسترجاع البيانات و جملة sql التي نريد أن ينفذها.
- ٣- ملئ مجموعة البيانات بنتيجة جملة الاستفسار sql.
- ٤- قمنا بإرجاع جدول الأشخاص إلى Datagrid ليتم عرضه كما يظهر لك من الشكل.

لاحظ أننا لم نقوم بفتح الاتصال بقاعدة البيانات لأن مهية البيانات يقوم بذلك. 

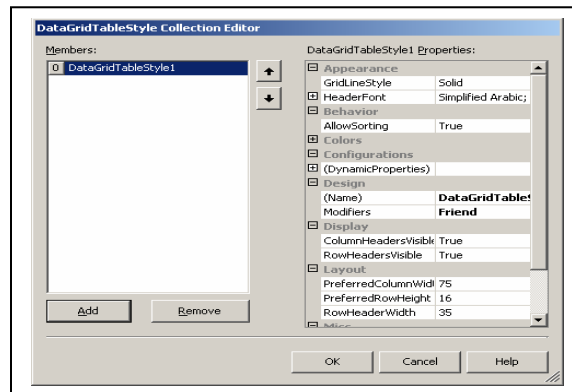


تطوير واجهة النظام

تعتبر شبكة البيانات من أهم أدوات عرض البيانات و التعامل معها. لذلك سوف نقوم الآن باستكشاف كيف يمكن أن نطور شكل عرضها و كيفية تعامل مجموعة البيانات معها. لو نظرنا إلى المثال السابق لوجدنا أنه قد ظهر لنا عنوان عمود رقم الموظف كما جاء من قاعدة البيانات ماذا لو أردت أن تغير هذا الاسم ليكون باللغة العربية فيظهر رقم الموظف بدلا من id و يظهر اسم الموظف بدلا من Name.

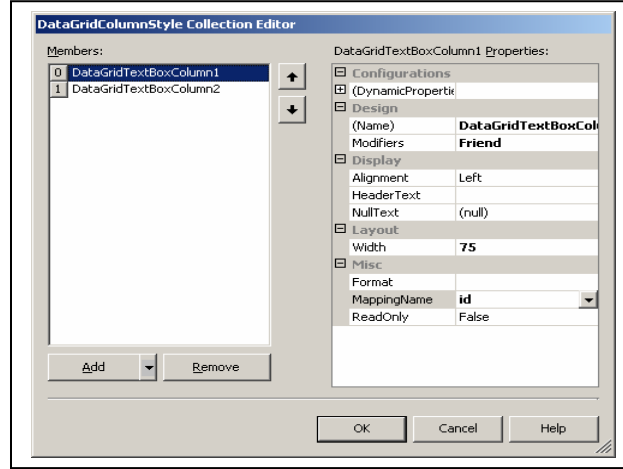
تستطيع أن تنشئ تصميم لجدول في شبكة البيانات ومن ثم تجعل هذه الجداول هي واجهات تلك البيانات التي في مجموعة البيانات. قم بالخطوات التالية لكي تنشأ مظهر لجدول Person الموجود في مجموعة البيانات da :

١ - اذهب إلى خصائص شبكة البيانات لتجد الخاصية TableStylas. واضغط زر الانتقال المجاور لها. ستظهر لك نافذة تحرير مجموعة أنماط الجداول



٢ - اضغط زر الإضافة Add. سيضاف الكائن DataGridTableStyle1 للمجموعة. سوف يكون هذا الجدول هو واجهة Person الموجود من ضمن مجموعة البيانات.

٣- اختر خاصية MappingName قم بكتابة Person وهو اسم الجدول في مجموعة البيانات و الذي سوف يعرض من خلال هذا الشكل..



٤- هذه الخطوة سوف تقوم بربط أعمدة الجدول الموجود في مجموعة البيانات هنا لنتمكن من إعطائها خصائص كما نريد. انقر على النقاط الثلاثة المقابلة لهذه الخاصية ليظهر لك الشكل. اضغط زر الإضافة Add ليتم إضافة GridColumnStyle ، اذهب إلى الخاصية MappingName و اكتب id ليتم ربط هذا الكائن بالعمود id الموجود في الجدول Person اذهب إلى خاصية HeaderText و اكتب رقم الموظف، هذا هو ما سيظهر عنواننا لعمود رقم الموظف في شبكة البيانات. أضف كائنا آخر و اذهب إلى MappingName و اكتب name. اذهب إلى خاصية HeaderText و اكتب اسم الموظف. اضغط Ok ثم Ok ثم شغل البرنامج ليظهر لك الشكل



البيانات

ماذا لو أردت أن تعرض الموظفين ثم بعد اختيار واحد منهما تقوم بعرض أرقام هاتفه كما يظهر من الشكل . لاحظ كيف تم عرض اختيار أرقام الموظف ثم عرضها بشكل رأس و تفاصيل.



قم بتغيير الكود في المثال السابق ليكون كما يلي :

```
Dim con As New OleDb.OleDbConnection()
Dim Myds As New DataSet()
Dim sql(1) As String
sql(0) = "select id,name from person"
sql(1) = "select ID,Number,person from Phone"
con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=C:\TelDir.mdb;Persist Security Info=False"
Try
    Dim daPerson As New OleDb.OleDbDataAdapter(sql(0), con)
    Dim daPhone As New OleDb.OleDbDataAdapter(sql(1), con)
    daPerson.Fill(Myds, "Person")
    daPhone.Fill(Myds, "Phone")
    Myds.Relations.Add("الموظف هاتف أرقام", " & _
    "Myds.Tables("Person").Columns(0), Myds.Tables("Phone").Columns(2), True)
    DataGrid1.DataSource = Myds
Catch e1 As OleDb.OleDbException
    Label1.Text = e1.Message
End Try
```

لتوضيح الكود التالي افهم الخطوات التالية :

- ١- تم تعريف اتصال ثم تعريف مجموعة بيانات.
- ٢- تم تعريف مصفوفة مكونة من صفين كل واحد منهما يمثل جملة استعلام، الأول من جدول الموظفين و الثاني من جدول أرقام الهاتف.
- ٣- قمنا بتعريف مهيتي بيانات الأولى لتنفيذ الاستعلام الخاص بالموظفين ثم وضع نتيجته في جدول اسمه Person في مجموعة البيانات، مهية البيانات الثانية سوف تنفذ الاستعلام الثاني و الذي يقوم باسترجاع رقم الموظف و أرقام هواتفه ثم وضعنا النتيجة في جدول اسمه Phone داخل مجموعة البيانات Myds..

- ٤ - هذه الخطوة هي الأهم حيث قمنا بربط الجدولين ببعضهما البعض من خلال العلامود الأول في جدول الموظفين و الذي يمثل رقم الموظف و العلامود الثاني في جدول Phone و الذي يمثل كذلك رقم الموظف.
- ٥ - قمنا بعرض نتيجة مجموعة البيانات كاملة على شبكة البيانات.

البحث في مجموعة البيانات

يمكنك البحث في مجموعة البيانات من خلال تطبيق جملة استفسار على جدول موجود في مجموعة البيانات. إن أفضل طريقة للفهم هي المثال :

 في المثال التالي نقوم بالبحث عن الصف الذي يحوي اسم الموظف عبدا ثم عرض رقم أول موظف في نتيجة البحث:

```
Dim SearchRow() As DataRow =
Myds.Tables("Person").Select("name='عبد الله'")
MessageBox.Show(SearchRow(0).Item(0))
```

ماذا لو أردت أن ترتب نتيجة البحث حسب حقل ما تصاعدياً أو تنازلياً ؟

 في المثال التالي نقوم بالبحث عن الصف الذي يحوي اسم الموظف عبدا ثم ترتيب نتيجة البحث حسب الرقم من الأعلى إلى الأدنى :

```
Dim SearchRow() As DataRow =
Myds.Tables("Person").Select("name='عبد الله'", "id DESC")
MessageBox.Show(SearchRow(0).Item(0))
```

لاحظ أننا أضفنا عاملاً ثانياً لدالة البحث وهو ما نريد أن نرتب النتيجة بناء عليه وهو رقم الموظف ثم أتبعناه بخيار الترتيب DESC وهو ما يرتب النتيجة من الأعلى إلى الأدنى

استخدام المستعرضات

تتعدد استخدامات المستعرضات لكن أحد استخداماتها الشائعة هي ترشيح عرض الصفوف لذلك تعتبر المستعرضات أحد طرق البحث في مجموعة البيانات.



في المثال التالي سنقوم بإنشاء مستعرض ومن ثم البحث فيه وترتيب الصفوف الناتجة

من عملية البحث :

```
Dim DV As New DataView(Myds.Tables("Person"))
DV.RowFilter = "name='عبد الله'"
DV.Sort = "ename"
```

في المثال السابق قمنا بإنشاء مستعرض للصفوف التي تحتوي اسم عبد الله ثم رتبنا الصفوف في المستعرض بناء على الاسم الأخير.

برنامج البحث في مجموعة البيانات

سوف نبني برنامجا يطبق الأفكار السابقة، فكرة هذا البرنامج هو أن يكون هناك زر اختياري أحدهما للبحث في مجموعة البيانات التي في الذاكرة و الثاني لاسترجاع البيانات من قواعد البيانات ومن ثم البحث فيها. سنقوم بعرض النتيجة على شبكة بيانات. قم بتصميم الشاشة حسب الشكل و حسب الخصائص الموجودة في الجدول :



اسم الأداة	الخاصية	القيمة
Button1	Name	cmdSearch

بحث	Text	
Label1	Name	Label1
	Text	
btnNewData	Name	RadioButton1
البحث بنسخة محدثة	Text	
rdoDSSearch	Name	RadioButton1
البحث في مجموعة البيانات	Text	
DataGrid1	Name	dataGrid
البحث عن المواطنين	Text	Form1
Yes	RightToLeft	

قم بإضافة الكود التالي

```
Public Class SearchProj
    Inherits System.Windows.Forms.Form
    Dim Myds As New DataSet()
    Private Sub cmdSearch_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmdSearch.Click
        Dim con As New OleDb.OleDbConnection()
        Dim sql As String = "select id,name,ename from person"
        con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
            "Data Source=C:\TelDir.mdb;Persist Security
Info=False"
        If rdoNewData.Checked Then
            Myds.Clear()
            Dim daPerson As New OleDb.OleDbDataAdapter(sql, con)
            daPerson.Fill(Myds, "Person")
        End If
        If Myds.Tables.Count = 0 Then Exit Sub
        Dim DV As New DataView(Myds.Tables("Person"))
        DV.RowFilter = "name Like '" & txtEmpName.Text & "''"
        DV.Sort = "ename"
        DataGrid1.DataMember = ""
        DataGrid1.DataSource = Nothing
        DataGrid1.DataSource = DV
    End Sub
End Class
```

الذي يخص موضوعنا من هذا الكود هو الموجود بخط عريض :

١ - نتأكد من أن مجموعة البيانات تحوي على جدول واحد على الأقل

٢ - نعرف المستعرض على أنه مستعرض لجدول الأشخاص

- ٣- عرفنا المرشح، لاحظ وجود النجمة وهو ما يعني أننا نريد البحث عن أول جزء يدخله المستخدم ثم نحضر كل ما يطابق هذا الجزء بغض النظر عن باقي الاسم
- ٤- قمنا بالترتيب حسب الاسم الأخير
- ٥- فرغنا شبكة البيانات من أي بيانات
- ٦- أعدنا عرض البيانات على شبكة البيانات، انظر الشكل النهائي بعد التنفيذ



id	name	ename
٧	عبد لله	أحمد
٣٦	عبد لله	موسى