ADO.net

الموضوع الأول : ال Connection
• ال connection هو المسئول عن الإتصال بين ال Data Source وبيئة ال .net

وهو يتبع ال System.Data namespace
• يدعم ال .net framework وسوف يتم التعامل دائما مع SqlConnection وال OleDbConnection وهما connections نوعين من ال
في الشرح Sql ال .

• خصائص ال Connection :
Data Source . كي يتصل بال Connection يقوم بتوجيه ال string وهو عبارة عن Connection String اهمها ال
والصيغة العامة له هي :
plain text
"Keyword=Value;Keyword=Value;Keyword=Value"

connection . هو أساس ال connection string وبالطبع ال

database . وهي تقوم بارجاع اسم ال database مثل connection توجد خواص اخري لل -
• عمل connection :
1- في ال design time :
ToolBox > Data > SqlConnection

واختياره من خلال اختيار connection string يمكن وضع ال connection الخاصة بال properties وفي قائمة ال
new connection string

2 - في ال run time :
plain text
using SystemData;

Sqlconnection MyConnection;
MyConnection = new SqlConnection;

وكذلك يمكن تحديد ال Connection string :
plain text
MyConnection.ConnectionString = "//put ur connection dtring here//";

• تستخدم ال connection عدد من ال methods مثل :
open,close,CreateCommand
لاحظ أن استخدام أي من ال open أو ال close methods تجعل ال connection ، مثلا : علي هذه الحالة
plain text
MyConnection.open();
DataAdapter.Fill(DataSet);

. ولا يقوم بإغلاقه .. والعكس صحيح open علي حالة ال connection هذا يجعل ال

• يوجد حدثين لل connection : اهمها ال StateChange event وهي تعمل عند تغير حالة ال connection من حالة لأخري
وهذه الحالات هي : open,closed,Broken,Connecting,Executing,Fetching
[CODE]
private void MyConnection_StateChange (object sender,StateChangeEventArgs e)
{
// show the original state
MessageBox.Show(e.OriginalState.ToString());

// show the current state
MessageBox.Show(e.currentState.ToString());

}

****

لخدمات تصميم وبرمجة المواقع

programmer4ever@yahoo.com

00201063879624

****

ع الدرس الثاني : ال commands
والذي عن طريقه يتم تبادل البيانات ، هذا التبادل للبيانات يتم net. وبيئة ال data source عبارة عن الأتصال بين ال connection قلنا أن ال commands . بتنفيذ من ال

• ال command التي سيرجع لها Stored Procedure التي سيتم تنفيذها أو ال SQL ببساطة عبارة عن جملة ال queries (ال data source يقوم بتنفيذ كل أنواع العمليات علي ال command هذا ال non-queries التي لا تقوم بإرجاع قيم أو ال ترجع أي بيانات) .

• عمل command : عن طريق :-
1 - ال design time : وهي ببساطة سحب SqlCommand من ال data من ال toolBox .
2 - ال run time : وهي تتبع ال SqlClient namespace .

```
plain text
using System.Data.SqlClient;

SqlCommand MyCommand;
MyCommand = new SqlCommand();
```

• أهم خواص ال command :
1 - commandText : يحمل جملة ال SQL التي سيتم تنفيذها

```
plain text
MyCommand.CommandText="Select * From Table1";
```

2 - CommandType : يحدد نوع ال command ... text مثلا أو Stored Procedure

```
plain text
MyCommand.CommandType = CommandType.Text;
```

3 - Connection : يحدد ال connection الذي يستخدمه ال command

```
plain text
MyCommand.Connection=MyConnection;
```

parameters و ال transaction كما يوجد ايضا خواص أخري مثل ال

••• ال parameters collection : ويتم اعطائها القيم في ٣ خطوات :
1 - تحديد ال parameter في ال query أو ال stored procedure .
2 - تحديد ال parameter في ال parameters collection .
3 -اعطاء قيم لهذه ال parameters .

- كل parameter يتم استخدامه يجب اضافته باستخدام الدالة add :

```
plain text
MyCommand.Parameters.Add(Name,Value);
```

• أهم ال methods الخاصة بال command :
1 - ExecuteNonQuery : لا تقوم بإرجاع أي row .

2 - ExecuteScalar : تقوم بإرجاع قيمة واحدة .
3 - ExecuteReader : تقوم بإرجاع عدة rows .

مثال لما سبق :

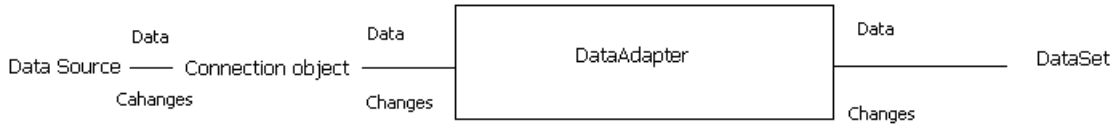
```
plain text
MyCommand.CommandText="Select * From Table1 Where ID=@ID";
MyCommand.CommandType=CommandType.Text;
MyCommand.Connection = MyConnection;

MyCommand.Parameters.Add("@ID",//Value of ID//);

MyConnection.Open();
MyCommand.ExecuteScalar();
MyConnection.Close();
```

to be continued

****

الموضوع الثالث (أو الرابع .. مش فاكر) : ال dataAdapter

• ال DataAdapter التي سيتم شرحها لاحقا.
عبارة عن المادة اللاصقة بين ال DataSource وبين ال DataSet
لل DataSet ثم تقوم بحمل التغيرات من ال DataSet وتعطيها لل connection object عبر ال data source أي أنها تحمل البيانات نت ال data source .



* عمل data adapter :
1) من ال desugn time :
أ- من ال server explorer : إذا كنت عامل connection من ال design time فيمكنك عمل drag لل table أو ال view أو ال Strored procedures علي ال form .
ب- من ال ToolBox :
ToolBox > Data > SqlDataAdapter
يظهر لك ال Data Adapter Configuration Wizard
من خلاله وتكمل عمل ال connection فتقوم بعمل ال data adapter .
2) في ال run time :

```
plain text
using System.Data.SqlClient;
SqlDataAdapter MyAdapter;
MyAdapter=new SqlDataAdapter();
```

• خواص ال data adapter :
1) ال commands : تحتوي ال DataAdapter علي مجموعة من ال commands لتنفيذ ال insert,update,dalete,select.
- إذا قمت بعمل drag لل table أو ال view أو ال Strored procedures علي ال form من ال server explorer فإن ال net. يقوم بتوليد

ال commandText للل DatAdapter commands من خلال ال معها تتعامل أن ويمكنك . أوتوماتيكيا properties windows .
اسمه SqlCommand فمثلا إذا كان لديك , dataAdapter commands الخاصة بك بال commands ومن خلال الكود يمكنك ربط ال -
: فإنه يمكنك DataBase من ال select يقوم بعمل MyCommand

plain text
MyAdapter.SelectCommand=MyCommand;

خارجية commands بدون استخدام DataAdapter مباشرة للل CommandText ويمكنك ايضا اعطاء ال
plain text
SqlDataAdapter da=new SqlDataAdapter("// put the SQL statement here//",MyConnection);

2) ال TableMapping :
في ال table1 في column1 وبين data source في ال table1 في column1 مثلا بين ... DataSet وبين ال data source هي الرابط بين
DataSet .

• ال methods الخاصة بال DataAdapter :
1- ال Fill method : تقوم بنقل البيانات من ال data source إلي ال DataSet , ولها صيغ عديدة .
plain text
MyAdapter.Fill(DataSet);

2- ال Update method : تنقل التغيرات في ال rows في ال DataSet إلي ال data source مرة أخري ، ولها ايضا صيغ عديدة
plain text
MyAdapter.Update(DataSet);

****

عبارة عن مجموعة من العمليات التي يتم التعامل معها علي أنها عملية واحدة ال transaction
فلا يصح نجاح عملية وفشل الأخري... فالكل ناجح أو الكل فاشل.
والمثال الشهير جدا علي هذه العملية هو عمليتي السحب والإيداع في حسابات البنوك ، فعملية السحب اذا تمت ولم تتم عملية الإيداع تضيع النقود
وبالتالي تفشل العملية
واذا لم تتم عملية السحب وتمت عملية الإيداع فهناك إهدار للنقود وتصبح العملية كلها فاشلة.

تستخدم عملية ال- في ال transaction نوعين من ال ADO.Net methods :
1 - commit : لتنفيذ عملية ال transaction .
2 - Rollback : للتراجع عن تنفيذ عملية ال transaction .
وعادة ما يتم استخدام هذه ال methods داخل try & catch كما بالمثال التالي .

المثال يستخدم database اسمها test وال tables :
1 - table1 : وبه ID,name columns
2 - table2 : وبه ID,salary columns

[/CODE]
```
using System.Data;
using System.Data.SqlClient;


//اعمل connection بال database//
string connString = "put ur connection string here";
Sqlconnection connect = new Sqlconnection(connString);
// عرف ال transaction //
SqlTransaction trans = nconnect.BeginTransaction();

// ابدأ التعامل مع ال database بال insert في ال tables //
Sqldataadapter da1 = new Sqldataadapter("Insert Into table1 (ID,name) Values (@ID,@salary",connect);
// هنا parameters لا تنسي وضع قيم ال //
Dataset ds1 = new Dataset();
```

```
Sqldataadapter da2 = new Sqldataadapter ("Insert Into table2 (ID,salary) Values (@ID,@salary),connect);
// هنا parameters لا تنسي وضع قيم ال //
Dataset ds2 = new Dataset();

// ربط ال commands بال transaction //
da1.InsertCommand.Transaction = trans;
da2.InsertCommand.Transaction = trans;

try
{
da1.fill(ds1);
da2.Fill(ds2);

// ابدأ بتنفيذ ال transaction //
trans.Commit();
MessageBox.Show("transaction committed , congratulations !! ");
}
catch
{
trans.Rollbach();
MessageBox.Show("transaction rolled back , sorry !!");
}


using System.Data;
using System.Data.SqlClient;


//اعمل connectionبال database//
string connString = "put ur connection string here";
Sqlconnection connect = new Sqlconnection(connString);
// عرف ال transaction //
SqlTransaction trans = nconnect.BeginTransaction();

// ابدأ التعامل مع ال database بال insert في ال tables //
Sqldataadapter da1 = new Sqldataadapter("Insert Into table1 (ID,name) Values (@ID,@salary",connect);
// هنا parameters لا تنسي وضع قيم ال //
Dataset ds1 = new Dataset();

Sqldataadapter da2 = new Sqldataadapter ("Insert Into table2 (ID,salary) Values (@ID,@salary),connect);
// هنا parameters لا تنسي وضع قيم ال //
Dataset ds2 = new Dataset();

// ربط ال commands بال transaction //
da1.InsertCommand.Transaction = trans;
da2.InsertCommand.Transaction = trans;

try
{
da1.fill(ds1);
da2.Fill(ds2);

// ابدأ بتنفيذ ال transaction //
trans.Commit();
MessageBox.Show("transaction committed , congratulations !! ");
}
catch
{
trans.Rollbach();
MessageBox.Show("transaction rolled back , sorry !!");
```

}

****

لخدمات تصميم وبرمجة المواقع

programmer4ever@yahoo.com

00201063879624

****

ع الدرس الخامس وهو ال DataSet ..
بينها relations وتمثيل ال data عبارة عن هيكل مقيم في الذاكرة يقوم بحمل ال DataSet يمكن اعتبار ال

• انواعها :
1- Typed : ال DataSet تقوم ب "تعرية" ال object . من جداولها بحيث يمكن التعامل معها ك
بأسماءها مباشرة cloumns وال tables ويمكنك التعامل مع هذه ال :
plain text
ex:
OrdersDS.OrdersTable[0].IDColumns;

2-Untyped : يمكنك أن تتعامل مع ال tables وال columns عبر ال tables & items collection
plain text
OrdersDS.Tables["OrdersTable"].Rows[0].Item("IDColumn");

• عمل ال DataSet :
1) في ال Design time :
أ- Typed DataSet :
After adding DataAdapter > Select it > Data menu > Generate DataSet
ب- Untyped DataSet :
ToolBox> Data > DataSet > drag it to the form
2) في ال run time : وهي تتبع ال System.Data namespace
plain text
using System.Data;

DataSet My DataSet;
MyDataSet=new DataSet();

وتكون ال DataSet المعمولة بالكود من النوع Untyped .

• خواص ال DataSet : معظمها مرتبط بال XML . ويوجد ايضا :
tables : يمكن التعامل مع ال 1- Tables collection :
في ال (أ desgin time :
DataSet > Properties window > Relations property > Add
ويمكنك أن تقوم بإضافة tables ويعدها يمكن أن تتعامل مع ال columns جديدة .
في ال(ب rum time :
plain text
DataTable MyTable;
MyDayaSet.Tables.Add("MyTable");

عن طريق table لل columns ويمكنك أن تضيف :
plain text
MyTable.Columns.Add("ID",Type.GetType("System.Int32"));
MyTable.Columns.Add("Name",Type.GetType("System.String"));

DataSet : وتصف العلاقة بين الجداول في ال 2- ال data relations

أ) في ال design time :
DataSet > Properties window > Relations property > Add
relations . وبعدها يمكن عمل customization لل
فقط untyped DataSet هذه الطريقة تخص ال

ب) في ال run time :
plain text
MyTable.Relations.Add("CustomersOrders",MyDataSet.Tables[0].Columns[0],MyDataSet.Tables[1].Columns[0]);

• ال DataSet methods : كما توجد ايضا ال , XML ايضا معظمها خاصة بال clone&copy methods
DataSet : تقوم بعمل نسخة فارغة لل clone ال -1
plain text
DataSet CloneDS=MyDataSet.Clone();

DataSet . تقوم بعمل نسخة كاملة لل Copy ال-1
plain text
DataSet CopyedDS=MyDataSet.Copy();

... انتهي هذا الدرس

****

الدرس السادس : ال DataTable
عبارة عن تمثيل للبيانات في ال DataSet قلنا أن ال memory
.. تحتوي البيانات الفعلية DataTable أما ال

• عمل ال DataTable :
1) في ال design time :
DataSet . لل tables كما قلنا في إضافي ال
2) في ال rum time : تستخدم ال System.Data namespace
plain text
using System.Data;

DataTable MyTable;
MyTable=new DataTable();

add method باستخدام ال DataSet لل tables كما يمكن إضافة ال
Add(),Add(TableName),Add(DataTAble),so on
plain text
MyDataSet.Tables.Add();
or
MyDataSet.Tables.Add(MyTable);

• خواص ال DataTable : اهمها ال Columns,Rows,Constraits
1) ال Columns collection :
New method جديد باستخدام ال column يمكن عمل
New(),New(ColumnName),New(ColumnName,dataType),so on
وأهم خواصها : AllowDBNull,AutoIncrement,Caption,ColumnName,DataType,Maxlength
والمثال :
plain text
using System.Data;

DataColumn MyColumn;
MyColumn=new DataColumn("Name");
MyColumn.DataType=System.Type.GetType("System.String");
//create table and fill the date from the dataset

```
MyDataSet.Tables.Add(MyTable);
MyDataAdapter.Fill(MyDataSet.Tables[0]);
//add the column
MyDataSet.Tables["MyTable"].Columns.Add(MyColumn);
```

2) ال Rows collection : تحتوي علي البيانات الفعلية ..

```
plain text
DataRow MyDataRow;
MyDataRow=MyDataSet.Tables["MyTable"].NewRow();
MyDataRow[0]="7amama";
MyDataSet.Tables["MyTable"].Rows.Add(MyDataRow);
```

row : وهي تقوم بإرجاع حالة ال RowState اسمها rows يوجد خاصية لل

```
Added,Deleted,Detqached,Modified,Changed
plain text
MessageBox.Show(MyDataRow.RowState.Tostring());
```

• ال DataTable methods :

```
Copy,Clone,Clear,Select,etc.
```

خاصية ال- select تقوم بعمل filtering لل rows في ال table نفسه دون تغيير

```
plain text
DataRow drFound;
drFound=MyDataSet.Tables["MyTable"].Select("Name like 'A*'");
foreach (DataRow dr in drFound)
{
MessageBox.Show(dr[0]);
}
```

DataRow array ووضعها في A لكل ما يبدأ بحرف Name اسمه column من select وهذا الكود يقوم بعمل

• ال DataTable methods :

```
ColumnChanged,ColumnChanging,RowChanged,RowChanging,RowDeleted,RowDeleting
```

يتبع,

****

الدرس السابع : ال DataView

ال- DataView تقوم بعمل sorting و filtering هي وسيلة تمكننا من عمل DataRows .

DataView . باستخدام ال rows لهذه ال sorting أو filtering فإنه يمكنني أن اعمل rows به عدد من ال DataTable بمعني إذا كان عندي

تقوم بنفس ما قامت به ال DataView في الدرس السابق .... حسنا ..ال DataTable الخاصة بال Select method كنا قد استخدمنا ال -
DataTable . كما يمكن التعامل معها ك rum time أو ال design time غير أنها عنصر منفصل يمكن عمله في ال Select method

• عمل ال DataView :

1- في ال Design time :

```
ToolBox>Data>DataView
```

لما تريد sorting لكي تعمل sort وال Table ضبط خواص ال properties ويمكنك من نافذة ال

2- في ال Run time :

.. ListBox لأسماء العملاء في loading وأقوم بعمل , MyDataSet في ال OrderTable اسمه Table بافتراض أن عندي
هذا .. DataGrid من حيث اسم العميل ويضغ الناتج في Table لهذا ال filtering فيقوم بعمل ListBox اريد أن اقوم باختيار عميل من ال

الكود يقوم بعمل ما قلنا :

```
plain text
using System.Data;

DataRowView drCurrent;
DataView MyDataView;
```

```
MyDataView=new DataView();

//make the drCurrent equals the selected client
drCurrent=(DataRowView)ClientListBox.SelectedItem;

//set the table for the DataView
MyDataView.Table=MyDataSet.OrdersTable;
//make filtering for the DataView according to the client name
MyDataView.RowFilter="CustomerID='"+drCurrent[0]+"'";

// fill the results in the DataGrid
MyDataGrid.DataSource=MyDataView
```

• خواص ال DataView :
- AllowDelete,AllowEdit,AllowNew ----> تحدد إمكانية تغيير ال data القادمة عبر ال DataView أم لا .
- Count -----> ترجع عدد ال DataRows في ال DataView
- RowFilter,Sort,RowStateFilter ------> تتحكم في ال Data القادمة عبؤ ال DataView .

1) RowFilter : كما في المثال السابق
2) Sort :
plain text
```
MyDataView.Sort="EmployeeID";
```

(Default) تصاعديا (لأن التصاعدي هو ال EmployeeID من خلال ال sorting تقوم بعمل

3) ال RowStateFilter : تقوم بعمل filtering للـ rows حسب حالتها :
plain text
```
MtDataView.RowStateFilter=DataViewRowState.Added;
```

التي تم أضافتها موخرا rows فيقوم بإرجاع ال

• ال DataView methods :
AddNew,Delete,Find
- Find : يقوم بعمل بحث عن ال Primary key عبر ال rows
plain text
```
DataRow [] FoundRows = MyDataView.Find(200);
```

DataRow array ويضعها في ال , ID = 200 التي لها rows فيقوم بارجاع ال

****

ADO.Net هذا الموضوع هو من أفضل المواضيع التي قرأتها في ال
Sorting أو Filtering ويتم من خلاله عمل DataView يتم استخدامه من قبل ال string وهو عبارة عن ... Expressions وهو موضوع ال لل Rows .
ففي الدرس السابق استخدمنا الكود التالي :
plain text
```
MyDataView.RowFilter="CustomerID='"+drCurrent[0]+"'";
```

فالجزء :
plain text
```
CustomerID= ' " + drCurrent[0]+" ' "
```

Rows . لل Filtering لعمل Expression وكما نري استخدمنا هذا ال , Expression هو ما اعنيه بال

• هذه ال strings يتم معاملتها ك String له ال concatinate العادي ، فمثلا يمكن عمل String عادي جدا ، أي انه خاضع لكل ما يخضع له :
plain text
```
MyExpression="MyColumn=' "+Column[0]+" ' ";
```

• ولهذه ال Expressions مجموعة من القواعد :

square brackets : [ ] يتم تضمينها في Special characters الأحرف الخاصة (1

```
plain text
MyExpression="[Miles/Gallons]>10";
```

hash : # يتم احاطتها ب Date قيم التاريخ (2

```
plain text
MyExpression="OrderDate>#08/24/1986#";
```

american format : لاحظ أن كتابة التاريخ يتم بال
mm/dd//yy

aggregate functions يمكن استخدام ال (3 : مثل ال
sum,count,min,max,.....,etc

Comparing : يمكن استخدام قيم المقارنة (4
AND,OR,NOT,<,>,<=,>=,<>,LIKE,IN

• IN : قيمة تساوي مجموعة من القيم :

```
plain text
MyExpression="MyColumn IN ('A','B','C')";
```

• LIKE : مقارنة غير متطابقة :

```
plain text
MyExpression="Name LIKE 'A*'";
```

Arithemtic operations : يمكن استخدام العمليات الحسابية (5
+ , - , * , / , %

****

لخدمات تصميم وبرمجة المواقع

programmer4ever@yahoo.com

00201063879624