[CLEAN USE OF PARAMS TO INSERT AND UPDATE 4](#_Toc119506976)

[USE OF STORED-PROCEDURE TAKING A PARAMETER QUERY 5](#_Toc119506977)

[FIND ID OF NEWLY INSERTED RECORD (IDENTITY, AUTONUMBER) 5](#_Toc119506978)

[STORED-PROCEDURE TAKING A PARAMETER, AND RETURNING RESULTS 5](#_Toc119506979)

[FILTERING USING VIEWS 6](#_Toc119506980)

[Working with DATES 6](#_Toc119506981)

[DATAFORMATSTRING 7](#_Toc119506982)

[DATA-ADAPTORS 8](#_Toc119506983)

[DATAGRIDS and EDITING 8](#_Toc119506984)

[DATAGRID ITEMS 9](#_Toc119506985)

[DATAGRID AND LITERALS 9](#_Toc119506986)

[DATAGRID reference 10](#_Toc119506987)

[DATAGRID AND DROPDOWNLIST 10](#_Toc119506988)

[SNEAKY WAY TO SET CORRECT INDEX ONITEMDATABOUND 10](#_Toc119506989)

[CACHING A DATASET 11](#_Toc119506990)

[ADDING JAVASCRIPT TO A CONTROL 11](#_Toc119506991)

[BOOLEAN ENTRIES 12](#_Toc119506992)

[DATAFORMATSTRING 12](#_Toc119506993)

[COMMAND ARGUMENT PROPERTY 12](#_Toc119506994)

[MOUSE-OVER popup captions 12](#_Toc119506995)

[CYCLING THROUGH DATAGRIDS ON A FORM 12](#_Toc119506996)

[DATES 13](#_Toc119506997)

[SERVER.TRANSFER 13](#_Toc119506998)

[READING DATA USING DATA-ADAPTOR 14](#_Toc119506999)

[WRITING DATA USING DATA-ADAPTOR 14](#_Toc119507000)

[PERMISSIONS PROBLEMS ‘MUST BE AN UPDATEABLE QUERY’ 16](#_Toc119507001)

[DATASETS 16](#_Toc119507002)

[USING VIEWSTATE TO HOLD PARAMS 17](#_Toc119507003)

[APPLICATION LEVEL RECORD LOCKING 17](#_Toc119507004)

[DATAREADER 17](#_Toc119507005)

[RANDOM PASSWORD RESETS 18](#_Toc119507006)

[JAVASCRIPT WINDOW POP-UP 18](#_Toc119507007)

[MODAL POPUP WINDOW – IE only 19](#_Toc119507008)

[BEHAVIOR HTC files 20](#_Toc119507009)

[JAVASCRIPT setTimeOut 20](#_Toc119507010)

[ADVANCED DATATABLE FILTERING – primary keys 21](#_Toc119507011)

[ADVANCED DATATABLE FILTERING – Select, no primary keys 21](#_Toc119507012)

[MORE ON DATAVIEW 22](#_Toc119507013)

[SQL TRASACTION TIMEOUTS 22](#_Toc119507014)

[DATASET TRANSACTIONS 24](#_Toc119507015)

[NON DATASET TRANSACTIONS (Command object) 25](#_Toc119507016)

[VB.NET ARRAYS 25](#_Toc119507017)

[FORMATS revisited 26](#_Toc119507018)

[COOKIES 26](#_Toc119507019)

[APPLYING STYLES TO LISTITEMS 26](#_Toc119507020)

[USING EXCEL AS A DATABASE TABLE 26](#_Toc119507021)

[TRY THROW CATCH 27](#_Toc119507022)

[ASCII AND UTF ENCODING OF STRINGS 27](#_Toc119507023)

[ADDING A NEW ROW TO EXISTING DATASET 28](#_Toc119507024)

[USE A DATAREADER TO POPULATE DROPDOWNS 28](#_Toc119507025)

[REGULAR EXPRESSIONS – STRIP WHITESPACE 28](#_Toc119507026)

[STRINGBUILDER CLASS 29](#_Toc119507027)

[DATAGRID IMG BUTTONS 29](#_Toc119507028)

[DATAGRID LINKBUTTON 30](#_Toc119507029)

[USEFUL LINKS 30](#_Toc119507030)

[MULTIPLE LIST ITEMS SELECTED 30](#_Toc119507031)

[MAILTO 30](#_Toc119507032)

[FILE.COPY AND FILE.MOVE 31](#_Toc119507033)

[OLEDBTYPE 31](#_Toc119507034)

[CUSTOM VALIDATORS 31](#_Toc119507035)

[PARTIAL PAGE VALIDATION 31](#_Toc119507036)

[RETURNING MULTIPLE VALUES FROM FUNCTIONS 32](#_Toc119507037)

[DUMP HTML TABLE DIRECT TO EXCEL 32](#_Toc119507038)

[DEFAULT BEHAVIOUR FOR RETURN KEY 32](#_Toc119507039)

[COMPARISON OF VB.NET AND C# 33](#_Toc119507040)

[CHECKING FOR DUPLICATES WHEN PROCESSING RECORDS 34](#_Toc119507041)

[PERSISTING LOGIN THROUGH COOKIES 34](#_Toc119507042)

[ADDING FILTER ROW TO DATAGRID IN JAVASCRIPT 35](#_Toc119507043)

[JAVA WINDOW POPUP, FEED FORWARD AND BACKWARD 36](#_Toc119507044)

[DATAGRID PAGER 37](#_Toc119507045)

[EXAMPLE WEB.CONFIG FILE 37](#_Toc119507046)

[REMOVE WHITESPACE CHARACTERS 38](#_Toc119507047)

[WEBREQUEST AND SCREENSCRAPING 38](#_Toc119507048)

[CONNECTING TO A TEXT FILE IN EXCEL ADD-IN 39](#_Toc119507049)

[GENERATING AN XL WORKBOOK DUMP OF DATATABLES 40](#_Toc119507050)

[DYNAMIC CONTROLS AT RUN TIME 42](#_Toc119507051)

[SHOW BUSY GIF 42](#_Toc119507052)

[SELF CLOSING WINDOW 42](#_Toc119507053)

[DYNAMIC CLIENT SCRIPT 43](#_Toc119507054)

[DROPDOWN LISTS IN DATAGRID 43](#_Toc119507055)

[REGISTER A CLIENT SCRIPT 44](#_Toc119507056)

[FIRE EVENTS ON PARENT WINDOW FROM CHILD WINDOW 45](#_Toc119507057)

[SOAP CONNECTIONS 45](#_Toc119507058)

[HTML ATTRIBUTES 47](#_Toc119507059)

[BASE64 ENCODE AND DECODE 47](#_Toc119507060)

[FILE UPLOAD FROM WEB PAGE (SUPPORTS XLSX) 47](#_Toc119507061)

[HIDING A WINDOW 48](#_Toc119507062)

[WORKING WITH DATAVIEW BACK TO DATASET 48](#_Toc119507063)

[GENERATING EXCEL REPORTS 48](#_Toc119507064)

[READING EXCEL WITH OLEDB (MORE RELIABLE) 49](#_Toc119507065)

[A FEW DESIGN TIPS FOR WEB DATABASE APPS 50](#_Toc119507066)

[JAVASCRIPT WINDOW EVENTS 50](#_Toc119507067)

[SSL TIPS 51](#_Toc119507068)

[RELATIONAL DATASETS AND BINDING TO A DATAGRID 51](#_Toc119507069)

[CONTEXT CACHE OF DATA 52](#_Toc119507070)

[RANDOM NUMBERS 52](#_Toc119507071)

[STREAM FILES OFF THE SERVER 52](#_Toc119507072)

[CONNECT TO TEXT FILES WITH OLEDB TEXT DRIVER 53](#_Toc119507073)

[NET 2.0 CONVERSION 54](#_Toc119507074)

[ADDING PARAMS TO A QUERY 54](#_Toc119507075)

[CONNECTION STRING 54](#_Toc119507076)

[APP SETTINGS 54](#_Toc119507077)

[VISUAL STUDIO 2008 55](#_Toc119507078)

[DEFAULT BUTTON IN NET 2.0 55](#_Toc119507079)

[TREEVIEW IN NET 2.0 55](#_Toc119507080)

[DATASETS AND GETINSERTCOMMAND 55](#_Toc119507081)

[GET URL OF PAGE 56](#_Toc119507082)

[GRIDVIEW, DROPDOWNLIST SELECTEDINDEXCHANGED EVENT 56](#_Toc119507083)

[GRIDVIEW, FIND ROW SENDING COMMAND 56](#_Toc119507084)

[GRIDVIEW ROW and COLSPAN 58](#_Toc119507085)

[AJAX TUTORIAL 59](#_Toc119507086)

[BUFFERING A QUERY RESULT 60](#_Toc119507087)

[BUFFERING A QUERY – ALTERNATIVE APPROACH 60](#_Toc119507088)

[GLOBALISATION – OSSOWSKI.ORG 60](#_Toc119507089)

[PARSING CURRENCY 61](#_Toc119507090)

[IFRAME CONTROL AND MESSAGE PASSING 61](#_Toc119507091)

[TAB STRIP CONTROLS 61](#_Toc119507092)

[SIMPLE URL BEAUTIFIER for PathInfo 61](#_Toc119507093)

[AJAX revisited for mouseover 62](#_Toc119507094)

[IFRAME SUB WINDOW 62](#_Toc119507095)

[MS-ACESS SET AUTO INCREMENT VALUE 63](#_Toc119507096)

[JAVASCRIPT WINDOW CLOSE WHEN LOST FOCUS 63](#_Toc119507097)

[USING TAB MENUS 64](#_Toc119507098)

[CACHE CONTROL 64](#_Toc119507099)

[WHAT TO USE INSTEAD OF INCLUDE 64](#_Toc119507100)

[CUSTOM CONTROLS 64](#_Toc119507101)

[CSV TEXT PARSER USING .NET CLASS 64](#_Toc119507102)

[Don’t roll your own. VB.NET has FileIO.TextFieldParser. It accepts a text file, but you can also use the StringReader class to feed it a stream. This way you do not need to save the incoming text file to the server. Neat. FileUploadVRD here is the fileupload control. 64](#_Toc119507103)

[It copes with escaped quotes, commas inside data, strings of commas denoting empty fields. The routine above is generic, in the sense it discovers the column headers in the first row of data. 65](#_Toc119507104)

[DETECT SESSION TIMEOUT 66](#_Toc119507105)

[ASP AJAX BUILT IN HANDLERS 68](#_Toc119507106)

[CREATE SITES FOR MOBILE USE 68](#_Toc119507107)

[OLEdb AND DATAADAPTORS ADVANCED 68](#_Toc119507108)

[DIRECT CAST (DirectCast) 69](#_Toc119507109)

[THAT THINGY IN GRIDVIEW TO FIND FIRING ROW 69](#_Toc119507110)

[IFRAMES IN DONET 4 69](#_Toc119507111)

[ROWPSAN AND COLUMN SPAN IN GRIDVIEW 69](#_Toc119507112)

[VB.NET EXTENTION METHODS 69](#_Toc119507113)

[GRIDVIEW AUTOGENERATED COLUMNS, SORTING AND GRIDVIEWROW 69](#_Toc119507114)

[GRIDVIEW FILTERING ROW 70](#_Toc119507115)

[POSTBACK FUN 70](#_Toc119507116)

[FORCE REFRESH OF CSS AND JAVASCRIPT 70](#_Toc119507117)

[CORRECT HANDLING OF WHITESPACE CR/LF 71](#_Toc119507118)

[TREEVIEW; SOME THINGS TO CONSIDER 71](#_Toc119507119)

[GRIDVIEW ADDING CELLS PROGRAMMATICALLY 72](#_Toc119507120)

[WORKING WITH COPY-PASTE FROM XL OR GOOGLE SHEET, SAVE AS XML 73](#_Toc119507121)

[CONFIRM URL IS STILL VALID 74](#_Toc119507122)

[SELECT TOP ROWS FROM A DATASET 75](#_Toc119507123)

[ENUMERATE A DATAVIEW 75](#_Toc119507124)

[ALIGN LEFT AND RIGHT ON SAME ROW WITH CSS 75](#_Toc119507125)

[NAMESPACE 75](#_Toc119507126)

[READ/WRITE XLSX USING EPPLUS 75](#_Toc119507127)

[PARSE DATE INFORMATION FROM CSV FILE 76](#_Toc119507128)

[PULL TAGS FROM PHOTO AND VIDEO FILES 77](#_Toc119507129)

[ISO DATES REVISITED 78](#_Toc119507130)

[BINDING TO AN ARRAY 78](#_Toc119507131)

[CLASSES AND SERIALIZATION 79](#_Toc119507132)

[CLASSES AND OJBECT CLONING 80](#_Toc119507133)

[ARRAYS, LIST(OF…) AND SORTING 81](#_Toc119507134)

# CLEAN USE OF PARAMS TO INSERT AND UPDATE

see http://www.vbdotnetforums.com/showthread.php?p=30943

dim strSQL as string = "INSERT INTO tblComment (DatePoint,Comment,IDrequest,IDUser) VALUES (@p1,@p2,@p3,@p4)"

dim objCmd as new OleDbCommand(strSQL, objConn)

(this is equiv to objCmd.CommandText = strSQL)

objCmd.Parameters.Add("@p1", fntoday())

objCmd.Parameters.Add ("@p2", txt1.Text)

objCmd.Parameters.Add ("@p3", hiddenReq.value)

objCmd.Parameters.Add ("@p4", Session("IDuser"))

ObjConn.Open()

objCmd.ExecuteNonQuery()

ObjConn.Close()

That’s it. No messing with Type.

Alternative way

objCmd.Parameters.Add ( "@p1", OleDbType.VarChar,15 ).Value = fntoday()

objCmd.Parameters.Add ( "@p2", OleDbType.VarChar,255 ).Value = txt1.Text

objCmd.Parameters.Add ( "@p3", OleDbType.VarNumeric ).Value = hiddenReq.value

objCmd.Parameters.Add ( "@p4", OleDbType.VarNumeric ).Value =

Session("IDuser")

**Actually, you’ll still need to use the odd specific type, e.g. Date**

**objCmd.Parameters.Add ("@p2", OleDbType.Date).Value= Now()**

objCmd.Parameters.Add ("@p1", ctype(e.Item.FindControl("dgE\_SiebelActivityID"),TextBox).Text)

objCmd.Parameters.Add ("@p4", cbool(ctype(e.Item.FindControl("dgE\_ExistingCustomer"),DropDownList).SelectedItem.Value))

# USE OF STORED-PROCEDURE TAKING A PARAMETER QUERY

Dim ObjConn as New OleDbConnection(strConn)

dim objCmd as new OleDbCommand("qryClone",objConn)

objCmd.CommandType = System.Data.**CommandType.StoredProcedure**

objCmd.Parameters.Add ("pIDrequest", Viewstate("Target"))

ObjConn.Open()

objCmd.ExecuteNonQuery() ‘returns # records affected

objConn.Close()

The above calls qryClone in the database, which itself expects a single param. It’s an action query, returning no results.

Note: when debugging, you might see a no default value error. E.g. pIDrequest has no default value. Check that the value you are adding is not null, it’s not an error in the query parameterization itself.

# FIND ID OF NEWLY INSERTED RECORD (IDENTITY, AUTONUMBER)

'\*\*\* http://www.mikesdotnetting.com/Article/54/Getting-the-identity-of-the-most-recently-added-record

This only works if you keep same connection open and use it on the same command object.

ObjConn.Open()

objCmd.ExecuteNonQuery() ‘\*\*\* execute our insert query

objcmd.CommandType = CommandType.Text

objcmd.CommandText = "Select @@Identity"

Dim newID As Long = objCmd.executeScalar()

ObjConn.Close()

# STORED-PROCEDURE TAKING A PARAMETER, AND RETURNING RESULTS

Note here, even though we are using a stored proc with a defined param, its actually a SELECT query, so the commandType is TEXT. In example below the qry itself expects @p1 to be passed to it. We then further limit results with extra adhoc params.

dim objCmd as new OleDbCommand("SELECT \* FROM qrySalesGUIeditor WHERE IDRequest=@p10", oleConn)

objCmd.CommandType = System.Data.CommandType.Text

objCmd.Parameters.Add ("@p1", OleDbType.UnsignedBigInt).value = x

objCmd.Parameters.Add ("@p10", OleDbType.UnsignedBigInt).value = x

Dim reader As OleDbDataReader = objCmd.ExecuteReader()

It’s a good idea to have clear param names in the adhoc string which do not conflict with the stored proc params

See

<http://authors.aspalliance.com/stevesmith/articles/sprocs.asp>

<http://www.dotnet247.com/247reference/System/Data/OleDb/OleDbType/__members>

you can also use executeScalar see

<http://aspnet.4guysfromrolla.com/articles/062905-1.aspx>

# FILTERING USING VIEWS

dim objDA as New OleDbDataAdapter()

objDA.SelectCommand = objCmd

Dim objDS as New Dataset()

objDA.Fill(objDS,"results") ‘give table a name

'Clever bit on how to filter/sort results http://aspnet101.com/aspnet101/tutorials.aspx?id=31

'you replace myT.Datasource = objDS with

' myView = objDS.Tables("results").DefaultView

' myT.Datasource = myView

'which allows you you manipulate myView with filter/sort

dim myView as DataView

objConn.Close()

'using myView you can apply SQL WHERE type operations

myView = objDS.Tables("results").DefaultView

myView.RowFilter = "ATG like 'a%'" 'NOTE \* and % seem to have same effect # for single numeric

myView.Sort = "mins DESC"

'myT.Datasource = objDS.Tables("results").DefaultView

myT.Datasource = myView

myT.DataBind()

# Working with DATES

When using a query, you do need to specify the oledbtype

objCmd.Parameters.Add("@p1", OleDbType.Date).Value= Now()  
  
use .Add and not .AddWithValue

you can use AddDate(“d”, -1 , Now()) to give you 24 hrs prior for db compares etc.

function fnToday() as String

'\*\*\* returns ten leftmost chars of ISO date, YYYY-MM-DD

fnToday =left(Now.ToString("s"),10)

end function

Note that for record locking purposes, you should use an ISO datestring and store as a string in the database. ASP.NET is flakey when it comes to date comparisons within SQL strings.

DATAFORMATSTRINGhttp://aspnet.4guysfromrolla.com/articles/041702-1.2.aspx   
The nice thing about these BoundColumn controls is that they contain some formatting properties, such as:

* **HeaderText** - Specifies the text that should appear in the column's header.
* **FooterText** - Specifies the text that should appear in the column's footer. (Remember to set ShowFooter to True if you want to display a footer in your DataGrid!)
* **HeaderStyle / FooterStyle / ItemStyle** - Contains all of the properties that the DataGrid's styles did. Useful for centering columns, specifying fore/background color, etc.

Note on this one;  
<asp:TemplateColumn HeaderText="Delete" ItemStyle-HorizontalAlign="center">  
will centre the header and rows, but NOT the footer. You need a separate footerstyle entry  
<headerstyle cssClass="grey" />  
<footerstyle HorizontalAlign="center" />

* **DataFormatString** - Specifies formatting instructions. (See the live demo for some examples; refer to the docs for a full listing of format specifications.)

<asp:DataGrid runat="server" id="dgPopularFAQs"

BackColor="#eeeeee" Width="85%"

HorizontalAlign="Center"

Font-Name="Verdana" CellPadding="4"

Font-Size="10pt" AutoGenerateColumns="False">

<HeaderStyle BackColor="Black" ForeColor="White"

Font-Bold="True" HorizontalAlign="Center" />

<AlternatingItemStyle BackColor="White" />

<Columns>

<asp:BoundColumn DataField="CatName" HeaderText="Category Name" />

<asp:BoundColumn DataField="Description" HeaderText="FAQ Description" />

<asp:BoundColumn DataField="ViewCount" DataFormatString="{0:#,###}"

HeaderText="Views" ItemStyle-HorizontalAlign="Center" ItemStyle-Wrap=”false” />

<asp:BoundColumn DataField="SubmittedByName" HeaderText="Author" />

<asp:BoundColumn DataField="SubmittedByEmail" HeaderText="Author's Email" />

<asp:BoundColumn DataField="DateEntered" HeaderText="Date Added"

DataFormatString="{0:MM-dd-yyyy}" />

</Columns>

</asp:datagrid>

DataFormatString="{0:c0}” is $nn,nnn no decimals

DataFormatString="{0:n0}” is nn,nnn no decimals

# DATA-ADAPTORS

Private Sub PrintRows ( ByVal myDataSet As DataSet )

Dim myTable As DataTable

Dim myRow As DataRow

Dim myColumn As DataColumn

' For each table in the DataSet, print the row values.

For Each myTable in myDataSet.Tables

For Each myRow In myTable.Rows

For Each myColumn in myTable.Columns

Response.Write ( myRow ( myColumn ) )

Next myColumn

Next myRow

Next myTable

End Sub

DataSet.Update

The DataSet.Table.Columns(“colname”) returns a column

Properties are

MaxLength

DataType

Dunno how much help this is, compared to just validating all data, setting input lengths and doing an UPDATE query

<http://msdn2.microsoft.com/en-US/library/x7h2bwa6.aspx> defines column properties

Note, to add a datacolumn:

Tables(0).Columns.Add(New DataColumn("Extended", GetType(Double)))

# DATAGRIDS and EDITING

**BoundColumn**

<asp:boundcolumn datafield="RecCreateDate" headertext="Date" DataFormatString="{0:yyyy-MM-dd}"/>

When you go into edit mode, this turns into a text box. You can stop this with ReadOnly=”true”

**TemplateColumn**

<asp:TemplateColumn HeaderText="Capex">

<ItemTemplate>

<%# DataBinder.Eval(Container.DataItem, "CapexUSD", "{0:c0}") %>

</ItemTemplate>

<EditItemTemplate>

<asp:TextBox ID="dgE\_CapexUSD" columns="8"

Text='<%# DataBinder.Eval(Container.DataItem, "CapexUSD", "{0:c0}") %>' Runat="server"/>

</EditItemTemplate>

</asp:TemplateColumn>

With template column, there is no dataformatstring, but instead if you use DataBinder.Eval, you can specify a format string as a 3rd argument.

Note that the item will be read only in edit mode unless you specify an EditItemTemplate.

See also, in terms of identifying event source.  
<https://stackoverflow.com/questions/7808104/asp-net-getting-datakey-from-gridview-on-edit-and-delete?rq=1>

This works

Dim dgi As DataGridItem = TryCast(CType(sender, Control).NamingContainer, DataGridItem) 'generic cast control, and then cast its container to datagriditem

Dim myCB As CheckBox = dgi.FindControl("chkGID")

dg\_vrd.DataKeys(dgi.ItemIndex) gives the datakey value for that item

# DATAGRID ITEMS

<http://authors.aspalliance.com/aspxtreme/sys/web/ui/webcontrols/ListItemTypeClass.aspx>

e.Item.ItemType = Item | AlternatingItem | EditItem | Header | Footer | Pager | SelectedItem | Separator

# DATAGRID AND LITERALS

Boundcolumns are easy, but unfortunately the only way to read from them is using e.items.cells(x) where x is the 0 based index of the column in question. This makes processing the datagrid (either onItemDataBound, or OnItemCommand) a pain if you later add columns / move them around.

A better way to do this is to render the item as an asp:literal and give it an id.

<asp:templatecolumn headertext="Literal">

<ItemTemplate>

<asp:Literal id="myLiteral" text='<%# databinder.eval(container.dataitem,"account")%>' runat="server" />

</ItemTemplate>

</asp:templatecolumn>

You can then reference this control using e.Item.findcontrol

sub doCommand(sender as Object, e as DataGridCommandEventArgs)

dim myL as literal

myL = e.Item.findcontrol("myLiteral")

if NOT myL is Nothing then response.write(myL.text)

end sub

# DATAGRID reference

Datagrids are class System.Web.UI.WebControls

The datagrid.items collection holds DataGridItem members which are the rows of the datagrid

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dv_vstechart/html/vbtchTopQuestionsAboutASPNETDataGridServerControl.asp>

# DATAGRID AND DROPDOWNLIST

Data is posted back from the edit line

**strCategoryID = CType(e.Item.FindControl("lstCategories"), \_**

**DropDownList).SelectedItem.Value**

**strCategoryName = CType(e.Item.FindControl("lstCategories"), \_**

**DropDownList).SelectedItem.Text**

You cast the object as a dropdownlist and then find the .selectedItem.text or .value note that selectedindex is the numeric nth value selected

When binding you need to use a function call to iterate the items in the list, and then set the selected index of the one which matches the Container.Dataitem(x). There is no ‘bound column’ equivalent

# SNEAKY WAY TO SET CORRECT INDEX ONITEMDATABOUND

You can define a bogus attribute in the dropdownlist

<asp:DropDownList ID="drop1" runat="server" oldV="arse">

oldV is available within x= e.Item.findcontrol(“drop1”)

will be x.Attributes(“oldV”)

it will also be rendered in the browser and accessible on the page to JAVA. But it will NOT be posted back – its value will be the default empty value. You can fix this by writing its value into ViewState see <http://msdn2.microsoft.com/en-us/library/yhzc935f.aspx>

but this is questionable faff – its only worth it if you really need java to modify the value on the page and have it posted back.

Also <http://www.codeproject.com/aspnet/DblPanePickList.asp>.

I think the issue here is that the Viewstate overrides the javascript changes made on the client (but strangely if the user makes the changes, they are picked up).

But this does mean you can do oldV = ‘<% Container.DataItem(“field”) %>’ in the datagrid and then run through the list collection at bind time to set the correct index. Saves you having to do another db list call when the grid is first rendered.

Wasted bloody hours on this but to Modify the DropDownList onDataBound, you are actually looking for the control in the **e.ItemType.EditItem** row, not the alternating or listitem entries.

Below is a generic sub that mimicks a databound field. This should be called from a databinder routing that iterates for each oCell in e.Items.Cells

sub setDropDown(oCell as object)

'\*\*\* pass a cell, with oldV or oldT attribute and this sub will set the dropdown

'\*\*\* note that x will be case sensitive

dim x as string

for each myControl as Control in oCell.Controls

if TypeOf(myControl) is DropDownList then

try

dim myD as DropDownList = CType(myControl,DropDownList)

x = myD.attributes("oldT")

if x <> "" then myD.Items.FindByText(x).Selected = true

x = myD.attributes("oldV")

if x <> "" then myD.Items.FindByValue(x).Selected = true

finally

end try

end if

next

end sub

BE WARNED! The above only works at bind-time. If you try directly setting selected=true in a Page Load event, it will fail with error “you cannot select more than one item”

The workaround is to find the IndexOf the item you want, and then set the SelectedIndex value

myD.SelectedIndex = myD.Items.IndexOf(myD.Items.FindByValue(<value>))

# CACHING A DATASET

If you want have a datagrid that displays a dynamic dropdown, then you need to bind this during the datagrid onDataBound event. Means you need the dataset for the drop down to hand. You don’t want to have to keep creating it, so best thing is to cache it either in the viewstate or the Session object

**ViewState** .**Add**("myData", ds)

To retrieve the **DataSet** from **ViewState**

Dim x as **DataSet** = CType(**ViewState**.Item("myData"), **DataSet**)

Remember to set the <asp:dropdown> correctly

<asp:DropDownList id="fld\_LocationA" runat="server"

DataTextField="LongNodeName" DataValueField="IDlocation" bindTV="V"/>

# ADDING JAVASCRIPT TO A CONTROL

Call this within a onItemdatabound (or OnItemCreated) event sub

dim myB as Button = e.Item.findcontrol("btnWithdraw")

if not (myB is nothing) then myB.Attributes.Add("onClick","return jv1(this)")

# BOOLEAN ENTRIES

<asp:CheckBox Runat=server Enabled=False Checked='<%

#DataBinder.Eval(Container.DataItem, "active")%>'/>

# DATAFORMATSTRING

|  |  |
| --- | --- |
| **Format character** | **Description** |
| **C** | Displays numeric values in currency format. |
| **D** | Displays numeric values in decimal format. |
| **E** | Displays numeric values in scientific (exponential) format. |
| **F** | Displays numeric values in fixed format. |
| **G** | Displays numeric values in general format. |
| **N** | Displays numeric values in number format. |
| **X** | Displays numeric values in hexadecimal format. |

{0:c0} is currency

{0:p} is percent to 2dp. {0:p0} is percent no dp. {0:##.00%} also

# COMMAND ARGUMENT PROPERTY

<asp:LinkButton runat=”server” id=”btn1” Text=”view this” CommandName=”details”

CommandArguement=‘<%# DataBinder.Eval(Container.Dataitem, “title\_id”) %>’ />

Sub dg\_ItemCommand(sender as object, e as DataGridCommandEventArgs)

Dim strTitleID as string = e.CommandArguement

End sub

# MOUSE-OVER popup captions

<td TITLE=”this is popup text” >

This is all nice, but datagrid <templateitem> won’t let you create this property, nor will <asp:literal> so I don’t see how we can emit to a page. <asp:textbox does let you add your own tags>

I think this can be fixed if we create a custom-class version of the asp literal control.

It DOES work if you add the title to the cell, via a boundcolumn

# CYCLING THROUGH DATAGRIDS ON A FORM

**METHOD1**

'\*\*\* 1st control is the form, perhaps body is 0th. Can't use Controls("form1"). If you have dg buried in a table/div somewhere I assume you could use in Page.FindControl(“divthing”).Controls

for each myC as control in Page.Controls(1).Controls

if (TypeOf(myC) is Datagrid) then

dim myD as Datagrid = myC

myD.Datasource = objDS.Tables(myD.id)

myD.DataBind()

end if

next

The above expects to find various tables defined through objDA.Fill(objDS,"dg\_balance")

**METHOD2**

‘\*\*\* create a set of tables, then use the Tablename to find the datagrid in the page.

for each tbl as DataTable in objDS.Tables

response.write(tbl.TableName)

dim myD as Datagrid = page.findcontrol(tbl.TableName)

myD.Datasource = objDS.Tables(tbl.TableName)

myD.DataBind()

next

# DATES

<http://www.microsoft.com/technet/scriptcenter/guide/sas_vbs_ufbq.mspx?mfr=true>

Datepart(“yyyy”,Now) returns the year as an integer

Quarter is returned by q

Integer 1-12 month by m

You can also use Now.ToString(“s”) s returns UTC date and time

UTC article regarding databases

<http://aspnet.4guysfromrolla.com/articles/081507-1.aspx>

Also format(now,”dd/MM/yyy”) works.

Note, DataFormatString in webcontrols takes “{0:c}” as an example for currency. The 0 represents the index of the element of the string to format.

X = datetime.utcNow() gives current UTC date

# SERVER.TRANSFER

You can use server.transfer to pass values between pages on the same server. To pass a value do this;

PageA.

Context.Items("txtBoxAText")=”thing I want to send”

Server.Transfer(“PageB.aspx”,true) ‘true posts the values across

On pageB

Dim foo as string =Context.Items("txtBoxAText")

response.write("RET=" & foo)

note that you have to pick the value up on the first page load, because it will be lost on subsequent postbacks

The other way to do it is use a Session variable, this is persistent.

Session(“returnURL”)=’return path’

Session.remove(“returnURL”) clean up before returning

Server.Transfer is a little bit neater because the site URL does not appear to change. But, if the receiving page needs to self-post, you will find it changes the URL. So in balance, there’s no real benefit, other than you can transfer any object

Also with “true” you can read passing form values in the request object

x = Request.Form(“TextBox1”)

Note, if you create public get classes for the items to be read off the source page, then these values will automatically appear in Context.items when you navigate to the second page.

http://www.developer.com/net/asp/article.php/3299641

# READING DATA USING DATA-ADAPTOR

Datagrids are fine for tables of data, but if you have a form, how do you write back all those fields easily?

Dim objDS as New Dataset()

objDA.Fill(objDS,"results")

‘so now we have a dataset with a results table in it.

<http://www.15seconds.com/issue/031223.htm> shows the DS structure

Dim s As String

s = CType(myDS.Tables("Customers").Rows(0).Item("CustomerID"), String)

Which takes row 0 of customers table, Item (“CustomerID”) to string

We can build a binder by reading all items in a row, and then looking for controls with a matching name, and populating them. It works, but you’ll have to build for Boolean controls and dropdowns etc.

# WRITING DATA USING DATA-ADAPTOR

The dataset is a memory cache. So you can modify it but the changes do not get written back to the db.

To write the data, you need to use a DataAdapter method.

<http://msdn2.microsoft.com/en-us/library/system.data.oledb.oledbcommandbuilder.aspx>

The short of it is that your DataAdapter will read the changes in the DataSet, then decide to do an INSERT, UPDATE or DELETE query. But to do this, you need to at least create an automatic command for it.

‘\*\*\* Make sure, you specify any param oledbtype explicitly in the query

‘\*\*\* Also, this DOES NOT WORK if you try to write back to queries built from multiple tables

Dim ObjConn as New OleDbConnection(strConn)

dim objCmd as new OleDbCommand("SELECT \* FROM tblGUID", objConn)

dim objDA as New OleDbDataAdapter()

objDA.SelectCommand = objCmd

Dim objDS as New Dataset()objDA.Fill(objDS,"tblGUID")

'\*\*\* need this next line to write back to db

Dim builder As OleDbCommandBuilder = New OleDbCommandBuilder(objDA)

Dim as myTable DataTable =objDS.Tables("tblGUID")

‘\*\*\* change a record

myTable.Rows(0).Item("FullName")="bonkers"

builder.GetUpdateCommand()

‘\*\*\* This next line skips any errored fields

objDA.ContinueUpdateOnError="true"

'\*\*\* Without the OleDbCommandBuilder this next line would fail.

objDA.Update(objDS, "tblGUID")

IMPORTANT: Fieldnames must NOT CONTAIN SPACES else OleDbCommandBuilder will FAIL. Set ContinueUpdateOnError=false to debug the errors

BEWARE SQL RESERVED WORDS! E.g. if your table contains Currency as a field name, this won’t save, neither will GUID. See <http://office.microsoft.com/en-us/assistance/HP010322491033.aspx>

**CAUTION!** You will get an error <<OleDbCommand.Prepare method requires all parameters to have an explicitly set type>> if you use params in the source query, but haven’t set their type

objCmd.Parameters.Add ("@p1",myID) FAILS WITH ERROR

objCmd.Parameters.Add ("@p1",OleDbType.unsignedBigInt).value = myID WORKS

Note that objDS.AcceptChanges() just modifies the DS and if you accept changes, you clear the state info that the DA needs to make a write to db decision. So don’t use objDS.AcceptChanges()

<http://msdn2.microsoft.com/en-us/library/33y2221y.aspx>

<http://msdn2.microsoft.com/en-us/library/y2ad8t9c.aspx>

<http://msdn2.microsoft.com/en-us/library/z1z2bkx2.aspx>

http://msdn2.microsoft.com/en-us/library/system.data.dataset.getchanges.aspx

Note, you can use

Dim myD as new Dataset

myD = objDS.getChanges()

then test myD is nothing (or for sub tables is nothing)

# PERMISSIONS PROBLEMS ‘MUST BE AN UPDATEABLE QUERY’

I wasted 4 hours on this. The answer is that the db security settings must be set to read/write for all users as well as System. This is easy enough on the Verizon server because you can right-click on the db to set its security permissions.

On the Web-hosting.uk.com site, you cannot change file permissions so, after many hours I found

<http://msdn2.microsoft.com/en-us/library/wyts434y.aspx>

you need to set up a web.config file containing

<configuration>

<system.web>

<identity impersonate="true" />

</system.web>

</configuration>

This works like a charm. The IIS server can extend its permission to the target database and magically updates are now possible.

<http://www.velocityreviews.com/forums/t100862-what-is-the-use-of-identity-impersonate-true.html>

# DATASETS

<http://msdn2.microsoft.com/en-us/library/feh3ed13.aspx>

DataSet1.Tables("Customers").Rows(0).Delete()

Finding specific record(s)

Dim s As String = "primaryKeyValue"

Dim foundRow As DataRow = DataSet1.Tables("AnyTable").Rows.Find(s)

If foundRow IsNot Nothing Then

MsgBox(foundRow(1).ToString())

Else

MsgBox("A row with the primary key of " & s & " could not be found")

End If

Adding columns to dataset

<http://www.thescarms.com/dotnet/AddColumns.asp>

Removes Datacolumns.remove(“colname”)

Private Sub TestAndRemove(ByVal colToRemove As DataColumn)

' Get the DataColumnCollection from a DataTable in a DataSet.

Dim columns As DataColumnCollection = \_

DataSet1.Tables("Orders").Columns

If columns.Contains(colToRemove.ColumnName) Then

columns.Remove(colToRemove)

End If

End Sub

Dim dt As DataColumn

dt = ds.Tables("Orders").Columns("Employeeid")

ds.Tables("Orders").Columns.Remove(dt)

Working with columns to modify data

<http://msdn2.microsoft.com/en-us/library/tat996zc.aspx>

Dim customerRow() As Data.DataRow

customerRow = DataSet1.Tables("Customers").Select("CustomerID = 'ALFKI'")

customerRow(0)("CompanyName") = "Updated Company Name"

customerRow(0)("City") = "Seattle"

# USING VIEWSTATE TO HOLD PARAMS

You can persist a hidden-field equivalent either as <asp:textbox id=”blah” visible=false runat=server/> which does not display it in the page source, or you can put it directly into the viewstate.

Viewstate.Add(“myParam”,somevalue) to set

X = viewstate(“myParam”) to get

Note that myParam is case sensitive

# APPLICATION LEVEL RECORD LOCKING

Problem with Date field types in oleDB is that if you pull this value into Viewstate(“recL”,..record..) or onto the web page, then you cannot easily test recLock=@p1 because date may be reformatted when processed by server.

To get around this, use Now().ToString(“s”) to generate a STRING value ISO datetime string. Note lower case s. The db record needs a dedicated 20 chr string field to hold reclockstring.

# DATAREADER

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfSystemDataOleDbOleDbDataReaderClassTopic.asp>

Dim reader As OleDbDataReader = objCmd.ExecuteReader()

if reader.Read() then

response.write(reader(0).ToString())

if reader(1).isDBnull then…..

reader.item(“field1”) also works for indexing

etc. There are a bunch of methods.

Comparing two hashedByte arrays (e.g. if you are using MD5 password encryption)

<http://support.microsoft.com/kb/301053>  
<https://stackoverflow.com/questions/12979212/md5-hash-from-string>

its actually easier to compare in an SQL statement.

# RANDOM PASSWORD RESETS

To quickly generate a random Hex string

<http://aspnet.4guysfromrolla.com/articles/101205-1.aspx>

Public Function GetRandomPasswordUsingGUID(ByVal length as Integer) as String

'Get the GUID

Dim guidResult as String = System.Guid.NewGuid().ToString()

'Remove the hyphens

guidResult = guidResult.Replace("-", String.Empty)

'Make sure length is valid

If length <= 0 OrElse length > guidResult.Length Then

Throw New ArgumentException("Length must be between 1 and " & guidResult.Length)

End If

'Return the first length bytes

Return guidResult.Substring(0, length)

End Function

# JAVASCRIPT WINDOW POP-UP

javascript:moveTo(screen.availWidth-794,-4);

window.resizeTo(800,screen.availHeight+4)

<http://www.experts-exchange.com/Web/Web_Languages/Q_20788214.html>

For controlling a POPup window.

Easiest way is put this java on the source page

<script language="JavaScript">

<!--//

function new\_window(url) { link = window.open(url,"Link","toolbar=0,location=0,directories=0,status=0,menubar=0,scrollbars=1,resizable=1,width=600,height=500,left=40,top=50");

}

//-->

</script>

Note: using window.open(url,x, ….)

Passes x to the opened window as window.name

You can use this to target behaviour on the opened window.

e.g. if you opened a main frame, then the sub frame would use parent.name to pick up x

And invoke with

btnOpen.Attributes.Add ("onClick", "new\_window('test.aspx')")

or

<asp:HyperLink id="lnkOpen" runat="server">Open</asp:HyperLink

lnkOpen.Attributes.Add("onClick", "new\_window('test.aspx')")

Not as easy as you want….

You can pop a new window, but you can’t server transfer back to the original one. Instead the new window starts to navigate.

<http://www.faqts.com/knowledge_base/view.phtml/aid/876/fid/124>

But the above might work – it’s a modal dialogue window. Er no – whilst the variable passing works well, problem is we cannot copy from just plain text on the target window (could be because there are no borders etc).

<http://www.irt.org/script/window.htm> more window stuff

<http://www.gridviewguy.com/ArticleDetails.aspx?articleID=114>

ergh gets messy with frames..

<http://www.js-examples.com/forums/viewtopic.php?p=6861&sid=d1bca8bd64faadc59caa44000575b37a>

this tells you about window.opener

<http://www.javascriptkit.com/javatutors/remote2.shtml>

<http://www.devguru.com/Technologies/ecmaScript/quickref/window.html>

Here’s the answer.

You open the frame window with x = window.open method.

From within any child frame, you can always find the top level frame holder, as top is always the top.

top.close() will close the window holding the frames

top.opener.document.title will show the title of the window that opened the remote window which holds the frames.

Opener allows you to communicate backward. So

top.opener.document.forms[0].somefield.value is a way to back populate values on the very first originating form

# MODAL POPUP WINDOW – IE only

The Javascript below opens target.htm, passing dummy to it.

The window is modal, and returns r which can then be used to write to parent page (presumably you can also use DOM on r.opener? as it’s the window obj)

function showRemotewindow() {

var r = showModalDialog('Target.htm', 'dummy', 'dialogWidth:500px;dialogHeight:500px;center:1;');

//dummy is passed to the window that is opened, its available in window.dialogArguments

if (typeof r != "undefined") {

// split returned result string and write

var r\_array=r.split("|")

document.form1.fld\_GLMRC\_USD.value=r\_array[0]

document.form1.fld\_GLNRC\_USD.value=r\_array[1]

}

}

# BEHAVIOR HTC files

Useful for mouseover and maxlength. Means you don’t have to faff around attaching javascript to your buttons, you just set the behavior in the class. Also, the element itself can pass up a variable. E.g. MaxLength2=55

<http://msdn.microsoft.com/msdnmag/issues/01/01/cutting/>

simple mouseover htc

<PUBLIC:HTC>

<PUBLIC:ATTACH event="onmouseover"

handler="fnOver"/>

<PUBLIC:ATTACH event="onmouseout"

handler="fnOut"/>

<script LANGUAGE="jscript">

function fnOver()

{element.style.color="red";}

function fnOut()

{element.style.color="";}

</script>

</PUBLIC:HTC>

And to use…

input { behavior: url(mouseover.htc); rest of style here }

# JAVASCRIPT setTimeOut

Useful if you want to have flashing buttons etc

the function called must be in quotes, and the timer value is ms

var FLASH=false

function flashOn(){

if (form1.btnSave.className=="btnGreen") {form1.btnSave.className="btnRegular"}

else {form1.btnSave.className="btnGreen"}

setTimeout("flashOn()",800)

FLASH=true

}

Trigger the above with

If (FLASH==false) {flashOn()}

Var FLASH is used to avoid multi-triggering the function

The above cannot be cancelled, but is reset on page repost

Side note on classes – some clever code in here <http://onlinetools.org/articles/unobtrusivejavascript/cssjsseparation.html>

To apply multiple classes, separate each with a space

<input type=”button” class=”special red thing”>

# ADVANCED DATATABLE FILTERING – primary keys

DataAdaptor.Fill does not copy through the db schema, you need to do this explicitly

Dim objDA as New OleDbDataAdapter()

objDA.SelectCommand = objCmd

Dim myDataset as New DataSet

myDataset.Tables.Add(new DataTable())

objDA.FillSchema(myDataSet.Tables(0), SchemaType.Mapped)

objDA.Fill(myDataSet.Tables(0))

note myDataset.Tables.Add(new DataTable(“table1”))

also works, allowing you to address myDataSet.Tables(“table1”)

this should also work

Dim myDataset as New DataSet

objDA.FillSchema(myDataSet, SchemaType.Mapped, “table1”)

objDA.Fill(myDataSet,”table1”)

you can then use the DataRowCollection.Find method

Dim foundRow As DataRow = myDataSet.Tables(0).rows.Find("BW64")

Which searches the primary key, returning a single row. BUT above line assumes just a single primary key. If you have multiple you have to define an array of keys to pass first and I am not sure if you can wildcard any.

IMPORTANT: If you are setting parameters for objCmd, then you need to set their types explicitly. E.g.

objCmd.Parameters.Add ("@p1", OleDbType.Char,10).Value = "%"

<http://aspnet.4guysfromrolla.com/articles/101602-1.3.aspx>

<http://samples.gotdotnet.com/quickstart/howto/doc/adoplus/adorstodataset.aspx>

# ADVANCED DATATABLE FILTERING – Select, no primary keys

This technique does not use primary keys. It searches on a field, and returns a DataRow array (hence () declaration)

Dim foundRows() As DataRow = inTable.Select("[IDthing]='banana'")

if ubound(foundRow) = -1 then did not find any

<http://msdn.microsoft.com/msdnmag/issues/03/10/DataPoints/>

# MORE ON DATAVIEW

http://www.asp101.com/articles/stuart/dataview/default.asp

Datasets come with a defaultView, which is a Dataview, so you can apply all the same methods to this

DataSet1.Tables(0).DefaultView.Sort = "CompanyName"

DataSet1.Tables(0).DefaultView.RowFilter = "Country = " & strCountry

MyDataGrid.DataSource = DataSet1.Tables(0).DefaultView

MyDataGrid.DataBind()

DataView itself consists of .item(0) etc for each of the datarowview items that it holds, these reference the actual table rows

You can apply .rowfilter to the dataview

myView.item(0).item("ID") is the first datarowview, and we are then indexing the column ID of the underlying dataset row that this holds

# SQL TRASACTION TIMEOUTS

I developed an application that worked fine in development and single user testing, but when multiple users starting using the application, I ran into some cases when users got timeout errors from ASP while the code was trying to execute a long SQL stored procedure. This long stored procedure uses a transaction because it involves a lot of updates, inserts, and deletes, and if any statement fails, then all of the statements must be rolled back to the state that they were in before the transaction began. When the timeout was occurring from ASP, the data in my database was getting messed up because some of the statements in the transaction were being executed but then the transaction was just stopped in the middle of execution without being rolled back.

I wanted to find out a way to detect when a timeout occurs so that I could then manually tell the stored procedure to rollback the transaction. I started searching for how to do this, and I found that I couldn't find much information on this topic.

I started out trying to use the LOCK\_TIMEOUT value in the SQL Server stored procedure on all of my SQL statements and to then detect if this was exceeded. However, I learned that if the LOCK\_TIEMOUT value is exceeded, your transaction skips the blocked operation and receives trappable error number 1222 with the message "Lock request time out exceeded." In this situation, the blocked statement is cancelled and the rest of the transaction continues as soon as a locking conflict occurs. A transaction isn't rolled back if you exceed the lock timeout. Exceeding the timeout skips the blocked operation, but any remaining SQL statements in the transaction still execute. You need to trap for error 1222 and take appropriate action if you want to roll back any user-defined transactions. This was unacceptable to me because it is too much of a hassle to write your own error-detection code after each statement in your stored procedure and then to write the code to rollback the transaction.

Well, I finally was able to figure out how to detect the timeout and to rollback the transaction from the ASP page in the case that a timeout does occur. It is kind of a roundabout way of doing things, but it works. I still declare a transaction inside my stored procedure and use the normal method of detecting errors and conditions where I need to rollback the transaction from within the stored procedure. However, I also begin a transaction from within the ASP page by calling adoConn.BeginTrans in my ASP code. Then I set a CommandTimeout value for the maximum time I want to allow for this stored procedure to execute, and if the ADO command takes longer than this specified time, I check for the timeout error and if it exists, then I rollback the transaction from the ASP code with RollbackTrans. Else, I go ahead and commit the transaction with CommitTrans.

It took me a lot of research and testing to figure this out, but it now works perfectly! Just thought I would pass along what I learned. Here is a code excerpt.

-----------------------------------------------------

Dim colErrs, objError, timeout, adoConn, adoCmd

timeout=False

Set adoConn = Server.CreateObject("ADODB.Connection")

adoConn.CursorLocation = adUseClient

adoConn.Open Session("UserConnStr")

adoConn.IsolationLevel = adXactReadUncommitted

'The next line starts a transaction

adoConn.BeginTrans

Set adoCmd = Server.CreateObject("ADODB.Command")

'I added this to give the command 45 seconds to execute.

adoCmd.CommandTimeout = 45

adoCmd.ActiveConnection = adoConn

adoCmd.CommandText = "sproc\_RecalculateQuantities"

adoCmd.CommandType = adCmdStoredProc

adoCmd.Execute

Set colErrs=adoConn.Errors

If adoConn.Errors.Count <> 0 then

For Each objError In colErrs

'This is the error number for a timeout.

If objError.Number=-2147217871 Then

adoConn.RollbackTrans

Response.Write "The query timed out before finishing. Please try again."

timeout=True

adoConn.Errors.Clear

Exit For

End If

Next

End If

Set adoCmd = Nothing

If Not timeout Then

adoConn.CommitTrans

End If

adoConn.Close

Set adoConn = Nothing

-------------------------------------------------------

Hope that this helps you. I know that I will be using this code a lot from now on for my most complex stored procedures involving transactions.

LaDonna Schneller, MCSE

Sr. Software Engineer

Decision Sciences Inc.

LSchneller@dsiwebsite.com

# DATASET TRANSACTIONS

For many transactions you are working with a oledbCommand object and you associate the transaction with this. But if you need to do bulk changes on an entire dataset, you need to associate the transaction object with the DataAdaptor

‘\*\*\*\* First open the connection object and fill the dataset

Dim oDA2 As New OleDb.OleDbDataAdapter("SELECT \* FROM tbldata;", oleConn)

Dim oDS2 As New DataSet

oDA2.Fill(oDS2, "data")

‘\*\*\* now modify the dataset

‘\*\*\* now we are ready to write back the changes through the DA.

‘\*\*\* create a builder off the DA

Dim builder As New OleDb.OleDbCommandBuilder(oDA2)

builder.quotePrefix=’[‘

builder.quoteSuffix=’]’

builder.GetUpdateCommand()

'oDA2.ContinueUpdateOnError = "true" ‘optional

‘\*\*\* begin transaction after builder creation

Dim myTrans = oleConn.BeginTransaction(IsolationLevel.ReadCommitted)

‘\*\*\* TRICKY part, you assign the transaction to the DA SELECT command, because this is used by the builder to derive the update, delete, insert etc commands

oDA2.SelectCommand.Transaction = myTrans ‘do before invoking update

‘\*\*\* now run the transaction itself

oDA2.Update(oDS2, "data")

myTrans.Commit()

‘trap errors and use myTrans.rollback for these, but use an IS NOTHING test on myTrans before rolling back

WARNING: See <http://social.msdn.microsoft.com/Forums/en-US/adodotnetdataproviders/thread/22185515-963d-40e5-875f-55c07aae99b4/>

You need to create separate oledbCommand objects and persist these

Dim oInsert As OleDb.OleDbCommand = Builder.GetInsertCommand()

And then separately assign the transaction to this object

oInsert.Transaction = myTrans

then it works. oDA2.InsertCommand.Transaction = myTrans will not work because it seems this object does not persist properly. (bit like the old Access currentdb object problem)

IMPORTANT note on KEYWORDS. Some field names might conflict with SQL keywords, use builder.quotePrefix=’[‘ and builder.quoteSuffix=’]’

Before calling getInsertCommand.

<https://www.oreilly.com/library/view/adonet-cookbook/0596004397/ch04s15.html>  
  
UPDATE 2022-08-23. Tested with SQL server. If modifying or adding to a dataset, you only need to add the transaction to the select command object.

# NON DATASET TRANSACTIONS (Command object)

You can also use transactions on the command object

Dim ObjConn As New OleDb.OleDbConnection(strConn)

Dim myTrans As System.Data.OleDb.OleDbTransaction

Try

ObjConn.Open()

myTrans = ObjConn.BeginTransaction(IsolationLevel.ReadCommitted)

Dim objCmd As New OleDb.OleDbCommand("DELETE \* FROM tblSpecificRequirement WHERE BidNumber=@p1;", ObjConn)

objCmd.Parameters.Add("@p1", fld\_BidNumber.Text)

objCmd.Transaction = myTrans

objCmd.ExecuteNonQuery()

…repeat for multiple new cmd objects and set their transaction = myTrans

…And then finally commit

myTrans.Commit()

ObjConn.Close()

Catch ex As Exception

myTrans.Rollback()

Finally

ObjConn.Dispose()

End Try

# VB.NET ARRAYS

Dim b() as String = split(fld\_something.text,vbLf)

‘\*\*\* declares an undimensioned string array and fills it with split results

Dim arr() as String = {"fld\_Customer","fld\_isExisting"}

Explicitly declares a fixed array of values

The System.Array class itself has various properties/methods. See

<http://msdn2.microsoft.com/en-US/library/system.array_properties.aspx>

e.g. you can use b.length instead of ubound(b)

ACTUALLY, be careful as b.length is not as reliable as ubound

You can also sort, but you cannot use .Trim() that’s only available on the individual elements. Array.ForEach is supposed to allow you to apply quickly to each element but I cannot work it out

# FORMATS revisited

**X = FormatNumber(***value* [, *trailing digits*] [, *leading digit*] [, *parentheses*] [, *group digits*]**)**

**X = FormatCurrency(***value* [, *trailing digits*] [, *leading digit*] [, *parentheses*] [, *group digits*]**)**

**X = FormatPercent(***value* [, *trailing digits*] [, *leading digit*] [, *parentheses*] [, *group digits*]**)**

In both cases, the [] params are boolean

# COOKIES

You need to write to the Response object and read from the Request object

You also need to do an IS NOTHING test before attempting to read.

Response.Cookies("LeaveGUID").Value = Session("GUID")

Response.Cookies("LeaveGUID").Expires = DateTime.Now.AddYears(1)

Note, you can also have keys in a cookie

Request.Cookies(“Item1”)(“SubItem1”) = “banana”

Request.Cookies(“Item1”)(“SubItem2”) = “orange”

But it is not that simple. Just checking for a cookie in the Request object will create a null cookie with an immediate expiry, so you may overwrite your original cookie values.

<http://www.codeproject.com/aspnet/aspnetcookies.asp>

if you want to modify the cookies on your page, you must first copy the request cookie (if it exists) to the response cookie in on page\_load. Then all code on your page reads/writes against the response cookie. You MUST set the expiry of the response cookie on page load. <http://www.thescarms.com/dotnet/cookies.asp>

# APPLYING STYLES TO LISTITEMS

<http://msdn2.microsoft.com/en-us/library/system.web.ui.webcontrols.style.aspx>

# USING EXCEL AS A DATABASE TABLE

You need a connection string thus, note that IMEX=1 is important if you have a preamble of blank or text rows above a column of numeric data. It forces the system to recognise the numeric values. Set HDR =NO if you have no header info.

Dim strXLconn As String = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=\\ausyd1esof1\rates\mybook.xls;Extended Properties=""Excel 8.0;IMEX=1;HDR=NO"""

Dim myDA As New OleDbDataAdapter("SELECT TOP 550 \* FROM [Sheet1$]", strXLconn)

IMPORTANT: If you want to write back to the spreadsheet, your connection string must omit IMEX=1 or you will get an ‘query must be updateable’ error.

See <http://blog.lab49.com/archives/196>  
Seems you still have a problem as XL driver will scan first 8 rows to determine type, and it chooses text, it will not return any numeric values it encounters, just nulls

# TRY THROW CATCH

Error trapping blocks are as follows;

Try

…something

….your own error trapping here

…dim myex as New exception(“Am throwing my hands up”)

Throw myex

Catch ex as System.ArgumentException

Trace.warn(“specific error related to say writing db text to numeric”)

‘\*\*\* this catch block is optional

Catch ex as Exception

‘\*\*\* this block will catch everything else

Trace.warn(ex.toString()) show the error

Finally

… do some clean ups. Finally is optional, catch is mandatory

End try

<http://www.codeproject.com/dotnet/exceptionbestpractices.asp#Don%27tthrownewException%28%297>

<http://www.dotnetspider.com/tutorials/ExceptionClasses.aspx>

# ASCII AND UTF ENCODING OF STRINGS

You can tidy up input strings and force them to 7 bit ascii by use of the Encoding object

dim objASCII as new ASCIIEncoding()

'\*\*\* the Default method of Encoding is a string of 8 bit bytes.

dim myCleanDataString as string = System.Text.Encoding.Default.GetString(myData)

'\*\*\* strip the nulls and 254,255. Then write file as act.csv

myCleanDataString = myCleanDataString.Replace(ControlChars.NullChar, "")

myCleanDataString = myCleanDataString.Replace(chr(254), "")

By default ASP.NET tries to handle strings as double-byte Unicode. So above code reads myData (by array) as single 8-bit chars.

UTF7Encoding class will support 7-bit pairs of Unicode chars. There’s also UTF8encoding. I don’t think there is a way to convert 8 bit to 7 bit, unless you scan the byte array and perform an AND operation with 127

# ADDING A NEW ROW TO EXISTING DATASET

[**http://msdn2.microsoft.com/en-us/library/5ycd1034(VS.80).aspx**](http://msdn2.microsoft.com/en-us/library/5ycd1034(VS.80).aspx)

For an untyped set;

Dim newCustomersRow As DataRow = DataSet1.Tables("Customers").NewRow()

newCustomersRow("CustomerID") = "ALFKI"

newCustomersRow("CompanyName") = "Alfreds Futterkiste"

DataSet1.Tables("Customers").Rows.Add(newCustomersRow)

# USE A DATAREADER TO POPULATE DROPDOWNS

Datareader is faster than a Dataset

myDropdown.datasource=objcmd.executeReader(commandBehavior.closeConnection)

myDropdown.bind

# REGULAR EXPRESSIONS – STRIP WHITESPACE

ASP.NET regex is different to that in classic ASP <http://authors.aspalliance.com/stevesmith/articles/dotnetreplace.asp>

<https://regex101.com/>

<%@ Import Namespace="System.IO" %>

<%@ Import Namespace="System.Text.RegularExpressions" %>

<script language="VB" runat="server">

Sub SubmitBtn\_Click(sender As Object, e As EventArgs)

Dim strInput As String = Text1.Text

Dim strOutput As String = Regex.Replace(strInput, "\s+", " ")

output.Text = strOutput

End Sub

</script>

^matches at start of string, $ at end

See for more, including groups

<http://www.ddj.com/dept/windows/184416603>

You need to bear in mind that .NET regex uses the Matches method to return a MatchCollection. The Match method will only return a single match.

Dim m as Match = regex.match(sPayload,"(<)(.\*)(>\r\n)")

‘\*\*\* returns the first match, even though other exist

‘\*\*\* code below will find all matches

dim mc as MatchCollection = regex.matches(sPayload,"(<)(.\*)(>\r\n)")

for each m in mc

‘\*\*\* 0 is entire match, 2nd is the .\* group here

trace.warn(m.groups(2).tostring())

next

regex.ismatch(sPayload,”exp”) is a quick notation to test for 1 or more matches. Note that / does not need escaping in .Net

CASE: use regex.ismatch(sPayload,”exp”, RegexOptions.IgnoreCase)

There are other options to handle single line see:

<http://www.regular-expressions.info/dotnet.html>

Use of Match object

Dim MatchObj As System.text.RegularExpressions.Match = System.Text.RegularExpressions.Regex.Match(changes.Value, "\(([^,]+),(\w+),([\w-]+),(\w\*)\)")

Do Until Not MatchObj.Success

'note MatchObj.Groups(0) is the entire match, all submatches come after 1,2,3..

If MatchObj.Groups(2).Value = 0 Then…

‘\*\*\* find next match

MatchObj = MatchObj.NextMatch()

Loop

Note (blah)? Is an optional group ?:(discard) will discard the group

# STRINGBUILDER CLASS

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpconusingstringbuilderclass.asp>

Dim myS as new Stringbuilder

Mys.append(“text”)

myS.append(“more”)

x = mys.ToString()

# DATAGRID IMG BUTTONS

<http://www.startvbdotnet.com/aspsite/controls/image.aspx>

<asp:TemplateColumn HeaderText=”Delete”

ItemStyle-HorizontalAlign=”center” ItemStyle-Wrap=”false”>

<ItemTemplate>

<asp:ImageButton runat="server" ImageURL="smlTrash.gif"

CommandName="Delete" CausesValidation="false">

</asp:ImageButton>

</ItemTemplate>

</asp:TemplateColumn>

Private Sub ImageButton1\_Click(ByVal sender As System.Object, ByVal e As \_

System.Web.UI.ImageClickEventArgs) Handles ImageButton1.Click

Label1.Text = "You clicked the image button at" & e.X & " " & e.Y

End Sub

# DATAGRID LINKBUTTON

CausesValidation: Gets/Sets whether the link button performs validation in other controls

CommandArgument: Gets/Sets an optional argument holding data about the command specified with CommandName

CommandName: Gets/Sets the command name for this button

Text: Gets/Sets the text displayed in the link button

# USEFUL LINKS

FAQ <http://www.syncfusion.com/FAQ/aspnet/WEB_c6c.aspx#q133q>

Whitespace <http://msmvps.com/blogs/bill/archive/2005/06/05/50497.aspx>

# MULTIPLE LIST ITEMS SELECTED

Asp:Dropdownlist does not allow multiple selections, but you could use an HTML control instead and then use this code to determine values

<http://www.velocityreviews.com/forums/t100880-how-to-get-the-values-from-multiple-selection-in-the-list-box-aspnet-program.html>

In practice, an account browser might be simpler to emit a datagrid with a checkbox and a filter mechanism and allow user to (un)Select and save.

# MAILTO

Useful when the SMTP server goes down

<http://www.angelfire.com/dc/html-webmaster/mailto.htm>

<A HREF="mailto:?subject=look at this website&body=Hi, I found this website and thought you might like it http://www.geocities.com/wowhtml/">tell a friend</A>

you may notice that there's no email address in the mailto example and although the link in the web address is not clickable in the body of the email, it is when it arrives at the recipient's inbox

# FILE.COPY AND FILE.MOVE

VB.NET - copy with overwrite  
File.Copy("C:\Dir1\SomeFile.txt", "C:\SomeOtherDir\SomeFile.txt", True)

<http://www.aspfaqs.com/aspfaqs/ShowFAQ.asp?FAQID=207>

# OLEDBTYPE

<http://msdn2.microsoft.com/en-us/library/system.data.oledb.oledbtype.aspx>

oledbtype.date

oledbtype.boolean

oledbtype.bigint

oledbtype.decimal

oledbtype.double

oledbtype.unsignedBigInt - record indexes

# CUSTOM VALIDATORS

Put this code in the datagrid. Remember that a blank value will not trigger the validator (you need required field validator to catch this)

<asp:CustomValidator OnServerValidate="custom1" ControlToValidate="fld\_startDate" ErrorMessage="must be yyyy-mm-dd" runat=”server”/>

And this in the code-behind

Sub custom1(ByVal sender As Object, ByVal args As serverValidateEventArgs)

'\*\*\* checks sender for valid date format

args.IsValid = False

If System.Text.RegularExpressions.Regex.IsMatch(args.Value, "^\d{4}-\d{2}-\d{2}$") Then

args.IsValid = IsDate(args.Value)

End If

End Sub

# PARTIAL PAGE VALIDATION

Page.isValid lets you down when you have validators on both the EditItem and Footer in a datagrid. If you try edit-and-update, the hidden footer validators cause page.isvalid to become false. Answer is to do a partial validation. Code below assumes you name all editor controls something like edit\_myDG\_Control1 and all add controls as add\_myDG\_Control1. you then call partial validation (“edit\_”) or (“add\_”) as required

function isPartValid(x as string) as boolean

'\*\*\* looks at the page validator controls, returning true for add/edit validators

'\*\*\* x = edit\_ or add\_ which are the naming conventions I use for datagrid editors

dim myV

for each myV in page.validators

if regex.isMatch(myV.controlToValidate,x) then

myV.validate

if NOT myV.isValid then return false

end if

next

return true

end function

<http://www.regular-expressions.info/dotnet.html>

# RETURNING MULTIPLE VALUES FROM FUNCTIONS

In VB.NET you need to declare a structure, then use this as the single return value from a function

public Structure myRate

public ListNRR as double

public SalesFloorNRR as double

public ListMRR as double

public SalesFloorMRR as double

end Structure

function doIt()as myRate

doit.ListNRR = 1

doit.SalesFloorNRR = 22

etc

return

end function

dim myUseRate as myRate = doit()

trace.warn(myUseRate.ListNRR)

etc

# DUMP HTML TABLE DIRECT TO EXCEL

<http://www.codingforums.com/archive/index.php?t-71472.html>

# DEFAULT BEHAVIOUR FOR RETURN KEY

If you have a search field and want the form to submit when user types in this field and hits return

Page load – assign this to the search box

txtSearch.Attributes.Add("onKeyDown","return okd()")

<script language="javascript">

function okd(){

if((event.which && event.which == 13) || (event.keyCode && event.keyCode == 13)) {

document.form1.btnSearch.click();return false;}

else return true;

}

</script>

On the page

<asp:textbox id="txtSearch" runat="server"/>

<asp:button text="Search" id="btnSearch" runat="server" onClick="doSearch"/>

# COMPARISON OF VB.NET AND C#

<http://www.codeproject.com/dotnet/vbnet_c__difference.asp>

TIMEZONE INFORMATION

By default, getTimezoneOffset returns the time zone offset in minutes, and it is negative for places like Australia which are ahead of GMT

setCookie("**timezone**", new Date().getTimezoneOffset()/60);

function setCookie(name, value, expires, path, domain, secure)

{

document.cookie= name + "=" + escape(value) +

((expires) ? "; expires=" + expires.toGMTString() : "") +

((path) ? "; path=" + path : "") +

((domain) ? "; domain=" + domain : "") +

((secure) ? "; secure" : "");

}

USE OF ODBC to read/write XL spreadsheets  
First off you need to have a schema.ini file in your root directory  
[act.csv]

Format=TabDelimited

ColNameHeader=true

MaxScanRows=10

CharacterSet=ANSI

[Analytic.csv]

Format=TabDelimited

ColNameHeader=true

MaxScanRows=10

CharacterSet=ANSI

These control the way the ODBC text driver tries to read the file.  
Dim odbcDataset As New DataSet

Dim OdbcConn As New Odbc.OdbcConnection(strODBCConn)

Dim oDA As New Odbc.OdbcDataAdapter("SELECT \* FROM Analytic.csv", OdbcConn)

oDA.Fill(odbcDataset)

There are additional connection string params that control how the file is handled;

“Driver={Microsoft Text Driver (\*.txt; \*.csv)}; Dbq=\\ausyd1esofrept\pricing\PCMpipeline\;Extensions=asc,csv,tab,txt"

'This is connection for a text file or csv file without Header   
connCSV.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" \_   
& path & ";Extended Properties='text;HDR=NO;FMT=Delimited'"   
  
'This is connection for a text file with Header (i.e., columns   
'connCSV.Open "Driver={Microsoft Text Driver (\*.txt; \*.csv)};Dbq=" \_   
& path & ";Extensions=asc,csv,tab,txt;HDR=NO;Persist Security Info=False"

Problems: the driver will scan ahead a few lines, set by MaxScanRows but it tries to work out what the field types are. If you set a whole bunch of text preamble in a spreadsheet (e.g. because it’s a customer facing sheet) it can stuff you up if you want to write numeric values in the sheet. You cannot skip rows. <http://msdn2.microsoft.com/en-us/library/ms709353.aspx> indicates you can force columns to a particular type, so maybe this is how to work around the problem.

<http://www.codeproject.com/KB/database/GenericParser.aspx?print=true> is a parser, might be of use.

More info on the topic; apparently you can use a schema.ini like this;

[filename.txt]

Format=FixedLength

ColNameHeader=False

Col1=FieldName1 Char Width 30

Col2=FieldName2 Date Width 15

Col3=FieldName3 Integer Width 15

Col4=FieldName4 Float Width 20

CharacterSet=ANSI

So here we are explicitly setting types for the columns and (possibly) renaming the columnheaders. <http://www.iwsinc.com/Support/PDFs/tsb5-1.pdf>

CONNECTING TO AN XL SPREADSHEET

Note that you have to provide the filepath and filename in the connstring

Driver={Microsoft Excel Driver (\*.xls)};DriverId=790;Dbq=C:\MyExcel.xls;

You then use the following select command

SQL syntax "SELECT \* FROM [sheet1$]". I.e. excel worksheet name followed by a "$" and wrapped in "[" "]" brackets

Note: you cannot write back to the worksheet unless it has some sort of indexing column

# CHECKING FOR DUPLICATES WHEN PROCESSING RECORDS

One way is to build a string with a concatenation of the records and use instr to test.

A cleaner way is to use an array list

Dim trapDupe As New ArrayList

trapDupe.Add(dr("CallType") & dr("Destination")) '\*\*\* flag this pairing as processed

If trapDupe.Contains(dr("CallType") & dr("Destination")) Then ‘\*\*\* reject etc

# PERSISTING LOGIN THROUGH COOKIES

Sessions will timeout after 20 mins. Use the page\_load event to search for cookie to update session. But watch out because there is a request and a response cookies collection.

'\*\*\* first copy the request cookie objects thro to the response. Create it in the response if its missing

'http://www.thescarms.com/dotNet/cookies.aspx

If Request.Cookies("PLNTID") Is Nothing Then

Response.Cookies.Set(New HttpCookie("PLNTID", "SomeValue"))

Else

Response.Cookies.Set(Request.Cookies("PLNTID"))

End If

'\*\*\* all tests/ operations are now on the RESPONSE cookie

If Session("NTID") & String.Empty = "" Then

'\*\*\* no session, check cookie

If Request.Cookies("PLNTID").Value = "SomeValue" Then

'\*\*\* lost cookie value, so login

Context.Items("returnURL") = "..\privateLine2\PrivateLine\_Main.aspx"

Server.Transfer("..\NTID\NTIDLogin.aspx", True)

Else

'\*\*\* cookie good, so restore session and persist cookie

Session("NTID") = Response.Cookies("PLNTID").Value

End If

ElseIf Response.Cookies("PLNTID").Value = "SomeValue" Then

'\*\*\* good session, but cookie missing its value, so update this

Response.Cookies("PLNTID").Value = Session("NTID")

Trace.Warn("updated cookie")

End If

'\*\*\* can only reach this point with both a good session and good cookie, so persist cookie

Response.Cookies("PLNTID").Expires = DateTime.Now.AddHours(24)

If you logout, you need to delete the cookie also by setting expiry -1

# ADDING FILTER ROW TO DATAGRID IN JAVASCRIPT

It’s difficult to programmatically add a filter row in VB.NET, so another approach is to add this to the rendered table in Javascript on the client.

function jvAddFooter(){

var myTable=document.getElementById('dg\_Market')

if (!myTable) return;

myTableFoot = document.createElement("TFOOT"); //or THEAD

mycurrent\_row=document.createElement("TR")

mycurrent\_cell=document.createElement("TD")

var newInput = document.createElement("<input type=\"text\" name=\"srcService\" >")

newInput.value=document.form1.fld\_Service.value

newInput.size="12"

mycurrent\_cell.appendChild(newInput)

mycurrent\_row.appendChild(mycurrent\_cell)

..repeat as required…

mycurrent\_cell=document.createElement("TD")

var newInput = document.createElement("<input type=\"Submit\" name=\"btnSearch\" value=\"search\" >")

mycurrent\_cell.appendChild(newInput)

mycurrent\_cell.colSpan=5

mycurrent\_row.appendChild(mycurrent\_cell)

myTableFoot.appendChild(mycurrent\_row)

myTable.appendChild(myTableFoot)

}

<body onload="jvAddFooter(); window.focus();">

When this gets posted back to server, you can examine Request(“srcService”) etc for all these items. They will NOT persist across page serves, hence the java above expects to find a set of hidden fields to repopulate these search criteria with.

If Page.IsPostBack Then

'\*\*\* is user running a search?

If Request("btnSearch") <> String.Empty Then

fld\_NodeA.Value = Request("srcNodeA")

fld\_NodeB.Value = Request("srcNodeB")

fld\_Bandwidth.Value = Request("srcBandwidth")

fld\_Service.Value = Request("srcService")

fld\_Customer.Value = Request("srcCustomer")

dg\_Market.CurrentPageIndex = 0 '\*\*\* reset pager index

End If

Else…..

# JAVA WINDOW POPUP, FEED FORWARD AND BACKWARD

Say you have a datagrid of items, and you want to double-click an entry to pop open a sub window with an item picker.

On the main form, include this javascript and it will be called onDblClick=jvPicker(this) from the datagrid field.

function jvPicker(x){ link= window.open("CPE\_itempicker.aspx",x.id ,"width=850,height=588,scrollbars=1,resizable=1"); link.focus(); }

It passes the datagrid control reference id to the window it opens.

On the target form, Body onLoad calls jv1();

function jv1(){

//pick up the reference fields from window.opener and then repost this form

if (document.Form1.hdn\_postbackfield.value == ""){  
 var w = window.name document.Form1.hdn\_postbackfield.value = w //id of return field document.Form1.hdn\_opco.value= window.opener.document.forms[0].fld\_OPCO.value // change w to point to the vendor value w = w.replace(/\_PartCode/i, "\_Vendor") document.Form1.hdn\_vendor.value = window.opener.document.getElementById(w).value; document.Form1.submit()}

else{

<asp:literal id=jsA text=”return” runat=server/>  
window.opener.document.getElementById(document.Form1.hdn\_postbackfield.value).value=myA[1] document.Form1.hdn\_postbackfield.value=document.Form1.hdn\_postbackfield.value.replace(/\_PartCode/i, "\_Vendor") window.opener.document.getElementById(document.Form1.hdn\_postbackfield.value).value=myA[0] window.close() } }

The way this works is that when the window first opens, the postbackfield value is empty, so this is populated with the window name (reference) passed from the parent. We also modify that ref with replace so we can pull up another seed value.

Make sure the hidden ref fields are server controls so they persist on page serves.

Form then self-posts, and on next serve, will execute the branch with the javascript literal in it. By default this will return doing nothing. But once user picks an item, the server code modifies the literal to define an array myA=[‘vendor’,’partcode’] as return values and then the rest of the code executes to paste these back to the parent and then closes the child window.

# DATAGRID PAGER

Implementing a pager

<asp:datagrid …. PageSize="40" AllowPaging="True" ….>

<PagerStyle HorizontalAlign="Left" PageButtonCount="5" Mode="NumericPages"/>

Then in the pager handler in the code behind

dg\_MyDatagrid.CurrentPageIndex = e.NewPageIndex

rebindMyDatagrid()

Remember that if you do any filtering of results, you need to reset CurrentPageIndex=0

# EXAMPLE WEB.CONFIG FILE

The below demonstrates ODBC connections, plus extended page execution time and also increased file upload sizes

<?xml version="1.0" encoding="utf-8" ?>

<configuration>

<appSettings>

<add key="strConn" value="PROVIDER=MICROSOFT.JET.OLEDB.4.0;DATA SOURCE=\\localhost\pricing\db\warehouse.mdb;" />

<add key="strODBCconn" value="Driver={Microsoft Text Driver (\*.txt; \*.csv)}; Dbq=\\localhost\pricing\warehouse\;Extensions=asc,csv,tab,txt" />

<add key="strConn2" value="PROVIDER=MICROSOFT.JET.OLEDB.4.0;DATA SOURCE=\\localhost\pricing\db\warehouse.mdb;Jet OLEDB:Database Password=monster;" />

<add key="jetConn" value= "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=\\ausyd1esofrept\pricing\warehouse\;Extended Properties='text;HDR=Yes;FMT=Delimited';" />

<add key="strUserFiles" value="\\ausyd1esofrept\pricing\warehouse\" />

<add key="strFileTypes" value=".zip,.ppt,.pdf,.txt,.csv,.doc,.xls,.rtf,.gif,.vsd,.tiff,.msg"/>

<add key="MailServer" value="mailhost7.sg.mci.com" />

<!-- alternative mail servers are

mailhost6.hk.mci.com

mailhost7.sg.mci.com

smtp.ap.mcilink.com

-->

</appSettings>

<system.web>

<!--http://msdn.microsoft.com/en-us/library/e1f13641.aspx-->

<httpRuntime useFullyQualifiedRedirectUrl="true" maxRequestLength="81920" executionTimeout="300" />

<!-- CUSTOM ERROR MESSAGES

<customErrors mode="Off" />

</system.web>

</configuration>

# REMOVE WHITESPACE CHARACTERS

Easy way to do this is with regex and a hex range of ASCII

myT.Text = System.Text.RegularExpressions.Regex.Replace(myT.Text, "[^\x21-\x7E]", String.Empty)

ASCII char set here. <http://www.blast.com/software/data_pump/DP_ASCII_Char.html>

# WEBREQUEST AND SCREENSCRAPING

See option at http://authors.aspalliance.com/damianm/article/4/2/WebClient.aspx.vb.view.aspx

Dim wc As New System.Net.WebClient

Dim s As System.IO.Stream

Dim sr As System.IO.StreamReader

Dim q As New System.Collections.Specialized.NameValueCollection

‘\*\*\* q is the query string

q.Add("text", "test text")

q.Add("password", "secret")

q.Add("checkbox", "on")

q.Add("textarea", "a longer text sentence")

q.Add("submit", "submit")

wc.QueryString = q

s = wc.OpenRead("http://authors.aspalliance.com/damianm/article/4/2/formreader.aspx")

'read in the page

sr = New System.IO.StreamReader(s)

litWeb.Text = sr.ReadToEnd()

'--tidy up

sr.Close()

s.Close()

# CONNECTING TO A TEXT FILE IN EXCEL ADD-IN

You can save on the hassle of a local database by downloading a csv text file from the server, then using ADODB to connect to it with SQL. Remember to set a reference to Access Data Objects 2.7 in the project.

Private Declare Function URLDownloadToFile Lib "urlmon" Alias \_

"URLDownloadToFileA" (ByVal pCaller As Long, ByVal szURL As String, ByVal \_

szFileName As String, ByVal dwReserved As Long, ByVal lpfnCB As Long) As Long

URLDownloadToFile(0, sURL, sPath, 0, 0)

‘\*\*\* Where sURL is server call, and sPath is the local full path

Dim connS As String

connS = GetSetting("PIPtool", "Settings", "LOD") & "\" ‘\*\*\* folder containing the saved file

connS = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & connS & ";Extended Properties='text;HDR=Yes;FMT=Delimited';"

Dim oConn As ADODB.Connection

Set oConn = New ADODB.Connection

oConn.Open (connS)

Dim rs As New ADODB.Recordset

rs.Open "SELECT DISTINCT [COUNTRY] FROM PIP2temp.txt", oConn

Schema.ini would normally sit in the same folder as the target file, but it seems for CSV files you do not need it. See also <http://www.connectionstrings.com/?carrier=textfile>

You need to watch out when generating the csv on the webpage. If a field value contains a comma (x2C) or a quote (x22) then the whole value needs to be enclosed in quotes. The quote chars occurring in the field need to be expressed as double quotes. '\*\*\* see http://www.avdf.com/jan98/art\_vb002.html

For Each myRow As DataRow In objTable.Rows

For i = 0 To objTable.Columns.Count() - 1

'\*\*\* 2008-11-14 we need to escape comma and single quotes in the csv output else this will confuse the field delimeters

'\*\*\* see http://www.avdf.com/jan98/art\_vb002.html

t = myRow.Item(i).ToString()

If Reg.IsMatch(t, "\x22|\x2C") Then

'\*\*\* field contains a quote x22 or a comma x2C. This means whole field must be enclosed in quotes

'\*\*\* additionally, the single quote x22 chars need to be modified to double quotes

t = Reg.Replace(t, "\x22", Chr(34) & Chr(34))

sOut.Append(Chr(34))

sOut.Append(t)

sOut.Append(Chr(34))

Else

sOut.Append(t)

End If

If i < objTable.Columns.Count() - 1 Then sOut.Append(",")

Next i

sOut.Append(vbCrLf)

# GENERATING AN XL WORKBOOK DUMP OF DATATABLES

This is useful for reports. Part of your workbook template contains table dump areas, another part runs a macro on-open to format the data for presentation. You generate the book on the server by copying from a template book, populating with this sub and then setting a hyperlink to point at the output. Note that you should put a macro in the book to copy-paste all cell values back onto themselves because the data is all output as TEXT with a leading apostrophe.

Sub writeToBook(ByVal XLbook As String, ByVal XLsht As String, ByVal myTable As DataTable)

'\*\*\*\* opens XLbook, writes to existing XLsht in that book, dumping myTable to that XLsht as a text table

'\*\*\* useful reference

'http://social.msdn.microsoft.com/Forums/en-US/adodotnetdataproviders/thread/822c9afb-3302-4810-86aa-1eb1ac46d000/

'https://blogs.msdn.com/spike/archive/2008/10/29/how-to-update-an-excel-worksheet-using-dataset-and-the-oledbdataadapter.aspx

'do NOT use commandbuilder for INSERT with XL sheets. It will never work. Instead you have to build your own Insert command

'do NOT use importRow because it does not set the Rowstate to new and therefore no records are appended by the .update action

'do USE itemarray to copy data from source row to a new row, then append this to your target table

'do USE .dispose to get rid of the connXL object in case of error, else the server will lock the spreadsheet

'note that it does not seem possible to create new worksheets; you can CREATE TABLE but only for Sheet1,Sheet2,Sheet3 if these are in the book

'already

**+++ WARNING 2017-12 Jet 4 no longer works. You will hit an error when trying to open an XL book. This is due to a Microsoft security patch. You can read/write to MS XML 2003 spreadsheets but these cannot support macros. +++**

Dim sXL As String = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & strUserFiles & XLbook & ";Extended Properties=""Excel 8.0;HDR=Yes;"""

Dim connXL As New OleDb.OleDbConnection(sXL)

Trace.Warn("write to book " & XLsht)

Try

Dim oDS As New DataSet

'\*\*\* test to create a new sheet

'syntax CREATE TABLE [Sheet1$] (ORIGIN text, BRAND text, MODEL text)

'also DROP TABLE [Sheet1$]

Dim strSQL As String = "CREATE TABLE [" & XLsht & "$] ("

For Each myC As DataColumn In myTable.Columns

'\*\*\* we dump all fields as TEXT type

strSQL += myC.ColumnName & " text,"

Next

'\*\*\* replace trailing comma with close bracket

strSQL = System.Text.RegularExpressions.Regex.Replace(strSQL, ",$", ");")

'\*\*\* now create blank table

connXL.Open()

Dim oCmd As New OleDb.OleDbCommand(strSQL, connXL)

oCmd.ExecuteNonQuery()

connXL.Close()

'\*\*\* now open this sheet

Dim myDA As New OleDb.OleDbDataAdapter("SELECT \* FROM [" & XLsht & "$] ", connXL) 'use connection object, not conn string

myDA.Fill(oDS, "XL")

'\*\*\* build the insert command

strSQL = "INSERT INTO [" & XLsht & "$] ("

Dim sParam As String = "("

For Each myC As DataColumn In myTable.Columns

strSQL += myC.ColumnName & ","

sParam += "? ,"

Next

'drop trailing commas

strSQL = System.Text.RegularExpressions.Regex.Replace(strSQL, ",$", ") VALUES ") & System.Text.RegularExpressions.Regex.Replace(sParam, ",$", ")")

myDA.InsertCommand = New OleDb.OleDbCommand(strSQL, connXL)

'\*\*\* now map the columnames to the params

For Each myC As DataColumn In myTable.Columns

'' myDA.InsertCommand.Parameters.Add("@" & myc.ColumnName, myc.DataType).SourceColumn = myc.ColumnName

myDA.InsertCommand.Parameters.Add("@" & myc.ColumnName, OleDb.OleDbType.VarChar, 255).SourceColumn = myc.ColumnName

Next

For Each myR As DataRow In myTable.Rows

'\*\*\* importRow doesnt set rowstate to INSERT so this doesn't get actioned

'\*\*\* so instead we need ADD. We also need to transfer data via itemarray to avoid

'\*\*\* row-belongs-to-another-table error

Dim newR As DataRow = oDS.Tables("XL").NewRow

newR.ItemArray = myR.ItemArray

oDS.Tables("XL").Rows.Add(newR)

Next

myDA.Update(oDS, "XL")

Catch ex As Exception

Trace.Warn(ex.ToString)

Finally

connXL.Dispose() ‘release lock on book

End Try

Trace.Warn("write to book ok")

End Sub

# DYNAMIC CONTROLS AT RUN TIME

You can build dynamic controls at run time and add these to a placeholder control on the page. But, you need to manually wire up the events using AddressOf, and you also need to do this in the page\_init event. I think page\_load might also work, but if you add the addressOf elsewhere then the event handler won’t work.

<http://forums.asp.net/t/391072.aspx>

plItm.Controls.Add(btnsRemove)

AddHandler btnsRemove.Click, AddressOf btnRemove\_click

Private Sub btnRemove\_click(sender As Object, e As System.Web.UI.ImageClickEventArgs)

End Sub

# SHOW BUSY GIF

IE has a bug, in that animated GIFs will freeze when you submit a page. Here’s a work around. Uses a timer to reload the gif on submit

function jvBusy(msg) {

var x=document.getElementById("statusBar")

x.innerHTML=msg

document.getElementById('divBusy').style.display ="";

setTimeout('document.images["imgLoading"].src="imgloading.gif"', 200);

}

<DIV id="divBusy" style="DISPLAY: none; Z-INDEX: 1; LEFT: 700px; POSITION: absolute; TOP: 200px"><IMG id="imgLoading" src="imgloading.gif"></DIV>

You then an a call to jvBusy() in the onClick attribute of the page button that invokes the lengthy processing.

# SELF CLOSING WINDOW

The problem with pop-up windows is that they can get lost amongst other windows and then won’t invoke and come to the front when user re-clicks the action that activated them. Solution is to make them self-close.

function handleBlur() {

//alert(event.clientX + " " + event.clientY);

if (event.clientY < 0 || event.clientY > document.body.clientHeight)

{ if (event.clientX < 0 || event.clientX > document.body.clientWidth){window.close()}

}

}

JO note: not quite correct, ifs should not be nested.

<body onblur="handleBlur()">

This routine will close the window if the click event happens outside its bounds

# DYNAMIC CLIENT SCRIPT

Useful if you want to pass params to a popup window and have this window read then self-refresh to process them. This saves putting the params in a URL which then sits in the user’s navigation history and causes potential data errors.

In the page <HEAD> section, include <asp:literal id=litClient runat=server>

<body onLoad=”jsSeed()”…

Then load the client script dynamically;

Dim strjscript As String = "<script language=""javascript"">"

strjscript &= "window.opener." & HttpContext.Current.Request.QueryString("formname") & ".value = '" & Format(Calendar1.SelectedDate, "yyyy-MM-dd") & "';window.close();"

strjscript = strjscript & "</script" & ">" 'Don't Ask, Tool Bug

litClient.Text = strjscript

Remember to use ; at end of every statement as they will be concatenated to a single line on the page. To disable the script, set litClient.text = “jsSeed(){};” on the postback

The above javacode actually works the other way around, the page returns values and the above code will write these back to window opener.

# DROPDOWN LISTS IN DATAGRID

These can be useful for ebay-style action options. But using autopostback is not a good idea because it won’t fire if you only have a single option item and you cannot cancel this event; resetting it to a 0th index ‘I want to..’ statement is ok to suppress the server processing but you will still hit a page serve.

A more elegant solution is to ItemDataBind an onClick and onChange client routine to it. OnClick will need to detect when there is just a single list item, and then it will call the onChange handler. The onChange handler needs to concatenate the text action descriptor with the x.parentNode.parentNode.rowIndex property. This is then put in a hidden field and Form1.submit() invoked.

The server checks the hidden value, subtracts 1 from the rowindex passed up and uses this to hook to the datagrid.itemindex value, thus you can find which datagrid row fired.

The client script can also intercept certain options and fire popup windows or confirm messages

I used this on privateline app

# REGISTER A CLIENT SCRIPT

Sometimes, it’s useful to emit a block of javascript that will run as soon as the client page loads. Note that you CANNOT declare global vars in such a script. These must exist on the client page already.

Session("idrequest") = pass target in session

Dim strScript As String = "<script language=JavaScript>"

strScript += "alert('Sorry, feeback not enabled');"

strScript += "window.open('PrivateLine\_Feedback.aspx','Feedback','scrollbars=1,width=1024,height=768,left=0,top=0');"

strScript += "</script>"

If (Not Page.IsStartupScriptRegistered("clientScript")) Then

Page.RegisterStartupScript("clientScript", strScript)

End If

This script will NOT persist between page serves

<http://www.4guysfromrolla.com/articles/021104-1.2.aspx>

‘\*\*\* not much help if you want it in the <HEAD> section, instead use a placeholder

<http://stackoverflow.com/questions/20083905/how-do-i-put-javascript-programmatically-into-head-block>

actually, just put a literal prior to </HEAD> save the page and then you can reference it and put the script in it.

Use RegisterArrayDeclaration("aRequestor", SB.ToString) if you want to generate a javascript array. SB would hold single quoted csv strings. No need to put in [] or a script string. Will not persist across page serves.

<http://msdn.microsoft.com/en-us/library/system.web.ui.clientscriptmanager.registerarraydeclaration(v=vs.110).aspx>

NOTE: the above are obsolete. Use ClientScript.RegisterStartupScript() instead. It will also add the script tags for you.

ClientScript.RegisterStartupScript(Me.GetType, "s1", "alert('hi');", True)

# FIRE EVENTS ON PARENT WINDOW FROM CHILD WINDOW

//http://www.java2s.com/Code/JavaScript/HTML/UsingthefireEventMethod.htm

try{

//these parent refs are still valid even if we repost the GL page back to the server

var prefix = parent.name.replace(/btnGLhelper/,"")

var pDoc = window.opener.document

//populate parent fields from child form

pDoc.getElementById(prefix + "view\_NRR").value=document.Form1.NRRUSD.value

pDoc.getElementById(prefix + "view\_MRR").value=document.Form1.MRRUSD.value

pDoc.getElementById(prefix + "view\_Option\_TB").value=document.Form1.AccessSpeed.value

pDoc.getElementById(prefix + "view\_GLcctID").value=document.Form1.txtCircuitID.value

//fire an onChange event on the target

var objID = prefix + "view\_MRR"

pDoc.all(objID).fireEvent("onChange")

}//try block

catch(e){

alert('Error writing fields back to IPL configuration')}

finally{

self.close()}

# SOAP CONNECTIONS

To connect to a SOAP service. Watch out, as you might get an error if your XML envelop refers to a schema that’s outside of the security domain. SOAP will return an XML doc which you can load into a dataset.

Sub HttpSOAPRequest()

rp\_GL.Visible = False

If txtCircuitID.Text = String.Empty Then

litMsg.Text = "Enter a numeric value for the GL circuit ID"

Exit Sub

ElseIf Not IsNumeric(txtCircuitID.Text) Then

litMsg.Text = "Enter a numeric value for the GL circuit ID"

Exit Sub

End If

'\*\*\* sends a SOAP request to the tardis server

'http://www.eggheadcafe.com/articles/20011103.asp

' http://dotnethelps.blogspot.com/2006/11/send-soap-request-request-with.html

Dim doc As System.Xml.XmlDocument = New System.Xml.XmlDocument

'\*\*\*create XML detail from text

Dim sXML As String = "<?xml version=""1.0"" encoding=""utf-8""?>" & \_

"<soap:Envelope xmlns:xsi=""http://www.w3.org/2001/XMLSchema-instance"" xmlns:xsd=""http://www.w3.org/2001/XMLSchema"" xmlns:soap=""http://schemas.xmlsoap.org/soap/envelope/"">" & \_

"<soap:Body>" & \_

"<getTardisCircuit xmlns=""http://tempuri.org/TardisCircuitService/Service1"">" & \_

"<circuitID>2479900</circuitID>" & \_

"</getTardisCircuit>" & \_

"</soap:Body>" & \_

"</soap:Envelope>"

sXML = sXML.Replace("2479900", txtCircuitID.Text)

doc.LoadXml(sXML)

Dim req As System.net.HttpWebRequest = System.Net.WebRequest.Create("http://ms-lon-glop02.emea.dsmain.com/tardiscircuitservice/tardiscircuitservice.asmx")

Trace.Warn("soap3")

'\*\*\* if SOAPAction header is required, add it here... not clear if you need quotes around the action

req.Headers.Add("SOAPAction", """http://tempuri.org/TardisCircuitService/Service1/getTardisCircuit""")

req.ContentType = "text/xml;charset=""utf-8"""

Trace.Warn("soap4")

req.Accept = "text/xml"

req.Method = "POST"

Dim stm As System.io.Stream = req.GetRequestStream()

doc.Save(stm)

stm.Close()

Dim resp As System.net.WebResponse = req.GetResponse()

stm = resp.GetResponseStream()

Dim r As System.IO.StreamReader = New System.IO.StreamReader(stm)

'\*\*\* process SOAP return doc here, using a dataset

Dim myDS As New DataSet

myDS.ReadXml(r)

Trace.Warn("tables=" & myDS.Tables.Count)

If myDS.Tables.Count <> 3 Then

r.Close()

stm.Close()

litMsg.Text = "No circuit details found. Make sure this is a valid Circuit ID (not GL project ID)"

dg\_GL.Visible = False

Exit Sub

End If

rp\_GL.DataSource = myDS.Tables(2)

rp\_GL.DataBind()

rp\_GL.Visible = True

litMsg.Text = "These circuit details found. Click USE-A or USE-USD as appropriate"

r.Close()

stm.Close()

Trace.Warn("soap5")

End Sub

# HTML ATTRIBUTES

You can add attributes to page objects on the server, and you can access these values on the client using javascript.

But, you cannot modify these on the client and expect them to be passed back to the server. The reason is the server will rebuild the page and bind all values in the viewstate, which means what you modded on the client is not passed back to the server. Its not even clear if these attributes would be present in the query string coming to the server.

# BASE64 ENCODE AND DECODE

Say you want to build a web service that reads POP3 email. For the consumer to read a file, we need the service to base64 encode it as text and then respond to the consumer  
http://www.nonhostile.com/page-vb-net-base64-encoding-decoding.asp

Public Function ToBase64(ByVal data() As Byte) As String

If data Is Nothing Then Throw New ArgumentNullException("data")

Return Convert.ToBase64String(data)

End Function

Public Function FromBase64(ByVal base64 As String) As Byte()

If base64 Is Nothing Then Throw New ArgumentNullException("base64")

Return Convert.FromBase64String(base64)

End Function

BTW, same thing in VB6 is http://www.nonhostile.com/howto-encode-decode-base64-vb6.asp

# FILE UPLOAD FROM WEB PAGE (SUPPORTS XLSX)

Dim myFile As HttpPostedFile = myH.PostedFile

Dim nFileLen As Long = myFile.ContentLength

Dim myData(nFileLen) As Byte

myFile.InputStream.Read(myData, 0, nFileLen)   
 '\*\*\* use System.IO to pull the filename out of the full file path

Dim strFilename As String = System.IO.Path.GetFileName(myFile.FileName)

‘\*\*\* It’s a good idea to strip illegal chars from filename and to test extensions for approved ones

Dim newFile As New System.IO.FileStream(strFilename, System.IO.FileMode.Create)

‘\*\*\* BUG, the postedfile len is actually 1 higher than required, so we ‘\*\*\* don’t add 1 to uBound. Seems .xls and .doc didn’t care about this ‘length problem

'\*\*\* http://sjc.ironspeed.com/post?id=3491565

newFile.Write(myData, 0, UBound(myData)) '\*\*\* 2010-04-19 magic fix, drop one byte

newFile.Close()

HIDING A WINDOW  
If you want system to do some data loading on a page and keep self-refreshing until its done, then a hidden popup is a good approach. Javascript prohibits hidden windows as a security measure. But you can do it as an iFrame in IE.

<iframe id="Download\_Frame" style="display:none; visibility:hidden;">

JunkMessage </iframe>

In the Javascript :

MyIFrame = document.getElementById("Download\_Frame");

MyIFrame.src = "MyUrl.php?params=" + Parameters;

# WORKING WITH DATAVIEW BACK TO DATASET

Sometimes you need to sort a dataset for processing and then save changes back to underlying table. This can be done by enumerating each datarowview in a dataview. Use drv.EndEdit, see http://www.dotnetmonster.com/Uwe/Forum.aspx/dotnet-ado-net/20831/Changes-to-DataSet-not-sticking

Dim myView As DataView = retDS.Tables("COSTNODE").DefaultView

myView.Sort = "isDefault DESC, NodeSearchName ASC" '\*\*\* puts the default item first in the list

Dim viewEnumerator As IEnumerator = myView.GetEnumerator()

While (viewEnumerator.MoveNext())

Dim drv As DataRowView = viewEnumerator.Current()

drv("vNode") = True ‘\*\*\* make a change to field vNode

drv.EndEdit() '\*\*\* need this to make the change stick!!

End While

retDS.Tables("COSTNODE").AcceptChanges() ‘\*\*\*update the dataset

NOTE: for intelligent search algorithms you are better off not sorting the dataset. Add a ‘points’ field and increment this for each pattern match. At end of this process, sort with a view to bubble the highest points to the top, then simply modify the first drv record to indicate your highest-points match.

# GENERATING EXCEL REPORTS

You have a number of options;

Place a button “copy friendly” on the page and have this hide all the button columns and controls in the datagrid (i.e. boundcols or literals only) user can then copy-paste to excel. Advantage is you re-use all the datagrid formatting. See <http://www.cflex.net/showFileDetails.cfm?ObjectID=298> as this can paste direct to XL

Or, copy the dataset/dataview for the grid to the session object and have a javascript button open a new window whose page references the dataset to populate a datagrid. The page needs to set

Response.ContentType = "application/vnd.ms-excel"

Response.Charset = ""

Or, copy the datagrid itself to the session object and have a javascript button open a new window who’s page references the datagrid and then renders it to html and outputs to page. This is complex and hits problems if the datagrid contains controls.

'http://www.tek-tips.com/faqs.cfm?fid=4893

'http://forums.aspfree.com/html-javascript-and-css-help-7/ctrl-c-keycode-is-47733.html

Dim tw As New System.IO.StringWriter

Dim hw As New System.Web.UI.HtmlTextWriter(tw)

Dim dgGrid As New DataGrid

'\*\*\* 2007-12-07 disabled this code. There's no point trying to format the doubles to 4dp because

'\*\*\* Excel will strip trailing zeros

'AddHandler dggrid.ItemDataBound, AddressOf dynamicGrid\_ItemDatabound

dgGrid.DataSource = objDS.Tables(0)

dgGrid.DataBind()

dgGrid.HeaderStyle.BackColor = System.Drawing.Color.Yellow

' Get the HTML for the control.

dgGrid.RenderControl(hw)

' Write the HTML back to the browser.

Response.Write(tw.ToString())

Response.End()

# READING EXCEL WITH OLEDB (MORE RELIABLE)

It seems that the ODBC driver, even with IMEX=1 gets confused by text entries and will return null instead of returning a numeric as a text value. Oledb copes better with this. Code below also demonstrates how to find all sheet names in a workbook.

'\*\*\* open ole db text driver http://www.connectionstrings.com/excel

Dim oConn As New OleDb.OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\pricing\eCosttest\irish.xls;Extended Properties=""Excel 8.0;HDR=No;IMEX=1"";")

oConn.Open()

'\*\*\* need to find the name of the eCost sheet in the workbook, the code comes from MSDN help

'http://stackoverflow.com/questions/1164698/using-excel-oledb-to-get-sheet-names-in-sheet-order

Dim myT As DataTable = oConn.GetOleDbSchemaTable(OleDb.OleDbSchemaGuid.Tables, \_

New Object() {Nothing, Nothing, Nothing, "TABLE"})

Dim target As String = myT.Rows(0).Item("TABLE\_NAME")

oConn.Close()

‘\*\*\* note that table target has to be referenced with a trailing $,   
e.g. sheet1$ the data returned from “TABLE\_NAME” has $ embedded

Dim oDS As New DataSet

Dim oDA As New OleDb.OleDbDataAdapter("SELECT \* FROM [" & target & "]", oConn)

oDA.Fill(oDS)

# A FEW DESIGN TIPS FOR WEB DATABASE APPS

Session variables. These will time out. If you want to avoid forcing the user to login again, then store a key session var such as NTID in a temporary cookie. If session times out you can use this to look up other session vars such as email address and profile/authority by querying the db and restoring the session vars.

Viewstate. Useful for persisting app state, and you can load custom variables to it, however you cannot pass these vars between pages using server.transfer. Instead, during the transfer routine you will need to copy the vars to Context, then pick them up on the receiving page and put them into the viewstate there.

AutoPostback. Useful for a small number of fields, but if you have a 20 field form it becomes annoying. You could put a javascript timer on the change event, so postback is delayed. Alternatively write the field values to a temp cache (hard work). Also its worth having a onePass flag in page load. Its possible that several events will fire on multiple controls during a single post back and you don’t want to process the save event multiple times. Set the flag on page load, and have your save routine only run if flag is set, clearing it after its first pass.

App level record locks. To prevent multiple users concurrently editing the same record, use an ISOdate recLock field on the db table. First user to save wins…

Datarow bind and unbind. I wrote these functions into a library. They now accept custom attributes associated with the control to determine their behaviour.

# JAVASCRIPT WINDOW EVENTS

See <http://www.quirksmode.org/js/events_properties.html>

Example, but this in the Javascript in <HEAD>

Window.onClick=”myeventHandler()”

Function myeventHandler(){//do something }

# SSL TIPS

You may see pop-ups ‘this site contains secure and non secure items’

This is often caused by links to non-https sites.

VS sometimes inserts <base href=<http://localhost>> in the <HEAD> this is intended to tell IE where to start looking for relative links. DELETE IT.

Iframes with src=”” are another problem. Have them load with SSLdummy.htm

For more tips look at

<http://www.sslshopper.com/article-stop-the-page-contains-secure-and-nonsecure-items-warning.html>

<http://info.ssl.com/article.aspx?id=10058>

# RELATIONAL DATASETS AND BINDING TO A DATAGRID

You can relate two datatables in a parent-child relation. These have to be full joins, you cannot do a left or right join. People programmatically build left join datasets from the results, this is quite involved. My tip is that if you wish to context.cache one of the tables, you should run a full SQL query, inner joins and all, then copy the desired fields to the cache including nulls so you can later call up the cache and join the cache data back to a simplified live dataset.

Dim oconn As New OleDb.OleDbConnection(strConn)

Dim oda As New OleDb.OleDbDataAdapter("SELECT \* FROM Table1", oconn)

Dim ods As New DataSet

oda.Fill(ods, "T1")

oda = New OleDb.OleDbDataAdapter("SELECT \* FROM Table2", oconn)

oda.Fill(ods, "T2")

ods.Relations.Add("thing", ods.Tables("T1").Columns("ID"), ods.Tables("T2").Columns("IDparent"))

'and bind T1 to the datagrid

In the datagrid, you use

<%#Container.Dataitem.Row.GetChildRows("thing")(0)("field2")%>

Container.dataitem.row is the row of data being bound, we are looking for childrows, of which there may be multiple, hence (0) the first and then index the target field

Or

<%#Container.Dataitem.Row.GetParentRow("thing")("Field1")%>

If you bound the child table. Only one parent row will exist hence no need for (0).

Watch out!

You can set a rowfilter across both parent and child, but it can only use aggregate functions on child. So

ods.Tables("T1").DefaultView.RowFilter = "MAX(Child.field2)='clean'"

does actually work.

BUT the biggest gotcha is that there needs to be at least a 1:1 relationship between child and parent. So if you ‘split’ some columns from a query to cache them, and use say the original record ID as the key, that's fine until you filter the dynamic dataset prior to making the relational link. This is because it now returns fewer records than the child dataset and you cannot have orphaned children. They must all have parents. This is why it's a pain that you can’t do LEFT joins.

All filtering therefore must take place after you have set up the relation.

<http://www.vinull.com/Post/2007/04/20/aspnet-have-a-meaningful-relationship-w.aspx>

<http://weblogs.asp.net/rajbk/archive/2004/07/20/what-s-the-deal-with-databinder-eval-and-container-dataitem.aspx>

# CONTEXT CACHE OF DATA

You can use context.cache to hold data at an app level, useful on webservices.

If Context.Cache.Item("cacheDS") Is Nothing Then

'\*\*\* Build dataset from database

'\*\*\* Hold in cache for 10 mins

Context.Cache.Insert("cacheDS", cacheDS, Nothing, DateAdd(DateInterval.Minute, 10), Now()), System.Web.Caching.Cache.NoSlidingExpiration)

Else

'\*\*\* Note that you need to reset cached tables defaultview to “” as it seems that playing with RowFilter in your app can change this.

Context.Cache.Item("cacheDS").Tables("qrySearchNode").defaultview.rowfilter = String.Empty

End If

And remember to dispose of connections. Datarowview should be set to nothing as should dataadatptors.

# RANDOM NUMBERS

Random number generation is straighforward

Dim RandomGenerator As Random

RandomGenerator = New Random

y = RandomGenerator.Next(100, 500) ‘generate a random between 100 and 500 inclusive

# STREAM FILES OFF THE SERVER

This also is a workaround for MIME types that have not been enabled on the server. It will stream any file to the client. Also has the advantage that IE will open a new window for the received file rather than navigating to it.

Private Sub StreamFile(ByVal sFullPath As String)

Dim tgByte() As Byte = Nothing

Try

Dim tgFStream As New IO.FileStream(sFullPath, IO.FileMode.Open, IO.FileAccess.Read)

Dim tgBinaryReader As New IO.BinaryReader(tgFStream)

tgByte = tgBinaryReader.ReadBytes(Convert.ToInt32(tgFStream.Length))

'\*\*\* write the response

Response.Clear()

Response.OutputStream.Write(tgByte, 0, tgByte.Length)

Response.AddHeader("Content-Disposition", "attachment; filename=" & System.IO.Path.GetFileName(tgFStream.Name))

Response.AddHeader("Content-Length", tgByte.Length.ToString())

Response.ContentType = "application/octet-stream"

Response.End()

tgBinaryReader.Close()

tgFStream.Close()

Catch ex As Exception

Response.Write(ex.ToString)

Finally

End Try

End Sub

# CONNECT TO TEXT FILES WITH OLEDB TEXT DRIVER

You can connect directly to a text file with the oleDB driver, which treats all fields as a text value and thus is easier to work with than the ODBC driver. You specify the directory in the connection string, and the file in the SELECT command.

'http://www.codeproject.com/Articles/27802/Using-OleDb-to-Import-Text-Files-tab-CSV-custom

Dim oConn As New OleDb.OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\inetpub\wwwroot\PCM3\;Extended Properties=""text;HDR=Yes;FMT=Delimited"";;")

oConn.Open()

Dim oDS As New DataSet

Dim oDA As New OleDb.OleDbDataAdapter

Dim oCmd As New OleDb.OleDbCommand("SELECT TOP 50 \* FROM [reportCSV.csv]", oConn)

oDA.SelectCommand = oCmd

oDA.Fill(oDS)

DataGrid1.DataSource = oDS

DataGrid1.DataBind()

# NET 2.0 CONVERSION

**SERVER.TRANSFER**

The first transfer to a new app will NOT load the web.config of the new app, therefore I had an issue migrating the login app to net 2. Subsequent postbacks on that app will load its net2 web.config file.

The error appears as ‘failure to execute child request’ or ‘object not intiailised’

# ADDING PARAMS TO A QUERY

‘\*\*\* you need to specify .value explicitly  
objCmd.Parameters.Add("@p1", OleDb.OleDbType.Date).Value = Now()

‘\*\*\* or you need to use .addWithValue

objCmd.Parameters.AddWithValue("@p3", hashedBytes)

‘\*\*\* you CANNOT just use .Add in net 2.0

objCmd.Parameters.AddWith("@p4", dropUser.SelectedItem.Value) ‘NO !

# CONNECTION STRING

Dim sConn As String = ConfigurationManager.ConnectionStrings("sConn").ConnectionString

‘\*\*\* web.config <configuration> area

<connectionStrings>

<add name="sConn" connectionString="PROVIDER=MICROSOFT.JET.OLEDB.4.0;DATA SOURCE=c:\inetpub\wwwroot\db\GUID.mdb;"/>

</connectionStrings>

# APP SETTINGS

ConfigurationManager.ConnectionStrings("strConn").ConnectionString

‘\*\*\* web.config <configuration> area

Dim s as string = System.Configuration.ConfigurationManager.AppSettings("MailServer")

<appSettings>

<add key="MailServer" value="smtp.ap.vzbi.com"/>

</appSettings>

**EMAIL**

Email is now handled by system.net.mail and it relies on the server being set in a specific <mailsettings> value in web.config

<system.net>

<mailSettings>

<smtp from="test@foo.com">

<!--network host="smtp.ap.vzbi.com" port="25" userName="username" password="secret" defaultCredentials="true" /-->

<network host="smtp.ap.vzbi.com" port="25" defaultCredentials="true" />

</smtp>

</mailSettings>

</system.net>

# VISUAL STUDIO 2008

When creating a new project do NOT check the create subdirectory option, else you end up with proj1 nested in proj1. It appears that the dev server runs with the project folder being the root, not wwwroot so the img file won’t work and nor will the login feature unless you fire up the IIS. Not sure there is a way to make the dev server run at the wwwroot level.

I cannot figure out a way to set the physical path of the development server in VS2008

<https://msmvps.com/blogs/paulomorgado/archive/2006/10/22/Using-the-ASP.NET-Development-Server.aspx>

Instead, you can fire it up independently from a desktop shortcut

One can find it by running task manager and inspecting the app properties. I created a shortcut with switches after the quotes

"C:\program files\Common Files\microsoft shared\DevServer\9.0\WebDev.WebServer.EXE" /port:8080 /path:"c:\inetpub\wwwroot

# DEFAULT BUTTON IN NET 2.0

Now it’s easy. Can set at form level, panel level or button level;

<http://forums.asp.net/t/985791.aspx?ASP+NET+2+0+Enter+Key+Default+Submit+Button>

<form id="form1" runat="server" defaultbutton="btn1">

<asp:Panel ID="Panel1" runat="server" defaultbutton="Button2">

# TREEVIEW IN NET 2.0

See this. Not sure if you can bind to a database

<http://quickstarts.asp.net/QuickStartv20/aspnet/doc/ctrlref/navigation/treeview.aspx>

# DATASETS AND GETINSERTCOMMAND

If you insert into a dataset using commandbuilder there is no easy way to retrieve the ID of the new record

<http://stackoverflow.com/questions/136536/possible-to-retrieve-identity-column-value-on-insert-using-sqlcommandbuilder-wi>

am thinking you might as well just call Scalar and get MAX of ID as long as you are using autonumber

# GET URL OF PAGE

<http://stackoverflow.com/questions/96029/get-url-of-asp-net-page-in-code-behind>

use this trick to handle apps in different dirs, https and server ports

newPWd = String.Concat(Request.Url.GetLeftPart(UriPartial.Authority), Request.FilePath, "?reset=",Server.UrlEncode(Convert.ToBase64String(.Item("Password"))))

# GRIDVIEW, DROPDOWNLIST SELECTEDINDEXCHANGED EVENT

If you put a DD list in a gridview, the event won’t be picked up by the gridview. Instead you need to define a generic dropdown list handler, point onSelectedIndexChanged to this, and use that routine to iterate all dds in the gridview to find the one with the matching clientID. BEWARE: do not rebind the gridview (e.g. on page load) too early or you will destroy the event and lose it.

Sub ddGeneric\_SelectedIndexChanged(ByVal sender As Object, ByVal e As EventArgs)

' http://www.codeproject.com/Articles/53559/Accessing-a-DropDownList-inside-a-GridView

Dim dd As DropDownList = sender

'\*\*\* find which one fired

For Each myR In gvConfig.Rows

dd = myR.findcontrol("ddAction")

If Not dd Is Nothing Then

If sender.clientid = dd.ClientID Then Trace.Warn(dd.ClientID) '\*\*\* this one

End If

Next

End Sub

# GRIDVIEW, FIND ROW SENDING COMMAND

For button columns (e.g. command Field) the gridview will automatically send the row index as the e.CommandArguement value. For other controls you need to mess around binding the index value somewhere.

’2021-06-06: for other template buttons, set CommandArguement thus

CommandArgument='<%# (TryCast(Container, GridViewRow)).RowIndex %>'/>

SIMPLER is this

Dim gvr As GridViewRow = TryCast(CType(e.CommandSource, Control).NamingContainer, GridViewRow) '\*\*\* cast to GridViewRow

If gvr Is Nothing Then

'\*\*\* gvr will be nothing if you click on sort, for example because the control will be the gridview and the namingcontainer the page. This happens if you are in the gridview rowCommand event, because standard events like sort, and button columns will bubble through this on their way to their own handlers

Trace.Warn("gvr nothing")

Exit Sub

End If

Trace.Warn("gvr =" & gvr.RowIndex)

‘\*\*\* work off the gvr, use .findcontrol etc. No need to set special command arguments on the controls

<https://www.aspsnippets.com/Articles/Using-Multiple-DataKeyNames-DataKeys-in-ASPNet-GridView-with-examples.aspx>

OLD This link claims a clever way to do it.. <http://stackoverflow.com/questions/13285918/row-index-in-gridview-rowcommand>

Another approach;

Dim I as int = gv.datakeys(parent.parent.mycontrol.rowindex)

Because generally the target control sits in a table cell, which sits in the gridview row so one can find the index from this

Private Sub gvFiles\_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) Handles gvFiles.RowCommand

Trace.Warn(e.CommandName)

Dim I As Long = e.CommandSource.parent.parent.rowindex

Trace.Warn(e.CommandSource.ToString())

Trace.Warn(I)

end Sub

Note: gridview will insist on using its own handlers for edit/update/cancel so don’t bother trying to write a combined handler that sits in the rowCommand handler.

This also works off the rowCommand

Dim index As Integer = Convert.ToInt32(e.CommandArgument)  
Dim gvRow As GridViewRow = gvEvent.Rows(index)  
  
Note2: The above works for a <asp:buttonCommand> generic buttons but these don’t have a CommandArguement parameter, so in effect the above returns their index position instead of a commandArguement value. If you want to pick up a Datakey index value instead, then you need to use a <template> field, put a button in it and set its commandArgument to an eval of the data index you need. You then need to use the parent.parent trick if you also want the gridviewRow index too.

Note3: This code below works for datagrids, so it should also work for gridviews. The control on the page (e.g. checkbox) needs to autopostback and make a client-side call to chkview\_checkedChanged

Sub chkview\_checkedChanged(ByVal sender As Object, ByVal e As EventArgs)

‘to use Namingcontainter you need to first cast to correct type

Dim dgr As DataGridItem = CType(sender, CheckBox).NamingContainer

Trace.Warn(dgr.ItemIndex)

And

<https://www.aspsnippets.com/Articles/ASP.Net-GridView---Get-Row-Index-on-RowCommand-and-Click-events.aspx>

also

<https://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.gridview.rowediting(v=vs.110).aspx?cs-save-lang=1&cs-lang=vb#code-snippet-1>

Update 2018-10-04

<https://stackoverflow.com/questions/6503339/get-row-index-on-asp-net-rowcommand-event>

Dim gvr As GridViewRow = CType(e.CommandSource, Control).NamingContainer ‘\*\*\* cast to generic control

‘\*\*\* etc  
oCmd.Parameters.AddWithValue("@p2", gvTaxSummary.DataKeys(gvr.RowIndex).Value)

NOTE: DataKeys(rowIndex)(sParam) is case sensitive, e.g. if Datakeys=”IDproj,someField” then (rowIndex)(“idproj”) will return null, and not the value in IDproj as hoped.

# GRIDVIEW ROW and COLSPAN

This works if you have a literal control within a <itemtemplate>, you can find the DataControlFieldCell, which is the parent of the literal, and work on this.

<Itemtemplate>

<asp:literal ID="fld\_toAccountName" Text='<%#Container.DataItem("toAccountName")%>' runat="server"/>

</Itemtemplate>

gvTrans\_RowDataBound(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewRowEventArgs)

If e.Row.RowType = DataControlRowType.DataRow Then

Dim lit As Literal = e.Row.FindControl("fld\_firstCellThing")

If Not lit Is Nothing Then

Dim dcfc As System.Web.UI.WebControls.DataControlFieldCell = lit.Parent

dcfc.ColumnSpan = 2

lit.Text = "a comment goes here"

Trace.Warn(lit.Parent.ToString) 'dcfc

Trace.Warn(lit.Parent.Parent.ToString) 'datagridrow

End If

'and need to hide or delete the next cell

lit = e.Row.FindControl("fld\_ secondCellThing")

dcfc = lit.Parent

dcfc.Visible = False 'hide it…or remove it

'e.Row.Controls.Remove(dcfc)

End If

This approach is useful if you want to modify some rows to show a comment for example

# AJAX TUTORIAL

<http://www.codeproject.com/Articles/401903/AJAX-for-Beginners-Part-Understanding-ASP-NET-AJ>

Allows partial page updating and posting of info back to server without a page refresh.

Not so easy… if you want to intercept a onClick event for a span, this allows you to submit the update panel, but onClick strangely does not fire if you select All the text (does if you are one char short).

Tried using onmousedown instead, but this event does not complete because it seems the ajax postback kills the event and clears the mouseselection.

Tried using a timer to not fire the ajax button for 5 sec, not reliable.

What I want is an async event sent to the server on mousedown that does not interfere with the click-select process. Mouseout is not reliable either because user might go outside of the text box when trying to select due to mouse-wobble.

Perhaps: mousedown just writes the id to a box, so multiple ones can be written, but the ajax call does not take place until say 20 sec later.

<http://encosia.com/why-aspnet-ajax-updatepanels-are-dangerous/>

try this. Use a JSON call, lightweight, we don’t want the panel updated anyway as it’s a one way call.

The asp.net controls are abstracted. To roll your own request, see

<http://www.w3schools.com/ajax/ajax_example.asp> VERY GOOD

this one is worth a read

<http://msdn.microsoft.com/en-us/magazine/cc163413.aspx>

Windows authentication  
In IIS7 you need to set the web app to disable anonymous auth and enable windows auth. The following asp code objects are used to test auth status;

User.Identity.IsAuthenticated resolves to true/false

User.Identity.Name is the auth\_user below

But we also have these Server Variables available to us

|  |  |
| --- | --- |
| AUTH\_TYPE | NTLM |
| AUTH\_USER | EMEA-DSMAIN\V817353 |
| AUTH\_PASSWORD |  |
| LOGON\_USER | EMEA-DSMAIN\v817353 |
| REMOTE\_USER | EMEA-DSMAIN\V817353 |

BUFFERING A QUERY RESULT  
If GUI allows user to run a search, we often want to remember the query parameters so we can offer a click to dump XL icon on the screen. Previously I did this with viewstate holding the entire Dataset but this is impractical for hundreds of rows. Instead we can buffer the SQL text command and its parameters so we can recreate it in the dump code.

<https://msdn.microsoft.com/en-us/library/system.data.oledb.oledbcommand.parameters%28v=vs.110%29.aspx>

When first running a TOP 100 query…

ViewState.Remove("dumpQ")

ViewState.Add("dumpQ", Regex.Replace(oCmd.CommandText, "TOP 100 ", String.Empty))

'\*\*\* ocmd.parameters is not serializable so instead lets build an arraylist. Supports objects, useful for dates

Dim ALP As New ArrayList

For Each p As OleDbParameter In oCmd.Parameters

ALP.Add(p.Value)

Next

ViewState.Remove("dumpP")

ViewState.Add("dumpP", ALP)

….and then dumping from the viewstate

Dim oCmd As New OleDb.OleDbCommand(ViewState("dumpQ").ToString, New OleDb.OleDbConnection(sConn))

'\*\*\* Add params individually from the arraylist

For Each p As Object In ViewState("dumpP")

oCmd.Parameters.Add(New OleDbParameter("@p1", p))

Next

Dim oDA As New OleDbDataAdapter(oCmd)

Dim oDS As New DataSet

oDA.Fill(oDS)

exportRFC4180table(Page, oDS.Tables(0))

oDA.Dispose()

BUFFERING A QUERY – ALTERNATIVE APPROACH  
Was there another way...

GLOBALISATION – OSSOWSKI.ORG   
Force the web-app to a particular culture.

<configuration>

<system.web>

<globalization uiCulture="en-AU" culture="en-AU" />

</system.web>

</configuration>

<http://stackoverflow.com/questions/17629225/change-the-currency-symbol-thoroughout-the-application>

PARSING CURRENCY   
Below is how to parse a currency. It will be culture sensitive. E.g. if culture is en-GB it will expect to see pound and not $. The below handles decimal notation.

'\*\*\* use the decimal parser on F3 to extract the value

newR("amount") = CDbl(Decimal.Parse(mydr("F3"), Globalization.NumberStyles.AllowCurrencySymbol + Globalization.NumberStyles.AllowDecimalPoint + Globalization.NumberStyles.AllowThousands))

IFRAME CONTROL AND MESSAGE PASSING  
For message of the day, I gave the MOTD page a style and that invoked my behavior.js library. To pass a message to a live iframe to re-query data, need something a bit more involved. But equally could just re-fresh the iframe with a querystring e.g. showDetail.aspx/1234

<http://robertnyman.com/html5/postMessage/postMessage.html>

can also render the content to a text literal on the page

<http://stackoverflow.com/questions/10907607/what-is-the-best-way-of-setting-the-src-of-an-iframe-from-a-codebehind-file-in-a>

# TAB STRIP CONTROLS

<http://www.codeproject.com/Articles/313197/Simple-way-to-Design-Tabs-in-ASP-NET>

There’s no native asp.net control, you can implement using buttons and styles. One for my javascript library?

# SIMPLE URL BEAUTIFIER for PathInfo

Useful to be able to hit a page showDetail.aspx/1234, and have page take 1234 as a query value and clean its URL to be showDetail.aspx

I updated checkNTLM in my library to automatically strip the path info and put it into Session(“PATHINFO”) and will then redirect to itself to clean the page.

To pick up path info (code below will pick up 1234 for example);   
If Session("PATHINFO") <> String.Empty Then

'drop leading \ and put rest into the txtSearch box

txtSearch.Text = Session("PATHINFO").ToString.Substring(1)

Session.Remove("PATHINFO")

SetFilter()

End If

# AJAX revisited for mouseover

You can create a <span onmouseover=”jtest(this);”> element that makes a js call to an .ashx handler which itself generates plain text responses. You can load this into the span.title and thus generate a dynamic mouseover.

function jtest(t){

var xhttp = new XMLHttpRequest();

xhttp.onreadystatechange = function() {

if (xhttp.readyState == 4 && xhttp.status == 200) {

t.title = xhttp.responseText;}

};

xhttp.open("GET", "Handler1.ashx", true);

xhttp.send();

}

//http://www.w3schools.com/xml/dom\_httprequest.asp  
its also possible to call a webservice that returns XML which you can load into DOM and then parse its nodes.

IFRAME SUB WINDOW

You can put an iframe on the page, initially not visible. A (link)button on the page can render it visible on next page serve, at which point it loads and is visible.

<iframe id="iNASP" runat="server" height="300px" width="500px" src="PB\_NaspSearch.aspx" visible="false" style="vertical-align: top"></iframe>

Then within the iframe page, perform some actions and pass a string back via parent.postMessage

Dim s As String = "onclick=parent.postMessage('" & gvNasp.DataKeys(e.CommandArgument).Value & "','\*');"

ClientScript.RegisterStartupScript(Me.GetType, "s1", s, True)

The parent form has a javascript event listener and can perform an action;

//add an event listener for the iFrame

Event.observe(window, "message", doNASP);

function doNASP(e) {

if ($("iNASP")) {

// e.data is the message content

$('txtNASP').value = e.data;

// $('iNASP').style.display = "none";

$("btnSearch").click();

}

}

Use display:none to hide the iframe in javascript, or trigger a parent page serve.

<https://davidwalsh.name/window-iframe>

<http://stackoverflow.com/questions/4249809/reload-an-iframe-with-jquery>

document.getElementById('cross\_domain\_page').contentWindow.postMessage(.......) in theory you can post to the iframe from the main window

MS-ACESS SET AUTO INCREMENT VALUE

One way is to create table (will want to start at 1), then create a second table with a long int field, set this to the start value, then append query it to the target table. Or you can write custom SQL code thus;

ALTER TABLE TableThatIncrements

ALTER COLUMN Id AUTOINCREMENT(1001,1)

You can set the step size. Above it is set to 1

Setting up page tabs

One easy way in html using command buttons

<http://www.codeproject.com/Articles/313197/Simple-way-to-Design-Tabs-in-ASP-NET>

Or you can do similar with the asp:menu and asp:multiview controls. The menu needs to be set to horizontal

<http://www.java2s.com/Code/ASP/Components/UseaspMenuandaspMultiViewtocreatetabcontrolC.htm>

A multiview is a bit more sophisticated than using multiple panel controls, because it can implement forward/back buttons

JAVASCRIPT WINDOW CLOSE WHEN LOST FOCUS

<http://forums.asp.net/t/1318318.aspx?window+open+window+losing+focus+to+button+postback>

For a calendar popup, if user clicks away from the popup window it will remain orphaned in the background.

One way to fix is with window.onblur event. Another is to have a 10 sec timer that will window.close() if window does not detect any click activity.

<script type="text/javascript">

window.onblur = function() {

window.close();

}

</script>

# USING TAB MENUS

<https://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.menu.renderingmode.aspx>

Beware that [www.ossowski.org](http://www.ossowski.org) runs Net 4.0 whereas the VES internal server runs Net 2.0 as does my work laptop. <asp: Menu> controls render differently in Net 3.5 onwards and don’t work for tab buttons unless you set RenderingMode="Table"

<asp:Menu id="Menu1" Orientation="Horizontal" RenderingMode="Table"

# CACHE CONTROL

<http://stackoverflow.com/questions/4480304/how-to-set-http-headers-for-cache-control>

<meta http-equiv="Cache-control" content="no-cache">

Alternatively the server must emit a random value on the target URL

Mypage.htm?random=1232141

# WHAT TO USE INSTEAD OF INCLUDE

<https://stackoverflow.com/questions/30434713/what-to-use-in-the-absence-of-include>

this is about classes – I was thinking of compiling a common.vb file or trying to use it as a library rather than copy-pasting to each project.

Btw you can compile several projects to individual bins and then put all these bins in the same application BIN folder. It will work. Allows me to create tools with one page and a binary behind and keep in a tools application folder. Or claim it’s all part of PCM4 for example.

# CUSTOM CONTROLS

<https://www.codeproject.com/articles/28783/your-first-asp-net-custom-control>

# CSV TEXT PARSER USING .NET CLASS

# Don’t roll your own. VB.NET has FileIO.TextFieldParser. It accepts a text file, but you can also use the StringReader class to feed it a stream. This way you do not need to save the incoming text file to the server. Neat. FileUploadVRD here is the fileupload control.

+++ library 2v8 contains an extension method fileToDatatable +++

Dim sR As StreamReader = New StreamReader(FileUploadVRD.FileContent)

Dim afile As FileIO.TextFieldParser = New FileIO.TextFieldParser(New StringReader(sR.ReadToEnd().ToString()))

sR.Dispose()

Dim CurrentRecord As String() ' this array will hold each line of data

afile.TextFieldType = FileIO.FieldType.Delimited

afile.Delimiters = New String() {","}

afile.HasFieldsEnclosedInQuotes = True

Dim dtI As New DataTable("myData")

Dim r As Long = 0

Dim c As Integer = 0

Dim dr As DataRow

Do While Not afile.EndOfData

Try

CurrentRecord = afile.ReadFields

c = 0

If r > 0 Then

dr = dtI.NewRow

End If

For Each s As String In CurrentRecord

If r = 0 Then

dtI.Columns.Add(s)

Else

'treat as data

dr.Item(c) = s

c += 1

End If

Next

If r > 0 Then dtI.Rows.Add(dr)

r += 1

Catch ex As FileIO.MalformedLineException

statusBar.InnerText = "ERROR the CSV file does not conform to RFC8140"

Exit Sub

End Try

Loop

afile.Dispose()

# It copes with escaped quotes, commas inside data, strings of commas denoting empty fields. The routine above is generic, in the sense it discovers the column headers in the first row of data.

<https://stackoverflow.com/questions/736629/parse-delimited-csv-in-net>

<https://stackoverflow.com/questions/7209484/how-to-read-data-via-streamreader-from-csv-file-as-it-is-means-it-should-come>

It also can accept a StringReader as its input, so you can parse a string from memory.

https://stackoverflow.com/questions/36209512/textfieldparser-parse-csv-from-string-Not-file

# DETECT SESSION TIMEOUT

Why do it? CP810 requires a page times out and redirects a user to SSO login. To me this is a waste of time because windows AD will still hold a valid, authenticated login if the browser is still open (as it caches the window login). For an SSO system, it will force you to enter your windows login again. For non SSO systems like mine, there’s no point asking for a password because I cannot check it (this is abstracted to AD). All I could do is force the user to close all browser windows in an effort to clear the cached pwd and sign in again. But this is illogical and impossible. The system could not tell the difference between a first-time SSO login and the system just picking up the Authenticated AD.

But wait a minute, it can… because it will challenge for an AD sign on if the session on the server has gone down.

So… I think I can check session vars, and if I don’t see any I can force a login… or can i? I can ask you to close all browser windows.

Nah. Won’t work. The only time the system asked for a windows login is first time you instantiate a browser.

[http://www.nullskull.com/a/1630/build-an-aspnet-session-timeout-redirect-control.aspx](https://urldefense.proofpoint.com/v2/url?u=http-3A__www.nullskull.com_a_1630_build-2Dan-2Daspnet-2Dsession-2Dtimeout-2Dredirect-2Dcontrol.aspx&d=DwMBaQ&c=udBTRvFvXC5Dhqg7UHpJlPps3mZ3LRxpb6__0PomBTQ&r=W2D1m0b0FYLRnNvpbd5iMJk1O5KTa99rOidH47scDo0&m=MUlY78JnYOlkDOJV2BJc4AjOPH4iWgxWzoGv6Vl6-oY&s=Zjh0Bhk9bNVOYgpDofbZK-eSMhn01zeewhElhB4qh_U&e=)

the ASP.NET HttpSessionState class's IsNewSession( ) method returns true if a new session was created for a given request. If this is a new session but the ASP.NET\_SessionId cookie is present, this indicates a timeout situation.  
<https://stackoverflow.com/questions/2874078/asp-net-session-sessionid-changes-between-requests>

Hmm. Right now, my systems ask the user to login once and that is prompted by AD. Even if the session vars timeout, they just go null. The user is not asked to login again. Is there a way to force this login?

<http://dotnetgallery.com/kb/resource6-Login-authentication-using-LDAP-Active-Directory-for-ASPNET-applications.aspx>

I don’t know the LDAP path.

Think again. Today, first time you open your browser and hit the site you need to authenticate. Thereafter you do not because browser keeps you authenticated. CP810 wants to force you to authenticate again after a timeout.

<https://stackoverflow.com/questions/1067263/asp-net-windows-authentication-logout>

yeah, I don’t think it can be done, as the article says;

windows authentication works at the IIS level by passing your Windows authentication token. Since authentication occurs at the IIS level you cannot actually log out from application code. However, there seems to be an answer to your problem here. It is the second question addressed and essentially involves using Forms Authentication and the LogonUser Windows api.

<https://visualstudiomagazine.com/articles/2004/05/01/activate-windows-impersonation-selectively.aspx>

so anyway, the way VES does this is apps are supposed to call out to SSO to authenticate you. SSO happens to use your windows login. SSO will inform the app you have signed in, and after a period the app will timeout and make you to go SSO again.

I don’t have integration with SSO because I am not sure its published anywhere in VES. I cannot force my IIS server to kick the user off. The only solution is to force all browser windows to close. This is a pain because if user has other apps open in their browser they will be forced to close all these too. It’s just no very helpful.

I also cannot take a trip to the LDAP server to check credentials because I don’t know where it is. I also don’t want users giving me their passwords.

A primitive approach is to set a cookie that only ‘goes away’ when user closes all browser windows.

<https://stackoverflow.com/questions/31326/is-there-a-browser-equivalent-to-ies-clearauthenticationcache/8497804#8497804>

so screen saver or not…. User walks away, app times out. CP810 wants user to login again. Cannot just do this by accepting the AD authentication token because the screensaver might never have activated.

Also have the issue that we are not aware of a user first logging in (i.e. using AD with a pwd) so we cannot differentiate between this and being passed a token from a compromised screen. E.g. if my app locks itself after 15 mins, all user needs do is close and open app and it picks up AD token.

So in short we need to move back to ye olde separate login form but really want to validate via SSO so we don’t have to keep separate pwds.

What a pain unless I can get an exemption.

Can I create a persistent object session or user cookie perhaps that hangs around until user closes their browswer?

Its worse with IE – cannot force it to log off! So custom form or SSO integration is the only way to go. Custom form is a pain in the ass because of the whole registration issue, and I saw it as a retrograde step forcing users to login again as we have a screensaver policy.

More notes.

Can you force a logout?

<https://stackoverflow.com/questions/31326/is-there-a-browser-equivalent-to-ies-clearauthenticationcache/8497804#8497804>

answer is not easily, forcing a 401 error does not clear the cache, your server has to pretend to accept some bogus login details

and then someone says you can do this via LDAP

<https://stackoverflow.com/questions/30861/authenticating-domain-users-with-system-directoryservices>

which is great except I don’t know my LDAP service paths. I also cannot initialize the directoryservices object. Hmm.

<https://stackoverflow.com/questions/10691001/the-type-or-namespace-name-directoryservices-does-not-exist-in-the-namespace>

need to add the assembley reference to your web.config

# ASP AJAX BUILT IN HANDLERS

There are some built-in handlers for AJAX call back.  
<http://www.thescarms.com/dotnet/ajax.aspx> is the roll your own.

You can also use a generic handler by adding this as a new item to your project.

<https://www.interfacett.com/blogs/getting-data-asp-net-applications-using-ajax-jquery-using-custom-httphandler/>

# CREATE SITES FOR MOBILE USE

Create sites for mobile apps, make use of Request.Browser object

<https://docs.microsoft.com/en-us/aspnet/whitepapers/add-mobile-pages-to-your-aspnet-web-forms-mvc-application>

# OLEdb AND DATAADAPTORS ADVANCED

To find the schema of Accessdb

Dim dConn As New OleDbConnection(sData)

Dim restrictions(3) As String

restrictions(3) = "TABLE"

'\*\*\* pull list of existing tables, there may be some temp tables in there

'\*\*\* https://stackoverflow.com/questions/1699897/retrieve-list-of-tables-in-ms-access-file

dConn.Open()

Dim dtTable As DataTable = dConn.GetSchema("tables", restrictions)

dConn.Close()

To add a new table, one quick way is via SELECT INTO from a template table, however you will need to set one column as a primary key if you want to later use update(). The quick way to do this is via a SQL command;

oCmd = New OleDbCommand("ALTER Table " & TheTable & " ADD PRIMARY KEY (ID);", dConn)

oCmd.ExecuteNonQuery()

When writing data, often we hit a field size issue. By default dataAdaptor does not pick up maxLen, you need to instruct it with oDA.MissingSchemaAction  
<https://social.msdn.microsoft.com/Forums/en-US/a0683bdb-8f77-4273-8388-15e8dd43db19/how-to-get-the-max-length-of-text-field-in-a-dataset?forum=adodotnetdataproviders>

Dim oDA As New OleDbDataAdapter("SELECT \* FROM tblSource WHERE ID=0", dConn)

oDA.MissingSchemaAction = MissingSchemaAction.AddWithKey

oDA.Fill(oDS, "target")

# DIRECT CAST (DirectCast)

To find the schema of Accessdb

Ipso locum

# THAT THINGY IN GRIDVIEW TO FIND FIRING ROW

To find the schema of Accessdb

Ipso locum

# IFRAMES IN DONET 4

To find the schema of Accessdb

Ipso locum

# ROWPSAN AND COLUMN SPAN IN GRIDVIEW

To find the schema of Accessdb

Ipso locum

# VB.NET EXTENTION METHODS

If you wish to extend an existing type or class, e.g. create String.Print() you can use an extension method.

I might find this useful for a directclass conversion on an arraylist.

Note also that List is useful. It has an add method and toArray.

<https://docs.microsoft.com/en-us/dotnet/visual-basic/programming-guide/language-features/procedures/extension-methods>

# GRIDVIEW AUTOGENERATED COLUMNS, SORTING AND GRIDVIEWROW

All template columns and boundcolumns will appear in the gv.datacolumns() collection as a DataControlField object. You can modify the headertext with UARR/DARR and this will persist in the viewstate.  
  
Autogenerated columns will all generate a sortExpression value that is the same as the column header name. These appear in the gv.HeaderRow.Cells collection as DataControlFieldHeaderCell. You can for-each these but need to test for hasControls, then cast control(0) to a LinkButton. The Text property on this LB is the headertext.

Dim myLb1 = TryCast(c.Controls(0), LinkButton)  
But be aware any change you make such as UARR/DARR is not persisted in viewstate and lost if you rebind the gridview.

For this reason, you are better off disabling autogenerate and instead programmatically adding extra BoundFields to the gv.Columns collection. These will persist including gv.databind() and so can be used for sorting with UARR/DARR.

For bound controls, clicking a control will raise a rowCommand event. If the control was a bound control such as buttonColumn, you can cast the GridViewCommandEventArgs e to a Int32 and this is the gridviewrow ref.  
IF the control was a templated control such as a link button, you must instead cast to a generic control first;

Dim gvr As GridViewRow = CType(e.CommandSource, Control).NamingContainer   
'\*\*\* cast to generic control

Trace.Warn(gvr.RowIndex)

# GRIDVIEW FILTERING ROW

Set up the gridview so first row is set of filter params

https://forums.asp.net/t/1746634.aspx?GridView+with+filtering+row+

# POSTBACK FUN

Set up

<https://code.msdn.microsoft.com/VBASPNETControlCausePostbac-75a0bace>  
Trace.Warn("eventtarget " & Page.Request.Params.Get("\_\_EVENTTARGET"))

note: if you dynamically change gridview items from text to say link buttons, then the dynamic control reference is returned such as

gvXtab$ctl11$ctl02, which you can match with regex to determine one of the dynamic controls was pressed

# FORCE REFRESH OF CSS AND JAVASCRIPT

Add a version key to web.config in AppSettings

< add key="VersionNumber" value="06032014"/> then on the page

<link href="styles/navigation.css?v=<%=ConfigurationManager.AppSettings("VersionNumber")%>" rel="stylesheet" type="text/css" />

And

<script src="\javascript\prototype.js?v=<%=ConfigurationManager.AppSettings("VersionNumber")%>" type="text/javascript"></script>

<script src="\javascript\behaviour.js?v=<%=ConfigurationManager.AppSettings("VersionNumber")%>" type="text/javascript"></script>

Now, when you wish to force a refresh of css and/or javascript you just change the version number in web.config

<https://stackoverflow.com/questions/2185872/force-browsers-to-get-latest-js-and-css-files-in-asp-net-application>

<https://stackoverflow.com/questions/4056827/how-to-read-web-config-app-key-settings-in-html-markup>

<https://stackoverflow.com/questions/7934160/asp-net-syntax>

# CORRECT HANDLING OF WHITESPACE CR/LF

If user can copy-paste to a control, give this the xss class to strip out html. Cr lf will be captured in the database, to render correctly apply the css class .preformatted {white-space: pre-line;} to that line.

# TREEVIEW; SOME THINGS TO CONSIDER

Treview control is a good way to explore datasets. When building the children, it is a good idea to add a VALUE that captures say SQL params and a command statement. This avoids the need to crawl back up the node structure on postback to find various hierarchy params. The value is also available on the client through the href value.

Example

Dim naspNode As New TreeNode With {

.Text = myR("fName") & " [" & myR("NASPID") & "]",

.Value = String.Concat("NASP=", myR("NASPID"), " CMD=NASP")

}

Example parser, shown as an extension method of string

<Extension> Function parseToDictionary(s As String) As Dictionary(Of String, String)

Dim mc As MatchCollection = Regex.Matches(s, "(\b\S+)=")

Dim d As New Dictionary(Of String, String)

'https://www.regular-expressions.info/lookaround.html

For i As Integer = 0 To mc.Count - 1

'need to look ahead to work out what part of substring to use

Dim sPos As Integer = mc(i).Index + mc(i).Length 'will point past the =

Dim ePos As Integer

'special case if i+1 is beyond end of collection

If i + 1 = mc.Count Then

d.Add(mc(i).Groups(1).ToString, s.Substring(sPos).Trim)

Else

ePos = mc(i + 1).Index

d.Add(mc(i).Groups(1).ToString, s.Substring(sPos, ePos - sPos).Trim)

End If

Next

Return d

End Function

And on the client

function OnTreeNodeSelected()

{

//event.srcElement is the node which triggered

//obj.getAttribute("href") will contain the node path and params (i.e. node.value)

var obj = event.srcElement; //this is the selected tree node

console.log(obj.getAttribute("href")); //needs to contain NASP to justify triggering a working message

}

Next line

# GRIDVIEW ADDING CELLS PROGRAMMATICALLY

If you have a variable dataset, e.g. a date series that might change its columns then you might want a mix of design-time columns and dynamically generated ones. There are a few things to watch out for;

1/ turn off viewstate for the datagrid. Else you will find when you remove date-columns and add them back, the design-time cols fail to render.

2/ Can easily add boundField columns and set their data values at the point you add them. You cannot dynamically bind them with <%%>

3/ you can also add templatecolumns but to add a control inside one such as a check box requires you to write your own iTemplate custom class handler. Note also that whilst these can generate postback event, you cannot hook them directly to a handler in your codebehind.

4/ to handle postbacks, add a javascript attribute that will populate a hidden field with a value. This happens just prior to the postback generated off the control. When handling the postback, you need to read the hidden field value and process it in the PageLoadComplete event, and NOT in the PageLoad event.

5/ use the RowDatabound event to local your new controls and bind them to the underlying Row.Dataitem(i) elements.

Public Class myTemplate

Implements ITemplate

Public Sub InstantiateIn(container As Control) Implements ITemplate.InstantiateIn

Dim ck As New CheckBox With {.Text = "select", .AutoPostBack = True}

'http://www.highoncoding.com/Articles/29\_Creating\_Datagrid\_columns\_programmatically.aspx

container.Controls.Add(ck)

'Throw New NotImplementedException()

End Sub

End Class

The class can be in your main codebehind file.

Example use (where lFields is a ListOf(String));

Dim tf As TemplateField

For Each f As String In lFields

tf = Nothing

If tf Is Nothing Then

'\*\*\* add it

'\*\*\* need this else UARR and DARR don't emit correctly

'.DataFormatString = "{0:c0}",

tf = New TemplateField With {.SortExpression = f, .HeaderText = f}

'\*\*\* add an itemtemplate

tf.ItemTemplate = New myTemplate

tf.FooterTemplate = New myTemplate

gvTimeMatrix.Columns.Add(tf)

End If

Next

# WORKING WITH COPY-PASTE FROM XL OR GOOGLE SHEET, SAVE AS XML

If you want to copy an entire XL or gsheet area, you can paste this to a multiline text box and process it. Code below parses it using a text parser and will also save part of it as a datatable converted to XML which you can then store in ACCESS as a memo text field and later re-render as a table

Dim oDA As New OleDb.OleDbDataAdapter("SELECT \* FROM qryTEST20200228", sConn)

Dim t As New DataTable("doggo")

oDA.Fill(t)

Dim sSQL As String = "Update tblRequest SET gSheetEntry=@p1 WHERE idrequest=297388"

Dim oConn As New OleDb.OleDbConnection(sConn)

Dim oCmd As New OleDb.OleDbCommand(sSQL, oConn)

'cannot easily write a dataset as a binary object to access

'instead render as xml and store this as a long string

Dim str As New MemoryStream()

t.WriteXml(str, True)

str.Seek(0, SeekOrigin.Begin)

With New StreamReader(str)

oCmd.Parameters.Add("@p1", oleDbType:=OleDb.OleDbType.Char).Value = .ReadToEnd()

End With

oConn.Open()

Response.Write(oCmd.ExecuteNonQuery())

oConn.Close()

and recovering

'read an xml string back to a table

Dim oConn As New OleDb.OleDbConnection(sConn)

Dim oCmd As New OleDb.OleDbCommand("SELECT gSheetEntry FROM tblRequest WHERE idrequest=297388", oConn)

oConn.Open()

Dim sData As String = oCmd.ExecuteScalar

oConn.Close()

'https://forums.asp.net/t/1146895.aspx?Xml+string+into+dataset

Dim ds As New DataSet

ds.ReadXml(New XmlTextReader(New StringReader(sData)))

gv1.DataSource = ds

gv1.DataBind()

Note: MSaccess does not support MEMO fields in UNION queries. It will truncate these to 255 chars and so mangle your XML.

# CONFIRM URL IS STILL VALID

I point users off to other content such as GDPR or product info. One can confirm this content is still valid by through one of these methods. Needs to make a call from the page, when user clicks the link so that the server is also triggered to test the link. Also want to put a limit on it, perhaps test once per day or month.

<https://stackoverflow.com/questions/924679/c-sharp-how-can-i-check-if-a-url-exists-is-valid>

public bool IsValidUrl(string url)

{

try

{

var request = WebRequest.Create(url);

request.Timeout = 5000;

request.Method = "HEAD";

using (var response = (HttpWebResponse)request.GetResponse())

{

response.Close();

return response.StatusCode == HttpStatusCode.OK;

}

}

catch (Exception exception)

{

return false;

}

}

You can also use methods such as GET. HEAD is good, as it does not pull the content, but it does require the far server to support it (some servers will return 404).

# SELECT TOP ROWS FROM A DATASET

Easy method using AsEnumerable. Net 3.5 and higher

'https://stackoverflow.com/questions/2787458/how-to-select-top-n-rows-from-a-datatable-dataview-in-asp-net/2787483

gv.DataSource = dt.AsEnumerable().Take(5).CopyToDataTable()

# ENUMERATE A DATAVIEW

Where myView is a DataView

For Each rowView As DataRowView In myView

‘do something

Next

# ALIGN LEFT AND RIGHT ON SAME ROW WITH CSS

Say you want a search bar with controls on left, and button off on the far right. <https://css-tricks.com/left-align-and-right-align-text-on-the-same-line/>

<div id="search">

<p class="alignleft">Text on the left.</p>

<p class="alignright">Text on the right.</p>

</div>

where

.alignleft {

float: left;

}

.alignright {

float: right;

}

And then after this, add another div to ‘reset’ thus

<div style="clear: both;"></div>

Easy.

NAMESPACE  
<https://docs.microsoft.com/en-us/dotnet/visual-basic/language-reference/statements/namespace-statement>

Overview of why use namespaces

READ/WRITE XLSX USING EPPLUS  
EPPlus is a NuGet plugin, version 4 is GNU, version 5 moves to a paid-license-for-commercial-use.

[https://github.com/EPPlusSoftware/EPPlus.Sample.NetFramework/blob/master/02-ReadWorkbook/ReadWorkbookSample.cs](https://github.com/EPPlusSoftware/EPPlus.Sample.NetFramework/blob/master/02-ReadWorkbook/ReadWorkbookSample.cshttps://dotnetcoretutorials.com/2019/12/09/reading-excel-files-in-net-core/)

[https://dotnetcoretutorials.com/2019/12/09/reading-excel-files-in-net-core/](https://github.com/EPPlusSoftware/EPPlus.Sample.NetFramework/blob/master/02-ReadWorkbook/ReadWorkbookSample.cshttps://dotnetcoretutorials.com/2019/12/09/reading-excel-files-in-net-core/)

Load it via NuGet.  
  
Imports System.IO

Imports OfficeOpenXml

Dim package As ExcelPackage = New ExcelPackage(fil.PostedFile.InputStream)

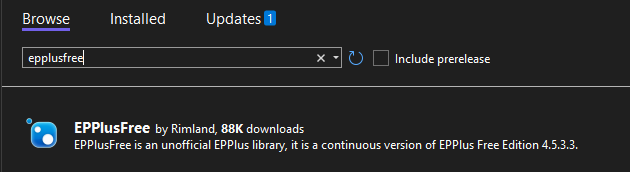
It can read from a file or from a stream. Or from streamReader.BaseStream  
<https://www.dart.com/pages/understanding-streams-in-net>

Dim worksheet As ExcelWorksheet = package.Workbook.Worksheets("Telarix Destinations")

You can refer to worksheet by ordinal or by name. Returns nothing if not present.

Worksheet.cells(r,c).value is Nothing if cell is blank, else it is a type such as string, double, date etc. Link below is useful and also covers ExcelRangeBase iteration

<https://pretagteam.com/question/can-epplus-distinguish-between-blank-cells-and-empty-text-cells-in-an-excel-worksheet>



# PARSE DATE INFORMATION FROM CSV FILE

Some information CSV sources have US m/d/y formats, others euro d/m/y. TryParseExact() will allow you to consistently handle these.

Dim d As Date

If Date.TryParseExact(myR("EFFECTIVE DATE"), "MM/dd/yyyy", System.Globalization.CultureInfo.InvariantCulture, System.Globalization.DateTimeStyles.None, d) Then

newR("effectiveDate") = d

Else

'\*\*\* tryParseExact, expects date in the defined format else it returns false

newR("effectiveDate") = DBNull.Value

End If  
  
'\*\*\* more info, you can also suggest multiple formats to tryParseExact

<https://stackoverflow.com/questions/47052779/parse-date-string-with-single-digit-day-e-g-1-11-2017-as-well-as-12-11-2017>

If you do not know if there will be 0 padding you can pass an array of acceptable formats, the parser will try each one in order they appear in the array.  
dt as DateTime = DateTime.ParseExact("1-11-2017", new string[]{"d-M-yyyy", "dd-MM-yyyy"}, System.Globalization.CultureInfo.InvariantCulture, System.Globalization.DateTimeStyles.None)

Note that if the day is zero padded, use dd, if not use d. same for M vs MM. Beware the source string may contain non printable chars, you need to strip these.

# PULL TAGS FROM PHOTO AND VIDEO FILES

Photo and video files contain meta-tags. For photos these are Exif tags. When dumping material from google photos, it generates a zip with the create and modified date =now() which is not ideal. So we’d want to use these tags to pull the actual date of the material. To do this, you need to import a COM library.

Imports System.ComponentModel

Imports System.IO

Imports System.Text.RegularExpressions

This code block will dump all the available tags on your specific OS

<https://stackoverflow.com/questions/62041605/updated-equivalent-code-for-using-shell-in-vb-net-with-folderitem-getdetailsof>

'Under COM import Microsoft Shell Controls And Automation

Dim objShell As Object

Dim objFolder As Object

objShell = New Shell32.Shell

objFolder = objShell.NameSpace(fi.DirectoryName) ’folder

’dump tags on this OS

Debug.WriteLine("The tags on your OS are")

Dim detail As Object

Dim j As Integer

For j = 0 To 400

detail = objFolder.GetDetailsOf(Nothing, j)

If Not detail Is Nothing Then

Debug.WriteLine("[" & j & "] " & detail.ToString)

End If

Next

’Under Win10 code208 is MediaCreated, 12 is DateTaken

Dim fi As New FileInfo(s) ’path to your file

Debug.WriteLine(fi.DirectoryName)

Debug.WriteLine(fi.Name)

Dim objShell = New Shell32.Shell

Dim objFolder = objShell.NameSpace(fi.DirectoryName) ’folder

If (Not objFolder Is Nothing) Then

Dim objFolderItem As Object

objFolderItem = objFolder.ParseName(fi.Name)

Dim dt As DateTime = Nothing

'\*\*\*don't need to worry about date formats because they are not stored as a culture specific format

'\*\*\*i.e. system will render them to itself in system culture and tryparse will work

If Not DateTime.TryParse(Regex.Replace(objFolder.getdetailsof(objFolderItem, 208), "[^ -~]+", ""), dt) Then

’Under Win10 code208 is MediaCreated,

If Not DateTime.TryParse(Regex.Replace(objFolder.getdetailsof(objFolderItem, 12), "[^ -~]+", ""), dt) Then

’code 12 is DateTaken

Debug.Writeline("fallback")

dt = fi.LastWriteTime ’default to using lastWritetime for non media files

End If

End If

Debug.WriteLine ("dt is " & dt)

Exit Sub

See also tryparseexact, which can accept multiple expected date formats

<https://stackoverflow.com/questions/11999912/datetime-tryparseexact-rejecting-valid-formats>

# ISO DATES REVISITED

Harder than you think

<https://code-maze.com/convert-datetime-to-iso-8601-string-csharp/>

Dim d As Date = myR("createdDate")

myR("createdDateISO") = Left(d.ToString("s"), 10)

cannot do on one line, myr(“createdDate”).toString(“s”) fails

# BINDING TO AN ARRAY

If you wish to use an array as a datasource, it does not have a fieldname, so instead you bind thus;

Dim test() As String = {"cat", "sat", "mat"}

And in the asp:gridview or asp:datalist use

<%# Container.DataItem %>

# CLASSES AND SERIALIZATION

If you wish serialize a class in order to store/retrieve in viewstate, this can only be done for the properties of the class, and these must be serializeable. E.g. if a property is a textbox, this is not serializeable. Listcollections are, listOf() is ok as are arraylists() and arraylists can be used to store multiple types of objects.

<Serializable()>

Class myThing

Private Property \_thisTbvalue As String

Private Property \_thisLC As ListItemCollection

Private Property \_thisCk As Boolean

Private \_thisDD As DropDownList

'https://learn.microsoft.com/en-us/dotnet/visual-basic/programming-guide/language-features/objects-and-classes/

'https://www.tutorialspoint.com/vb.net/vb.net\_classes\_objects.htm

Public Sub New()

'\*\*\* constructor, required if you wish to serialize

End Sub

Public Sub setMe(dd As DropDownList, tb As TextBox, ck As CheckBox)

If tb IsNot Nothing Then \_thisTbvalue = tb.Text

If dd IsNot Nothing Then \_thisLC = dd.Items : \_thisDD = dd

If ck IsNot Nothing Then \_thisCk = ck.Checked

End Sub

Public Sub fixMe(dd As DropDownList, tb As TextBox, ck As CheckBox)

'\*\*\* dd.items is read only, so have to rebuild

If dd IsNot Nothing Then

dd.Items.Clear()

For Each li In \_thisLC

dd.Items.Add(li)

Next

End If

If tb IsNot Nothing Then tb.Text = \_thisTbvalue

If ck IsNot Nothing Then ck.Checked = \_thisCk

End Sub

Public Function serializeMe() As String

Dim xml\_serializer As New XmlSerializer(GetType(myThing))

Dim string\_writer As New StringWriter

xml\_serializer.Serialize(string\_writer, Me)

Return string\_writer.ToString()

string\_writer.Close()

End Function

Public Sub deSerializeMe(s As String)

'reinstate object from a string

Dim xml\_serializer As New XmlSerializer(GetType(myThing))

Dim string\_reader As New StringReader(s)

Dim tempThing As myThing = DirectCast(xml\_serializer.Deserialize(string\_reader), myThing)

'\*\*\* copy over properties, you cannot assign objects because they are a reference, not a cloned object

\_thisTbvalue = tempThing.\_thisTbvalue

\_thisLC = tempThing.\_thisLC

tempThing = Nothing

string\_reader.Close()

End Sub

# CLASSES AND OJBECT CLONING

If your class captures an object, it does so as a reference. One simple way to clone a copy (to hold existing values for later use) is to load object into an Arraylist and then use .Clone method on the arraylist. To retrieve you need to copy the members/properties of the clone back to the current target object, you cannot assign.

Class headFilter

'has a reference to the headerrow. can ask it for the current value of a given control therein

'https://stackoverflow.com/questions/5299435/how-to-create-control-arrays-in-vb-net

'https://stackoverflow.com/questions/5555674/create-a-copy-of-an-asp-net-control-object

' https://stackoverflow.com/questions/78536/deep-cloning-objects

'we need to make a COPY of the controls in the headerrow

'\*\*\* not likely this class is serializable because it contains arraylists of non serializeable web controls

'\*\*\* can also use ListOf() for a typed list

Dim \_controlClone As New ArrayList 'will be adding controlClones

Public Sub New()

End Sub

''' <summary>

''' picks up a live reference to the HeaderRow

''' and clones the filter objects therein, thus snapshotting them

''' </summary>

''' <param name="gv"></param>

Sub setHR(gv As GridView)

Dim temp As New ArrayList 'holds references to existing headerrow objects

For Each fhc As DataControlFieldHeaderCell In gv.HeaderRow.Controls

For Each crl As Control In fhc.Controls

If TypeOf (crl) Is DropDownList Then

temp.Add((DirectCast(crl, DropDownList)))

ElseIf TypeOf (crl) Is TextBox Then

temp.Add((DirectCast(crl, TextBox)))

ElseIf TypeOf (crl) Is CheckBox Then

temp.Add((DirectCast(crl, CheckBox)))

End If

Next

Next

'clone the headerrow objects

\_controlClone = temp.Clone

End Sub

''' <summary>

''' restores the cloned controls with their values and selections to the header row

''' </summary>

''' <param name="gv"></param>

Sub restoreHR(gv As GridView, pg As Page)

For Each ctrlToRestore As Control In \_controlClone

Dim ctrlTemp As Control = gv.HeaderRow.FindControl(ctrlToRestore.ID)

If ctrlTemp IsNot Nothing Then

'\*\*\* you cannot assign, instead you must copy contents of cloned control back

If TypeOf (ctrlTemp) Is DropDownList Then

TryCast(ctrlTemp, DropDownList).Items.Clear()

For Each li As ListItem In TryCast(ctrlToRestore, DropDownList).Items

TryCast(ctrlTemp, DropDownList).Items.Add(li)

Next

ElseIf TypeOf (ctrlTemp) Is TextBox Then

TryCast(ctrlTemp, TextBox).Text = TryCast(ctrlToRestore, TextBox).Text

ElseIf TypeOf (ctrlTemp) Is CheckBox Then

TryCast(ctrlTemp, CheckBox).Checked = TryCast(ctrlToRestore, CheckBox).Checked

End If

End If

Next

End Sub

Above is a partial code snippet, part of a gridview filter in the HeaderRow, you can pass in the current gv and enumerate objects in the HeaderRow DataControlFieldHeaderCell to find pairs of textbox-dropdowns which work together in the filter. The control ID will be common across the live and the cloned control.

# ARRAYS, LIST(OF…) AND SORTING

List(Of String) is a convenient way to capture an array of strings. List(Of…) is typed, whereas ArrayList can contain multiple types.

If you have a list of strings, by default this will sort alpha, but you can force a numeric sort using Icomparer. Cols is list(of string)

'https://stackoverflow.com/questions/41423646/how-to-sort-an-arraylist-if-it-is-numeric-or-alphabetical

cols.Sort(Comparer(Of String).Create(Function(s1, s2) Int32.Parse(s1).CompareTo(Int32.Parse(s2))))

'\*\*\* note the above only works if the strings truly can be converted to ints, which in our case they can because we used isNumeric earlier