DataMode Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproDataModeC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDataModeX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDataModeA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDataModeS"}

Sets a value specifying whether the **DBGrid** operates in bound or unbound mode. This property is not available at run time.

Remarks

The **DataMode** property may be set to one of the following values:

Constant	Value	Description
dbgBound	0	The **DBGrid** displays data available from the bound data source.
dbgUnbound	1	The **DBGrid** uses the unbound control events to retrieve and update displayed data.

Note The above constants may only be used for comparisons because the property is not available at run time.

The **DataMode** property controls how the data is handled for the **DBGrid** control. In unbound mode, you are responsible for maintaining data and supplying the **DBGrid** control with the appropriate data when requested through the unbound events. In bound mode, the data is retrieved and updated automatically using the **Data** control's **Recordset**.

Normally, the unbound mode of the **DBGrid** control is used when displaying data that is not stored in a database accessible by the Microsoft Jet Database Engine. You can use the unbound mode for whatever type of data you have available. For example, you can use the unbound mode of the **DBGrid** control to display data from a proprietary database format or use it to manage data that you keep track of in a text file.

Note When the **DBGrid** is bound to a data source, setting the **DataMode** to 1 results in an error.

UnboundAddData Event

Occurs in an unbound **DBGrid** control when a new row is added to it. This event alerts your application that it must add a new row of data to its data set.

Syntax

Private Sub *object*_**UnboundAddData(***rowbuf* **As RowBuffer,** *newrowbookmark* **As Variant)**

The UnboundAddData event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
rowbuf	The **RowBuffer** object which contains the retrieved row of data.
newrowbookmark	A bookmark which acts as a unique identifier for each row of data.

Remarks

The number of rows to be retrieved is determined by the **RowCount** property, which is always set to a value of 1 in this event, because you can add only one row at a time.

A new row of data can be added to the data set only if the **AllowAddNew** property is **True**.

The **RowBuffer** object may not be fully populated, but contains entries only for those cells that were modified. Therefore check the **Value** property of the **RowBuffer** object for **Null** for each column. For **Null** valued columns, you can assign the **DefaultValue** property of the **Column** object to your data set.

To cancel the UnboundAddData event, set the **RowCount** property of the **RowBuffer** object to 0. This signals the **DBGrid** that the add attempt failed. By doing this, you can gracefully handle data conversion and insufficient permission errors.

The *rowbuf* argument contains a single row of data to be written to the data set. Before returning from this event, *newrowbookmark* must be set to the bookmark of the newly added row.

UnboundAddData Event Example

This example illustrates the use of the UnboundAddData event to add a row of data to a database, in this case, a simple array.

```
Private Sub DBGrid1_UnboundAddData(ByVal RowBuf As RowBuffer,
NewRowBookmark As Variant)
Dim Col%

mTotalRows = mTotalRows + 1
ReDim Preserve UserData(MAXCOLS - 1, mTotalRows - 1)
'Sets the bookmark to the last row.
NewRowBookmark = mTotalRows - 1

' The following loop adds a new record to the database.
For Col% = 0 To UBound(UserData, 1)
    If Not IsNull(RowBuf.Value(0, Col%)) Then
        UserData(Col%, mTotalRows - 1) = RowBuf.Value(0, Col%)
    Else
        ' If no value set for column, then use the
        ' DefaultValue
        UserData(Col%, mTotalRows - 1) = DBGrid1.Columns(Col%).DefaultValue
    End If
Next Col%
End Sub
```

UnboundDeleteRow Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtUnboundDeleteRowC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtUnboundDeleteRowX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtUnboundDeleteRowA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtUnboundDeleteRowS"}

Occurs whenever a row of data is deleted from the unbound **DBGrid**. This event alerts your application that it must delete a row of data from its data set.

Syntax

Private Sub *object*_**UnboundDeleteRow(***bookmark* **As Variant)**

The UnboundDeleteRow event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
bookmark	A value representing the bookmark of the row to be deleted.

Remarks

The *bookmark* argument must be set to the bookmark value provided when the row was retrieved with the UnboundReadData event or added by the UnboundAddData event.

To cancel the UnboundDeleteRow event, set the *bookmark* parameter to **Null** which will prevent the row from being deleted.

UnboundDeleteRow Event Example

This example illustrates the use of the UnboundDeleteRow event to delete a row of data from a database, in this case, a simple array.

```
Private Sub DBGrid1_UnboundDeleteRow(Bookmark As Variant)
Dim Col%, Row%

' Move all rows above the deleted row down in the
' array.
For Row% = Bookmark + 1 To mTotalRows - 1
    For Col% = 0 To MAXCOLS - 1
        UserData(Col%, Row% - 1) = UserData(Col%, Row%)
    Next Col%
Next Row%

mTotalRows = mTotalRows - 1
End Sub
```

UnboundReadData Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtUnboundReadDataC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtUnboundReadDataX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtUnboundReadDataA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtUnboundReadDataS"}

Occurs whenever an unbound **DBGrid** control requires data for display, such as when you scroll the **DBGrid** display.

Syntax

Private Sub *object*_**UnboundReadData(***rowbuf* **As RowBuffer,** *startlocation* **As Variant,** *readpriorrows* **As Boolean)**

The UnboundReadData event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
rowbuf	The **RowBuffer** object that will contain the retrieved data. When entering this event, the **RowCount** property contains the number of rows to be retrieved.
startlocation	A variant bookmark which specifies the row to position to before fetching the next or previous set of records. If *readpriorrows* is **False** then *startlocation* specifies the bookmark before the first record to be read. If *readpriorrows* is **True** then *startlocation* specifies the bookmark after the first record to be read. If *startlocation* is null, then fetching occurs at the first record or the last record, depending on the *readpriorrows* argument.
readpriorrows	**True** if the grid is requesting rows before the *startlocation*. **False** if the grid is requesting rows after the *startlocation*.

Remarks

Once the data is retrieved, it is stored in the **DBGrid** control's cache.

It is not necessary to fill the row buffer completely, and it is in fact acceptable to return no rows at all. The value of the **RowCount** property of the **RowBuffer** object can be set to indicate that fewer rows were returned than requested. The **DBGrid** interprets this to mean that there are no more rows to retrieve in the indicated direction. Thus, it is only necessary to fill the row buffer completely if there are more valid rows to be retrieved. You can inform **DBGrid** of BOF/EOF and error conditions by setting the **RowBuffer** object's **RowCount** property to 0.

For each row retrieved, you must provide a bookmark, which can later be used to refer to that row. This bookmark is specified by setting the **Bookmark** property of the **RowBuffer** object for each row returned. The data itself is specified by setting the **Value** property of the **RowBuffer** object for each column of data in each row returned.

UnboundReadData Event Example

This example illustrates the use of the UnboundReadData event to read a row of data from a database, in this case, a simple array.

```
Private Sub DBGrid1_UnboundReadData(ByVal RowBuf As RowBuffer,
StartLocation As Variant, ByVal ReadPriorRows As Boolean)
Dim CurRow&, Row%, Col%, RowsFetched%, Incr%

If ReadPriorRows Then
    Incr% = -1
Else
    Incr% = 1
End If

' If StartLocation is Null then start reading at the
' end or beginning of the data set.
If IsNull(StartLocation) Then
    If ReadPriorRows Then
        CurRow& = RowBuf.RowCount - 1
    Else
        CurRow& = 0
    End If
Else
    ' Find the position to start reading based on the
    ' StartLocation bookmark and the lngIncr% variable
    CurRow& = CLng(StartLocation) + Incr%
End If

' Transfer data from our data set array to the RowBuf
' object which DBGrid uses to display the data
For Row% = 0 To RowBuf.RowCount - 1
    If CurRow& < 0 Or CurRow& >= mTotalRows& Then Exit For
    For Col% = 0 To UBound(UserData, 1)
        RowBuf.Value(Row%, Col%) = UserData(Col%, CurRow&)
    Next Col%
    ' Set bookmark using CurRow& which is also our
    ' array index
    RowBuf.Bookmark(Row%) = CStr(CurRow&)
    CurRow& = CurRow& + Incr%
    RowsFetched% = RowsFetched% + 1
Next Row%
RowBuf.RowCount = RowsFetched%
End Sub
```

UnboundWriteData Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtUnboundWriteDataC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtUnboundWriteDataX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtUnboundWriteDataA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtUnboundWriteDataS"}

Occurs when an unbound **DBGrid** control has an entire row of modified data to write to the data set. It alerts your application that it must update an edited existing row of data to its dataset.

Syntax

Private Sub *object*_**UnboundWriteData**(*rowbuf* **As RowBuffer**, *writelocation* **As Variant**)

The UnboundWriteData event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
rowbuf	The **RowBuffer** object that contains the modified row of data.
writelocation	A value that identifies the unique bookmark of the row of data as specified in the UnboundReadData and UnboundWriteData events.

Remarks

The value of the **RowCount** property of the **RowBuffer** object will always be 1, because you can update only one row of data at a time.

The **RowBuffer** object may not be fully populated, but contains entries only for those cells that were modified. Therefore, check the **Value** property of the **RowBuffer** object for **Null** for each column.

To cancel the UnboundWriteData event, set the **RowCount** property of the **RowBuffer** object to 0. This signals the **DBGrid** that the write attempt failed. By doing this, you can gracefully handle data conversion and insufficient permission errors.

UnboundWriteData Event Example

This example illustrates the use of the UnboundReadData event to read a row of data from a database, in this case, a simple array.

```
Private Sub DBGrid1_UnboundWriteData(ByVal RowBuf As RowBuffer,
WriteLocation As Variant)
Dim Col%

' Update each column in the data set array
For Col% = 0 To MAXCOLS - 1
    ' Only columns that have been changed will be
    ' updated. Otherwise, the value will be set to NULL
    If Not IsNull(RowBuf.Value(0, Col%)) Then
        UserData(Col%, WriteLocation) = RowBuf.Value(0, Col%)
    End If
Next Col%
End Sub
```

RowBuffer Object

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbobjRowBufferC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbobjRowBufferX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"vbobjRowBufferP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"vbobjRowBufferM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"vbobjRowBufferE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbobjRowBufferS"}

The **RowBuffer** object is used to contain one or more rows of retrieved data and transfer it between your application and an unbound **DBGrid** control.

Note The **RowBuffer** object is not a derivable object which can be used by other applications. It is an internal class usable only by unbound **DBGrid** controls.

Syntax

DBGrid.**RowBuffer**

Remarks

A **RowBuffer** is used in the following circumstances:

- The **RowBuffer** object is passed as an argument to the UnboundReadData event, where one or more rows of data are placed in the **RowBuffer** object, which the **DBGrid** uses to fill its cache with data.
- When data is updated in the **DBGrid**, the **RowBuffer** object is passed to the UnboundWriteData event. Only one row of data is updated at a time, and the row updated is supplied with the original bookmark provided when the row was retrieved.
- When a new row is added, the newly-added data is placed in the **RowBuffer** object and passed to the UnboundAddData event.

In all of the these cases, the **RowBuffer** object is passed to you by the unbound **DBGrid** control. It is incumbent upon you and your application to add this data to your data set (when passed by the UnboundReadData event) or update existing data in your dataset (when passed by the UnboundWriteData, UnboundAddData, or UnboundDeleteRow events).

Bookmark Property (DBGrid)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBookmarkDBGridC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproBookmarkDBGridX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproBookmarkDBGridA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproBookmarkDBGridS"}

Returns or sets a bookmark for the specified row within a **RowBuffer** object in an unbound **DBGrid** control.

Syntax

object.**Bookmark** (*row*) [= *value*]

The **Bookmark** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
row	An integer specifying the row where the bookmark is placed. The range of this value can be from 0 to **RowCount** – 1.
value	A variant representing the bookmark for the specified *row*.

Remarks

Use the value returned by the **Bookmark** property to save a reference to the current row that remains valid even after another row becomes current.

When you set the **Bookmark** property to a valid value in code, the row associated with that value becomes the current row, and the grid adjusts its display to bring the new current row into view if necessary.

The **Bookmark** property is defined as a Variant to accommodate user-defined bookmarks in unbound mode.

In the UnboundReadData event there may be multiple rows, so you must provide a bookmark for each row.

The UnboundWriteData event passes a bookmark to you to identify the row of data to be updated.

The UnboundAddData event passes a bookmark to you to identify the row of data to be added.

Note In unbound mode, setting the **Bookmark** property to itself will force the current row to be updated via the UnboundWriteData event.

Bookmark Property (DBGrid) Example

In this example, when the user deletes a row in the unbound **DBGrid** control, the UnboundDeleteRow event is triggered, allowing you to manually delete the row from your data set, in this case, a simple array. The following code fragment shows how a bookmark is passed as an argument in the UnboundDeleteRow event to identify the row to be deleted.

```
Private Sub DBGrid1_UnboundDeleteRow(Bookmark As Variant)
    For i% = Bookmark + 1 To RowCount - 1
        For j% = 0 to MAXCOLS - 1
            UserData(j%, I% - 1) = UserData(j%, I%)
        Next j%
    Next I%
End Sub
```

Refer to the UnboundReadData event example for an example of assigning data to the **Bookmark** property.

ColumnName Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproColumnNameC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproColumnNameX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproColumnNameA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproColumnNameS"}

Returns the name of the specified column within a **RowBuffer** object in an unbound **DBGrid** control.

Syntax

object.**ColumnName** (*column*)

The **ColumnName** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
column	A string specifying the column for which you want the name.

Remarks

Enables you to obtain the name of a specified column.

ColumnCount Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproColumnCountC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproColumnCountX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproColumnCountA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproColumnCountS"}

Returns the number of columns present in a **RowBuffer** object in an unbound **DBGrid** control.

Syntax

object.**ColumnCount**

The *object* placeholder represents an object expression that evaluates to an object in the Applies To list.

RowCount Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproRowCountC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproRowCountX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproRowCountA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproRowCountS"}

Returns or sets the number of rows contained in the **RowBuffer** object in an unbound **DBGrid** control.

Syntax

object.**RowCount** [= *number*]

The **RowCount** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
number	A long integer indicating the number of rows.

Remarks

In the UnboundWriteData event, this property specifies how many rows will be written, and is always set to a value of 1, since you can write only one row of data at a time. You can set this property to a value of 0, however, to indicate that the write operation failed.

In the UnboundAddData event, this property is always set to a value of 1, since only a single row can be added at a time. You can set this property to a value of 0, however, to indicate that the add operation failed.

In the UnboundReadData event, this property indicates how many rows have been requested to be filled. After filling those rows, you should set **RowCount** to the number of rows actually stored in the **RowBuffer** object.

RowCount Property Example

In this example, the **RowCount** property is used to process the correct amount of rows in the **RowBuffer** object.

```
For Row% = 0 To RowBuf.RowCount - 1
' Process RowCount number of rows
Next Row%
```

Value Property (RowBuffer)

Returns or sets the value of an item of data within the **RowBuffer** object in an unbound **DBGrid** control.

Syntax

object.**Value** (*row, column*) [= *value*]

The **Value** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
row	An integer from 0 to **RowCount** – 1 indicating the row number of the item.
column	An integer from 0 to **ColumnCount** – 1 indicating the column number of the item.
value	A variant value entered or displayed in the cell specified by *row* and *column*.

Remarks

The **RowBuffer** object can contain several columns and several rows of data.

In the UnboundWriteData and UnboundAddData events, the user enters the setting of the **Value** property into the unbound **DBGrid** control. You then use this value to update your database.

In the UnboundReadData event, you supply the values of the *row* and *column* arguments requested by the unbound **DBGrid** control.

Value Property (RowBuffer) Example

This example illustrates how a previously read row, contained in the **Value** property, is added to a database; in this case, a simple array. In the following code fragment, if the **RowBuffer** object contains a row (**Not IsNull**), the row is written to the array.

```
For i% = 0 To MAXCOLS - 1
   If Not IsNull(RowBuf.Value(0, i%)) Then
      UserData(i%, RowCount - 1) = RowBuf.Value(0, i%)
   End If
Next i%
```

DBGrid Control Constants

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstDBGridControlConstantsC;vbproBooksOnlineJumpTopic"}

Alignment Constants

Constant	Value	Description
dbgLeft	0	Left
dbgRight	1	Right
dbgCenter	2	Center
dbgGeneral	3	General

Borderstyle Constants

Constant	Value	Description
dbgNoBorder	0	None
dbgFixedSingle	1	FixedSingle

Datamode Constants

Constant	Value	Description
dbgBound	0	Bound
dbgUnbound	1	Unbound

Dividerstyle and RowDividerStyle Constants

Constant	Value	Description
dbgNoDividers	0	NoDividers
dbgBlackLine	1	BlackLine
dbgDarkGrayLine	2	DarkGrayLine
dbgRaised	3	Raised
dbgInset	4	Inset
dbgUseForeColor	5	UseForeColor

Error Constants

Constant	Value	Description
dbgADDCELLCOND	6252	Error adding cell condition
dbgADDERROR	6154	Error occurred while trying to add record
dbgAPPLYSTYLE	6254	Error applying style
dbgBADEVENT	6161	Operation is invalid within the event
dbgBADSELRIDX	6150	Invalid selected row bookmark index
dbgBINDERROR	4097	Cannot initialize data bindings
dbgBMPTOOLARGE	6255	Bitmap is too large
dbgCAPTOOLONG	6164	Caption text is too long
dbgCNOTFOUND	6147	Column not found
dbgCOLDATA	6156	Data type mismatch during field update
dbgCOLINDEX	6145	Invalid column index
dbgDATAACCESS	6160	Data access error

dbgDELERROR	6155	Error occurred while trying to delete record
dbgDELMULTROWS	6159	Cannot delete multiple rows
dbgDUPSTYLE	6248	Duplicate style name
dbgFIELDERR	6158	Not a valid field name
dbgINCOMPAT	6157	Data type incompatible with column data type
dbgINVBOOKMARK	6149	Invalid bookmark
dbgINVPROPVAL	4098	Invalid setting for property
dbgINVROWNUM	6148	Invalid row number
dbgINVSBSTYLE	6152	Invalid setting for ScrollBars property
dbgNOCURREC	6163	No current record
dbgNOPROPNOW	6162	Property is not available in this context
dbgNOTINIT	6146	Control not properly initialized
dbgREMSTYLE	6251	Error removing style
dbgSCROLLRANGE	6151	Scroll arguments out of range
dbgSPLITINDEX	6244	Invalid split index
dbgSTYLEERR	6249	Error accessing style
dbgSTYLEINDEX	6247	Invalid style index
dbgSTYLENAME	6253	Invalid style name
dbgUPDERROR	6153	Error occurred while trying to update record
dbgUPDSTYLE	6250	Error updating style
dbgVITEMERR	6246	Error accessing value item
dbgVLINDEX	6245	Invalid value list index

Scroll Bar Constants

Constant	Value	Description
dbgNone	0	None
dbgHorizontal	1	Horizontal
dbgVertical	2	Vertical
dbgBoth	3	Both
dbgAutomatic	4	Automatic

SplitSizeModeConstants

Constant	Value	Description
dbgScalable	0	Scalable
dbgExact	1	Exact
dbgNumberOfColumns	2	Number of columns

Split Object

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbobjSplitObjectC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbobjSplitObjectX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"vbobjSplitObjectP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"vbobjSplitObjectM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"vbobjSplitObjectE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbobjSplitObjectS"}

A **Split** object represents a split within a **DBGrid** control.

Remarks

DBGrid supports Excel-like splits that divide the grid into vertical panes to provide users with different views of a database. Each split is represented by a **Split** object and contains a group of adjacent columns that scroll as a unit. When a **DBGrid** object is created, it contains one **Split** object by default.

You can use splits to present your data in multiple vertical panes. The data panes (or splits) can display data in different colors and fonts. They can scroll (vertically) together or independently of each other, and they can display the same or different columns. You can also use splits to fix one or more columns from scrolling. Unlike other grid products, the fixed columns do not have to be at the left edge of the grid, but can be at the right edge or anywhere in the middle. You can even have multiple groups of fixed columns within a grid.

Each **Split** object maintains its own Columns collection. These independent splits and columns provide you with very powerful and flexible data presentation capabilities.

As mentioned above, a grid (a **DBGrid** object) initially contains a single split. If additional splits are created, you can determine or set the current split (i.e., the split that has received focus) using the grid's **Split** property as follows:

```
' Read the zero-based index of the current split
Variable% = DBGrid1.Split

' Set focus to the split with an index equal to
' Variable%
DBGrid1.Split = Variable%
```

Each split in a grid is a different view of the same data source, and each split behaves just like an independent grid. If you create additional **Split** objects without customizing any of the split properties, all splits will be identical and each will behave very much like the original grid with one split.

Some of the properties of the **DBGrid** control are the same as the properties of a **Split** object and are considered common. Changes made to a **DBGrid** control common property also change the same property of the current **Split** object and vice versa. For example, consider a grid with two splits, and assume that the current split index is 1 (i.e., the grid's **Split** property is set to 1). If you want to determine the marquee style in use, the following statements are identical:

```
marquee% = DBGrid1.MarqueeStyle
marquee% = DBGrid1.Splits(1).MarqueeStyle
```

If the current split index is set to 1, then the following code is equivalent for setting the **MarqeeStyle** property to **dbgSolidCellBorder**:

```
DBGrid1.MarqueeStyle = dbgSolidCellBorder
DBGrid1.Splits(1).MarqueeStyle = dbgSolidCellBorder
```

Note Common properties are unique to **DBGrid** objects and their associated **Split** objects. No other object pairs possess similar relationships.

Splits Collection

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcolSplitsCollectionC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbcolSplitsCollectionX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"vbcolSplitsCollectionP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"vamthItem;vbcolSplitsCollectionM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"vbcolSplitsCollectionE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbcolSplitsCollectionS"}

The Splits collection contains all stored **Split** objects in a **DBGrid** control.

Syntax

Splits(*index***)**
Splits.Item(*index***)**

Remarks

You can create splits at design time using the grid's UI-active context menu. At run time, you can create and remove splits using the Splits collection's **Add** and **Remove** methods. Each method takes a zero-based split index. The following code demonstrates adding and removing splits at run time:

```
' Create a Split object with index 0
DBGrid1.Splits.Add 0
' Remove the Split object with index 1
DBGrid1.Splits.Remove 1
```

You can determine the number of splits in a grid using the Splits collection's **Count** property.

Split Object, Splits Collection Example

The following example adds a new fixed split to a **DBGrid** control. To try this example, add a **Data** control and a **DBGrid** control to the form of a new Standard EXE project. Set the **DatabaseName** and **RecordSource** properties of the **Data** control to "BIBLIO.MDB" and "AUTHORS". Paste the following code into the General Declarations section of the form:

```
Private Sub Form_Load()
   ' Before modifying the grid's properties, make sure
   ' the grid is initialized by refreshing the Data
   ' control.
   Data1.Refresh

   ' Create an additional splits:
   DBGrid1.Splits.Add 0 ' Create an additional split

   ' Hide all columns in the leftmost split,
   ' Splits(0), except for columns 0 and 1
   Dim Cols As Columns, C As Column
   Set Cols = DBGrid1.Splits(0).Columns
   For Each C In Cols
      C.Visible = False
   Next C
   Cols(0).Visible = True
   Cols(1).Visible = True

   ' Configure Splits(0) to display exactly two
   ' columns, and disable resizing
   With DBGrid1.Splits(0)
      .SizeMode = dbgNumberOfColumns
      .Size = 2
      .AllowSizing = False
   End With

   ' Usually, if you fix columns 0 and 1 from
   ' scrolling in a split, you will want to make them
   ' invisible in other splits:
   Set Cols = DBGrid1.Splits(1).Columns
   Cols(0).Visible = False
   Cols(1).Visible = False

   ' Turn off the record selectors in Split 1:
   DBGrid1.Splits(1).RecordSelectors = False
End Sub
```

AfterColEdit Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtAfterColEditEventC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtAfterColEditEventX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtAfterColEditEventA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtAfterColEditEventS"}

Occurs after editing is completed in a grid cell.

Syntax

Private Sub *object*_**AfterColEdit(**[*index* **As Integer**,] **ByVal** *colindex* **As Integer)**

The AfterColEdit event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
Index	An integer that identifies a control if it is in a control array.
colindex	An integer that identifies the column that was edited.

Remarks

When the user completes editing within a grid cell, as when tabbing to another column in the same row, pressing the ENTER key, or clicking on another cell, the BeforeColUpdate and AfterColUpdate events are executed, and data from the cell is moved to the grid's copy buffer. The AfterColEdit event immediately follows the AfterColUpdate event.

When editing is completed in a grid cell, this event is always triggered, even if no changes were made to the cell or the BeforeColUpdate event was canceled.

The AfterColEdit event will not be fired if the BeforeColEdit event is canceled.

BeforeColEdit Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtBeforeColEditEventC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtBeforeColEditEventX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtBeforeColEditEventA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtBeforeColEditEventS"}

Occurs just before the user enters edit mode by typing a character.

Syntax

Private Sub *object*_**BeforeColEdit(**[*index* **As Integer**,] **ByVal** *colindex* **As Integer**, **ByVal** *keyascii* **As Integer**, *cancel* **As Integer)**

The BeforeColEdit event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
Index	An integer that identifies a control if it is in a control array.
colindex	An integer that identifies the column to be edited.
keyascii	An integer representing the ANSI key code of the character typed by the user to initiate editing, or 0 if the user initiated editing by clicking the mouse. KeyAscii is passed by value, not by reference; you cannot change its value to initiate editing with a different character.
cancel	An integer that may be set to **True** to prevent the user from editing the cell, as described in Settings.

Settings

The settings for *cancel* are:

Setting	Description
True	The cell will not enter edit mode
False	(Default) The ColEdit event is fired immediately, followed by the Change and KeyUp events for the KeyAscii argument, if non-zero.

Remarks

If a floating editor marquee is not in use, this event also occurs when the user clicks the current cell or double clicks another cell.

Use this event to control the editability of cells on a per-cell basis, or to translate the initial keystroke into a default value.

Note The *keyascii* argument can only be 0 if a floating editor marquee is not in use.

ButtonClick Event (DBGrid Control)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtButtonClickEventDBGridControlC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtButtonClickEventDBGridControlX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtButtonClickEventDBGridControlA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtButtonClickEventDBGridControlS"}

Occurs when the current cell's built-in button is clicked.

Syntax

Private Sub *object*_**ButtonClick(**[*index* **As Integer**,] **ByVal** *colindex* **As Integer)**

The ButtonClick event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
Index	An integer that identifies a control if it is in a control array.
colindex	An integer that identifies the column whose button was clicked.

Remarks

The built-in button is enabled for a column when its **Button** property is set to **True**.

Typically, you enable the column button when you want to drop down a **Visual Basic** control (such as the built-in combo box, a bound list box, or even another **DBGrid** control) for editing or data entry. When the button in the current cell is clicked, the ButtonClick event will be fired. You can then write code to drop down the desired control from the cell.

ColEdit Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtColEditEventC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtColEditEventX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtColEditEventA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtColEditEventS"}

Occurs when a cell first enters edit mode by typing a character.

Syntax

Private Sub *object*_**ColEdit(**[*index* **As Integer**,] **ByVal** *colindex* **As Integer)**

The ColEdit event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
Index	An integer that identifies a control if it is in a control array.
colindex	An integer that identifies the column being edited.

Remarks

If a floating editor marquee is not in use, this event also occurs when the user clicks the current cell or double clicks another cell.

The ColEdit event immediately follows the BeforeColEdit event only when the latter is not canceled.

When the user completes editing within a grid cell, as when tabbing to another column in the same row, pressing the ENTER key, or clicking on another cell, the BeforeColUpdate and AfterColUpdate events are executed if the data has been changed. The AfterColEdit event is then fired to indicate that editing is completed.

Error Event (DBGrid Control)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtErrorEventDBGridControlC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtErrorEventDBGridControlX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtErrorEventDBGridControlA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtErrorEventDBGridControlS"}

Occurs only as the result of a data access error that takes place when no Visual Basic code is being executed.

Syntax

Private Sub *object*_**Error(**[*index* **As Integer**,] **ByVal** *dataerror* **As Integer**, *response* **As Integer)**

The Error event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
Index	An integer that identifies a control if it is in a control array.
dataerror	An integer that identifies the error that occurred.
response	An integer that may be set to 0 to suppress error message display, as described in Settings.

Settings

The settings for *response* are:

Setting	Description
0	No error message will be displayed.
1	(Default) The message associated with the error will be displayed.

Remarks

Even if your application handles run time errors in code, errors can still occur when none of your code is executing, as when the user clicks a **Data** control button or changes the current record by interacting with a bound control. If a data access error results from such an action, the Error event is fired.

Not adding code to this event is equivalent to setting the *response* argument to 0.

Note Use the **ErrorText** property to retrieve the error string that will be displayed.

OnAddNew Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtOnAddNewEventC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtOnAddNewEventX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtOnAddNewEventA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtOnAddNewEventS"}

Occurs when a user action invokes an AddNew operation.

Syntax

Private Sub *object*_**OnAddNew(**[*index* **As Integer**]**)**

The OnAddNew event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.

Remarks

The OnAddNew event occurs when an AddNew operation has been initiated by either of the following:

- The user modifies a cell within the AddNew row. Typically, this occurs as soon as the user types a character, but may also occur as a result of a built-in radio button or combo box selection.
- The **Value** or **Text** property of a column is set in code when the current cell is within the AddNew row.

This event is fired in both bound and unbound modes. However, it will only be fired if the grid's **AllowAddNew** property is **True**.

When the OnAddNew event is fired, the value of the **AddNewMode** property is 2 - AddNew Pending.

SelChange Event (DBGrid Control)

Occurs when the user selects a different range of rows or columns.

Syntax

Private Sub *object*_**SelChange(**[*index* **As Integer**,] *cancel* **As Integer)**

The SelChange event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
cancel	A Boolean expression that determines if a change is canceled, as described in Settings.

Settings

The settings for *cancel* are:

Setting	Description
True	If your event procedure sets the Cancel argument to **True**, the previous row and column selections (if any) are restored, and the **SelStartCol**, **SelEndCol**, and **SelBookmarks** properties revert to their previous values
False	(Default) Continues with change.

Remarks

This event is triggered under several circumstances:

- When the user selects a single row by clicking its record selector.
- When the user adds a row to the list of selected rows by clicking its record selector while holding down the CTRL key.
- When the user selects a single column by clicking its header.
- When the user changes the range of selected columns by dragging to an adjacent column within the header row.
- When the user extends the range of selected columns by holding down the SHIFT key and clicking on an unselected column header.
- When the user clears the current row or column selection by clicking an individual cell, this event will fire before the RowColChange event.

The current range of selected columns is provided by the **SelStartCol** and **SelEndCol** properties. The bookmarks of the selected rows are available in the collection provided by the **SelBookmarks** property. Within this event procedure, these properties reflect the user's pending selection(s).

This event is only triggered by user interaction with the grid. It cannot be triggered by code.

Note When the user selects a column, any row selections are cleared. Similarly, when the user selects a row, any column selections are cleared.

SplitChange Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtSplitChangeEventC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtSplitChangeEventX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtSplitChangeEventA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtSplitChangeEventS"}

Occurs when the current cell changes to a different cell in another split.

Syntax

Private Sub *object*_**SplitChange([** *index* **As Integer])**

The SplitChange event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.

Remarks

This event is triggered under several circumstances:

- When the grid is first displayed.
- When the user clicks a cell in another split (subject to the setting of the **AllowFocus** property).
- When the user presses a navigation key to cross a split boundary (subject to the setting of the **TabAcrossSplits** property).
- When the **Split** property is changed in code to a different value.
- When a new split is inserted before the current split via code or user interaction.
- When the current split is removed via code or user interaction.

If the user edits data and then moves the current cell position to a new row in another split, the update events for the original row are completed before the SplitChange event is executed.

If a split change also results in a change to the current row or column, then the SplitChange event will always precede the RowColChange event.

UnboundGetRelativeBookmark Event

Occurs whenever the grid requires data for display.

Syntax

Private Sub *object*_**UnboundGetRelativeBookmark(**[*index* **As Integer**,] *startlocation* **As Variant**, **ByVal** *offset* **As Long**, *newlocation* **As Variant**, *approximateposition* **As Long)**

The UnboundGetRelativeBookmark event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
startlocation	A bookmark which, together with *offset*, specifies the row to be returned in *newlocation*. A *startlocation* of **Null** indicates a request for a row from BOF or EOF.
offset	Specifies the relative position (from *startlocation*) of the row to be returned in *newlocation*. A positive number indicates a forward relative position while a negative number indicates a backward relative position.
newlocation	The bookmark of the row which is specified by *startlocation* plus *offset*. If the row specified is beyond the first or the last row (or beyond BOF or EOF), then *newlocation* should be set to **Null**.
approximateposition	A value which indicates the ordinal position of *newlocation*. Setting this variable will enhance the ability of the grid to display its vertical scroll bar accurately. If the exact ordinal position of *newlocation* is not known, user can set it to a reasonable, approximate value, or just ignore this parameter.

Remarks

This optional event is to be used in conjunction with the UnboundReadData event (when the **DataMode** property is set to 1 - Unbound). It is fired by the grid whenever it requires data for display. You can add code to this event to improve your project performance (often quite dramatically), otherwise, you can ignore this event and your project will function properly.

Before returning from this event, the user is expected to set *newlocation* and (optionally) *approximateposition*. For example:

If *offset* is 1 (or -1), then the user returns in *newlocation*, the bookmark of the next (or previous) row from *startlocation*.

If *startlocation* is **NULL** and *offset* is 2 (or -2), then the user returns in *newlocation*, the bookmark of the second (or second to last) row.

If the requested row is beyond the first or last row (or beyond BOF or EOF), then the user returns

Null in *newlocation*.

ClearFields Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproClearFieldsMethodC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproClearFieldsMethodX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproClearFieldsMethodA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthClearFieldsMethodS;vbproClearFieldsMethodS"}

Restores the default grid layout.

Syntax

object.**ClearFields**

The *object* placeholder represents an object expression that evaluates to an object in the Applies To list.

Remarks

The **ClearFields** method restores the default grid layout (with two blank columns) so that subsequent ReBind operations will automatically derive new column bindings from the (possibly changed) data source. You can cancel the grid's automatic layout behavior by invoking the **HoldFields** method.

ClearSelCols Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproClearSelColsMethodC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproClearSelColsMethodX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproClearSelColsMethodA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthClearSelColsMethodS;vbproClearSelColsMethodS"}

Deselects all columns in a split. If no columns are selected, then this method does nothing.

Syntax

object.**ClearSelCols**

The *object* placeholder represents an object expression that evaluates to an object in the Applies To list.

Remarks

If a grid contains multiple splits, then invoking its **ClearSelCols** method has the same effect as invoking the **ClearSelCols** method for the current split. The index of the current split is available through the **DBGrid** control's **Split** property.

Use the **SelStartCol** and **SelEndCol** properties to determine the current column selection range for a split.

HoldFields Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproHoldFieldsMethodC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHoldFieldsMethodX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHoldFieldsMethodA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthHoldFieldsMethodS;vbproHoldFieldsMethodS"}

Sets the current column/field layout as the customized layout.

Syntax

object.**HoldFields**

The *object* placeholder represents an object expression that evaluates to an object in the Applies To list.

Remarks

The **HoldFields** method sets the current column/field layout as the customized layout so that subsequent ReBind operations will use the current layout for display. You can resume the grid's automatic layout behavior by invoking the **ClearFields** method.

SplitContaining Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproSplitContainingMethodC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproSplitContainingMethodX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproSplitContainingMethodA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthSplitContainingMethodS;vbproSplitContainingMethodS"}

Returns the **Index** value of the split containing the specified coordinate pair.

Syntax

object.**SplitContaining** *x, y*

The **SplitContaining** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
x	Required. A single precision value that defines the horizontal coordinate, based on the coordinate system of the grid's container.
y	Required. A single precision value that defines the vertical coordinate, based on the coordinate system of the grid's container.

Remarks

This value ranges from 0 to 1 less than the setting of the **Count** property of the Splits collection (0 to **Splits.Count** - 1).

This method is useful when working with mouse and drag events when you are trying to determine where the user clicked or dropped another control in terms of a grid column.

If either argument is outside of the grid's data area, this method returns -1.

Split

Similar to the "split window" capabilities available with Microsoft Excel where a window is divided into separate vertical panes to provide users with different views of data. Each split is represented by a **Split** object and contains a group of adjacent columns that scroll as a unit.

Button Property (Column Object)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproButtonPropertyColumnObjectC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproButtonPropertyColumnObjectX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproButtonPropertyColumnObjectA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproButtonPropertyColumnObjectS"}

Sets or returns a value that determines whether a button is displayed within the current cell.

Syntax

object.**Button** [**=** *value*]

The **Button** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines if a button is displayed within the current cell, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	A button will be displayed in the upper right corner of the current cell at run time.
False	(Default) No button will be displayed.

Remarks

Typically, you enable the column button when you want to drop down a control (such as the built-in combo box, a bound list box, or even another **DBGrid** control) for editing or data entry. When the button in the current cell is clicked, the ButtonClick event will be fired. You can then write code to drop down the desired control from the cell.

Value Property (Column Object)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproValuePropertyColumnObjectC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproValuePropertyColumnObjectX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproValuePropertyColumnObjectA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproValuePropertyColumnObjectS"}

Sets or returns the underlying data value in a column for the current row. Not available at design time.

Syntax

object.**Value** [**=** *value*]

The **Value** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A string expression that represents the underlying data value in a column for the current row.

Remarks

The **Value** property is useful for simulating data entry within a cell. When this property is set, the value displayed in the cell respects the setting of the column's **NumberFormat** property.

This property always returns a string variant, even if the data type of the underlying field is numeric.

Use the **Text** property to access the formatted data value in a column for the current row.

Caption Property (DBGrid Control, Column Object)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproCaptionPropertyColumnObjectC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproCaptionPropertyColumnObjectX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproCaptionPropertyColumnObjectA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproCaptionPropertyColumnObjectS"}

For a **DBGrid** control, this property determines the text displayed in the caption bar at the top of the grid.

For a **Column** object, this property determines the text displayed in the column's heading area.

Syntax

object.**Caption** [**=** *value*]

The **Caption** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A string expression that determines what is displayed, as described below.

Remarks

Setting the **Caption** property to an empty string for a **DBGrid** control hides its caption bar.

Setting the **Caption** property to an empty string for a **Column** object clears the text in the column's heading area but does not hide the heading. Column captions are only displayed if the **DBGrid** control's **ColumnHeaders** property is set to **True** and the **HeadLines** property is not set to 0.

WrapText Property (Column Object)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproWrapTextPropertyColumnObjectC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproWrapTextPropertyColumnObjectX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproWrapTextPropertyColumnObjectA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproWrapTextPropertyColumnObjectS"}

Sets or returns a value indicating whether an object word wraps text at cell boundaries.

Syntax

object.**WrapText** [**=** *value*]

The **WrapText** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines whether an object word wraps, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	A line break occurs before words that would otherwise be partially displayed.
False	(Default) No line break occurs and text is clipped at the cell's right edge.

Remarks

Use this property in conjunction with the **RowHeight** property to produce multi-line displays.

AddNewMode Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproAddNewModePropertyC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproAddNewModePropertyX"} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproAddNewModePropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproAddNewModePropertyS"}

Returns a value that describes the location of the current cell with respect to the grid's AddNew row. Read-only at run time and not available at design time.

Syntax

object.**AddNewMode**

The **AddNewMode** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.

Values

The **AddNewMode** property returns one of the following:

Constant	Value	Description
dbgNoAddNew	0	The current cell is not in the last row, and no **AddNew** operation is pending.
dbgAddNewCurrent	1	The current cell is in the last row, but no **AddNew** operation is pending.
dbgAddNewPending	2	The current cell is in the next to last row as a result of a pending **AddNew** operation initiated by the user through the grid's user interface, or by code as a result of setting the **Value** or **Text** properties of a column.

Remarks

If the **AllowAddNew** property is **True**, the last row displayed in the grid is left blank to permit users to enter new records. If the **AllowAddNew** property is **False**, the blank row is not displayed, and **AddNewMode** always returns 0.

Note This property is valid in both bound and unbound modes.

AllowArrows Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproAllowArrowsPropertyC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproAllowArrowsPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproAllowArrowsPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproAllowArrowsPropertyS"}

Sets or returns a value that determines whether the control uses the arrow keys for grid navigation.

Syntax

object.**AllowArrows** [**=** *value*]

The **AllowArrows** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines the arrow keys are used for grid navigation, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	(Default) The user can use the arrow keys to move from cell to cell within the same row.
False	The left and right arrow keys will move focus from control to control and cannot be used to move between cells.

Remarks

The user cannot use the arrow keys to move out of the **DBGrid** control when this property is set to **True**. If the **WrapCellPointer** property is also set to **True**, then the arrow keys will wrap around rows and the user can navigate the entire grid using the arrow keys.

ApproxCount Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproApproxCountPropertyC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproApproxCountPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproApproxCountPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproApproxCountPropertyS"}

Sets or returns the approximate number of rows in the grid.

Syntax

object.**ApproxCount** [**=** *value*]

The **ApproxCount** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	An integer that represents the approximate number of rows in the grid.

Remarks

This property sets or returns the approximate row count used by the grid to calibrate the vertical scroll bar.

Typically, the **ApproxCount** property is used in unbound mode to improve the accuracy of the vertical scroll bar. This is particularly useful for situations where the number of rows is known in advance, such as when an unbound grid is used in conjunction with an array.

Note For a bound grid, setting the **ApproxCount** property has no effect. However, getting the **ApproxCount** property will query the underlying data source.

CurrentCellModified Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproCurrentCellModifiedPropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproCurrentCellModifiedPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproCurrentCellModifiedPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproCurrentCellModifiedPropertyS"}

Sets or returns modification status of the current cell. Not available at design time.

Syntax

object.**CurrentCellModified** [**=** *value*]

The **CurrentCellModified** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines the modification status of the current cell, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	Editing is in progress and the current cell (indicated by the **Bookmark** and **Col** properties) has been modified by the user.
False	The cell has not been modified or editing is not in progress.

Remarks

You can use this property to cancel any changes the user has made to the current text. For example, to program a function key to discard the user's changes (like the ESC key), trap the key code in the grid's KeyDown event and set **CurrentCellModified** to **False**. This will revert the current cell to its original contents.

CurrentCellVisible Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproCurrentCellVisiblePropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproCurrentCellVisiblePropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproCurrentCellVisiblePropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproCurrentCellVisiblePropertyS"}

Sets or returns the visibility of the current cell. Not available at design time.

Syntax

object.**CurrentCellVisible** [**=** *value*]

The **CurrentCellVisible** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines the visibility of the current cell, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	The current cell (indicated by the **Bookmark** and **Col** properties) is visible within the displayed area of a grid or split.
False	The cell is not visible.

Remarks

For a **DBGrid** control, setting the **CurrentCellVisible** property to **True** causes the grid to scroll so that the current cell is brought into view. If a grid contains multiple splits, then the current cell becomes visible in each split.

For a **Split** object, setting the **CurrentCellVisible** property to **True** makes the current cell visible in that split only.

In all cases, setting this property to **False** is meaningless and is ignored.

EditActive Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproEditActivePropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproEditActivePropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproEditActivePropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproEditActivePropertyS"}

Sets or returns the editing status of the current cell. Not available at design time.

Syntax

object.**EditActive** [**=** *value*]

The **EditActive** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines the editing status, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	The current cell is currently being edited by the user.
False	No editing is in progress.

Remarks

If the grid is not already in edit mode, setting **EditActive** to **True** will initiate editing on the current cell. The caret will be positioned at the end of the cell and the ColEdit event will be triggered.

If the grid is already in edit mode, setting **EditActive** to **False** will exit edit mode. If the cell has been modified, this will trigger the following events: BeforeColUpdate, AfterColUpdate, and AfterColEdit.

Note To cancel editing completely, set the **CurrentCellModified** property to **False**, then set **EditActive** to **False**.

ErrorText Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproErrorTextPropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproErrorTextPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproErrorTextPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproErrorTextPropertyS"}

Returns the error message string from the underlying data source. Not available at design time.

Syntax

object.**ErrorText**

The **ErrorText** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.

Remarks

When a database error occurs as a result of user interaction with the grid, such as when the user enters text into a numeric field and then attempts to update the current record by moving to another row, the grid's Error event will fire. However, the error code passed to the event handler in the *DataError* parameter may not identify the specific error that occurred, or may even differ across operating environments. For these reasons, the **ErrorText** property is provided so that your application can parse the actual error message to determine the nature of the error.

Note The **ErrorText** property is only valid within a **DBGrid** control's Error event handler. A trappable error will occur if you attempt to access it in any other context.

hWndEditor Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproHWndEditorPropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHWndEditorPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHWndEditorPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHWndEditorPropertyS"}

Returns the unique window handle assigned to a **DBGrid** control's editing window by the Microsoft Windows operating environment. Not available at design time.

Syntax

object.**hWndEditor**

The **hWndEditor** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.

Remarks

Experienced users can pass the value of this property to Windows API calls that require a valid window handle.

When editing is not in progress, this property returns 0.

Note Since the value of this property can change while a program is running, never store the **hWndEditor** value in a variable. Also, do not use the **hWndEditor** property to test whether editing is in progress. The **EditActive** property is provided for this purpose.

MarqueeStyle Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproMarqueeStylePropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMarqueeStylePropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMarqueeStylePropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMarqueeStylePropertyS"}

Sets or returns the Marquee style for the **DBGrid** control or **Split** object.

Syntax

object.**MarqueeStyle** [**=** *value*]

The **MarqueeStyle** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A number or constant that specifies the Marqee style, as described in Settings.

Settings

The settings for *value* are:

Constant	Value	Description
dbgDottedCellBorder	0	The current cell within the current row will be highlighted by drawing a dotted border around the cell. In Microsoft Windows terminology, this is usually called a focus rectangle.
dbgSolidCellBorder	1	The current cell within the current row will be highlighted by drawing a solid box around the current cell. This is more visible than the dotted cell border, especially when 3-D divider properties are used for the grid.
dbgHighlightCell	2	The entire current cell will be highlighted by inverting the colors within the cell. This provides a very distinctive block-style highlight for the current cell.
dbgHighlightRow	3	The entire row containing the current cell will be highlighted by inverting the colors within the row. In this mode, it is not possible to visually determine which cell is the current cell, only the current row. When the grid or split is not editable, this setting is often preferred, since cell position is then irrelevant.
dbgHighlightRowRaiseCell	4	The entire row will be highlighted. The current cell within the row will be "raised" so that it appears distinctive. This setting doesn't appear clearly with all background color and divider settings. The best effect is achieved

		by using 3-D dividers and a light gray background.
dbgNoMarquee	5	The marquee will not be shown. This setting is useful for cases where the current row is irrelevant, or where you don't want to draw the user's attention to the grid until necessary.
dbgFloatingEditor	6	The current cell will be highlighted by a floating text editor window with a blinking caret (as in Microsoft Access). This is the default setting.

Remarks

If a grid contains multiple splits, then setting its **MarqueeStyle** property has the same effect as setting the **MarqueeStyle** property of each split individually.

Note If the floating editor marquee setting is in effect and the current cell contains radio buttons or graphics, then a dotted focus rectangle will be displayed.

MarqueeUnique Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproMarqueeUniquePropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMarqueeUniquePropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMarqueeUniquePropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMarqueeUniquePropertyS"}

Sets or returns a value that determines whether the marquee is displayed only in the current split.

Syntax

object.**MarqueeUnique** [= *value*]

The **MarqueeUnique** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines whether the marquee is displayed only in the current split, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	(Default) The current cell marquee is only displayed within the current split.
False	All splits with a **MarqueeStyle** setting as described in Remarks will display a marquee at the current cell, provided that the current cell is visible.

Remarks

The current cell marquee is only displayed when the **MarqueeStyle** property for a grid or split has a value of 0, 1, 2, or 4.

In most cases, a single current cell marquee is preferable, and you will not need to change this property.

If this property is set to **False**, you may then see several different current cell marquees. The actual current cell is determined by the setting of the **Split** property.

Note Although the floating editor **MarqueeStyle** (6) is technically a current cell marquee, only one floating editor will be displayed, even if **MarqueeUnique** is set to **False**.

Split Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproSplitPropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproSplitPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproSplitPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproSplitPropertyS"}

Sets or returns the index of the current split. Not available at design time.

Syntax

object.**Split** [**=** *value*]

The **Split** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	An integer that specifies the index of the current split, as described in Remarks.

Remarks

The **Split** property specifies a zero-based index of the current split.

Splits Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproSplitsPropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproSplitsPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproSplitsPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproSplitsPropertyS"}

Returns a collection of **Split** objects. Not available at design time.

Syntax

object.**Splits**

The **Splits** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.

TabAcrossSplits Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproTabAcrossSplitsPropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproTabAcrossSplitsPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproTabAcrossSplitsPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproTabAcrossSplitsPropertyS"}

Sets or returns the behavior of the tab and arrow keys at split borders.

Syntax

object.**TabAcrossSplits** [**=** *value*]

The **TabAcrossSplits** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines the behavior of the tab and arrow keys at split borders, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	Tab and arrow keys will move the current cell across split boundaries. When at the last column of the rightmost split (or the first column of the leftmost split), they will either wrap to the next row, stop, or move to other controls depending on the values of the **WrapCellPointer** and **TabAction** properties.
False	(Default) The tab and arrow keys will not move the current cell across split boundaries. They will either wrap to the next row, stop, or move to other controls depending on the values of the **WrapCellPointer** and **TabAction** properties.

Remarks

The **TabAcrossSplits** property does not determine if the tab and arrow keys will move from cell to cell, or from control to control, or wrap to the next row. Use the **AllowArrows**, **WrapCellPointer**, and **TabAction** properties to control this behavior. If the tab and arrow keys are able to move from cell to cell, this property determines whether they will move across split boundaries to adjacent splits.

TabAction Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproTabActionPropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproTabActionPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproTabActionPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproTabActionPropertyS"}

Sets or returns a value that defines the behavior of the tab key.

Syntax

object.**TabAction** [**=** *value*]

The **TabAction** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A number or constant that defines the behavior of the tab key, as described in Settings.

Settings

The settings for *value* are:

Constant	Value	Description
dbgControlNavigation	0	(Default) The tab key moves to the next or previous control on the form.
dbgColumnNavigation	1	The tab key moves the current cell to the next or previous column. However, if this action would cause the current row to change, then the next or previous control on the form receives focus.
dbgGridNavigation	2	The tab key moves the current cell to the next or previous column. The behavior of the tab key at row boundaries is determined by the **WrapCellPointer** property. When this setting is used, the tab key never results in movement to another control.

Remarks

The **TabAction** property does not determine if the tab key will cross split boundaries. Use the **TabAcrossSplits** property to control this behavior.

WrapCellPointer Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproWrapCellPointerPropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproWrapCellPointerPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproWrapCellPointerPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproWrapCellPointerPropertyS"}

Sets or returns a value that determines the behavior of the arrow keys.

Syntax

object.**WrapCellPointer** [**=** *value*]

The **WrapCellPointer** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines the behavior of arrow keys, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	The cell pointer will wrap from the last column to the first in the next row (or from the first column to the last in the previous row).
False	(Default) The cell pointer will not wrap to the next (or previous) row, but will stop at the last (or first) column of the current row.

Remarks

If **TabAcrossSplits** is **False**, the cell pointer will wrap only within the current split. If **TabAcrossSplits** is **True**, the cell pointer will move from one split to the next before wrapping occurs.

If **TabAction** is set to 2 - Grid Navigation, the tab key will behave like the arrow keys, and will automatically wrap to the next or previous cell.

AllowFocus Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproAllowFocusPropertyC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproAllowFocusPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproAllowFocusPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproAllowFocusPropertyS"}

Sets or returns a value that determines whether cells within a split can receive focus.

Syntax

object.**AllowFocus** [**=** *value*]

The **AllowFocus** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines whether a cell receives focus, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	(Default) The user will be able to interactively select the split, giving it focus.
False	The user will not be able to interactively select the split. When clicked on, the split will not receive focus and the control that previously had focus will retain it.

Remarks

Use this property in combination with the **AllowSizing** property to completely prohibit the user from making any changes to a split (by setting both properties to **False**). Unselectable splits are passed over when **TabAcrossSplits** is set to **True**.

ScrollGroup Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproScrollGroupPropertyC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproScrollGroupPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproScrollGroupPropertyA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproScrollGroupPropertyS"}

Used to synchronize vertical scrolling between splits.

Syntax

object.**ScrollGroup** [**=** *value*]

The **ScrollGroup** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	An integer expression that determines the scroll group that a split belongs to.

Remarks

This property is used to synchronize vertical scrolling between splits. All splits with the same **ScrollGroup** setting will be synchronized when vertical scrolling occurs within any one of them. Splits belonging to different groups can scroll independently, allowing different splits to display different parts of the database.

If the **ScrollBars** property for a split is set to 4 - Automatic, only the rightmost split of the group will have a vertical scroll bar. If there is only one split, setting this property has no effect.

Setting the **FirstRow** property for one split affects all other splits in the same group, keeping the group synchronized.

Newly created splits have a **ScrollGroup** value of 1.

Size Property (Split Object)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproSizePropertySplitObjectC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproSizePropertySplitObjectX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproSizePropertySplitObjectA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproSizePropertySplitObjectS"}

Sets or returns the size of a split.

Syntax

object.**Size** [**=** *value*]

The **Size** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	An integer expression that specifies the size of the split.

Remarks

The meaning of the value returned by this property is determined by the split's **SizeMode** property setting.

If **SizeMode** is set to the default value of 0 - Scalable, the value returned by the **Size** property is an integer indicating the relative size of the split with respect to other scalable splits.

If **SizeMode** is set to 1 - Exact, the value returned by the **Size** property is a floating point number indicating the exact size of the split in terms of the coordinate system of the grid's container.

If **SizeMode** is set to 2 - Number of Columns, the value returned by the **Size** property is an integer indicating the number of columns displayed in the split.

Note When there is only one split (the grid's default behavior), the split spans the entire width of the grid, the **SizeMode** property is always 0 - **dbgScalable**, and the **Size** property is always 1. Setting either of these properties has no effect when there is only one split. If there are multiple splits, and you then remove all but one, the **SizeMode** and **Size** properties of the remaining split automatically revert to 0 and 1, respectively.

SizeMode Property (Split Object)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproSizeModePropertySplitObjectC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproSizeModePropertySplitObjectX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproSizeModePropertySplitObjectA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproSizeModePropertySplitObjectS"}

Sets or returns a value that determines how the **Size** property is used to determine the actual size of a split.

Syntax

object.**SizeMode** [**=** *value*]

The **SizeMode** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A number or constant that determines how the **Size** property is used, as described in Settings.

Settings

The settings for *value* are:

Constant	Value	Description
dbgScalable	0	(default) The value returned by the **Size** property is an integer indicating the relative size of the split with respect to other scalable splits. For example, if a grid contains 3 scalable splits with **Size** properties equal to 1, 2, and 3, the size of each split would be 1/6, 1/3, and 1/2 of the total grid width, respectively.
dbgExact	1	The value returned by the **Size** property is a floating point number indicating the exact size of the split in terms of the coordinate system of the grid's container. This setting allows you to fix the size of the split so that it always has the same width, even if new splits are added or existing splits are removed.
dbgNumberOfColumns	2	The value returned by the **Size** property is an integer indicating the number of columns displayed in the split, and the split will adjust its width to display the number of full columns specified by the **Size** property. For example, if **Size** is set to 2, and the user scrolls the split horizontally, the width of the split will change so that 2 full columns are displayed, regardless of how wide the columns

are.

Remarks

When there is only one split (the grid's default behavior), the split spans the entire width of the grid, the **SizeMode** property is always 0 - **dbgScalable**, and the **Size** property is always 1. Setting either of these properties has no effect when there is only one split. If there are multiple splits, and you then remove all but one, the **SizeMode** and **Size** properties of the remaining split automatically revert to 0 and 1, respectively.

Consider a grid containing both scalable splits and splits with a fixed number of columns. If a split with a fixed number of columns is scrolled horizontally, the total width remaining for the scalable splits may change because grid columns are generally of different widths. However, the ratios of the sizes of the scalable splits remain the same as specified by their **Size** properties.

CaptureImage Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthCaptureImageMethodC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthCaptureImageMethodX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthCaptureImageMethodA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthCaptureImageMethodS"}

Returns a captured image of the grid's display in its current state.

Syntax

object.**CaptureImage**

The *object* placeholder represents an object expression that evaluates to an object in the Applies To list.

Remarks

Use the **CaptureImage** method to retrieve a snapshot of the grid.

The following code uses the **CaptureImage** method to assign a snapshot of a **DBGrid** control to a **PictureBox** control.

```
Picture1.Picture = DBGrid1.CaptureImage
```

Note The **CaptureImage** method retrieves the image as a metafile typed image. Therefore, the image will resize to the size of its container.

DBGrid Support, Documentation, and Updates

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscAPEXProductInformationC;vbproBooksOnlineJumpTopic"}

DBGrid is Supported by Microsoft

The versions of DBGrid that ship with Visual Basic, Visual C++, and other Microsoft development systems are supported by Microsoft. Please refer to the appropriate Microsoft product documentation for Microsoft's support and update policy.

- APEX Provides Free On-line Documentation for DBGrid
- APEX Offers Premium Support for DBGrid Users
- True DBGrid Standard Edition
- True DBGrid Pro Edition
- Other Products from APEX
- Visit APEX on the World Wide Web

APEX Provides Free On-line Documentation for DBGrid

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscAPEXProvidesFreeOnlineDocumentationForDBGridC;vbproBooksOnlineJumpTopic"}

As a service to DBGrid users, APEX provides a free on-line help file that contains a wealth of detailed information not found in the Visual Basic documentation, with accompanying tutorial projects. This complete reference is the definitive source for information on DBGrid, and is a "must-have" for any DBGrid user. To download a copy, visit the APEX web site at http://www.apexsc.com.

APEX Offers Premium Support for DBGrid Users

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscAPEXOffersPremiumSupportForDBGridUsersC;vbproBooksOnlineJumpTopic"}

For a reasonable fee, APEX will provide e-mail and phone support for DBGrid users. Premium support subscribers receive the following benefits:

- Guaranteed e-mail responses within one business day.
- Access to support technicians specifically trained to handle DBGrid inquiries.
- Regular releases of True DBGrid Standard Edition.
- Discounts for upgrading to True DBGrid Pro Edition.

Please see the APEX web site for current support policy and rate information.

True DBGrid Standard Edition

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscTrueDBGridStandardEditionC;vbproBooksOnlineJumpTopic"}

True DBGrid Standard Edition is a minor upgrade to DBGrid. It is released and supported by APEX in order to achieve the following goals:

- Add minor enhancements to DBGrid as suggested by users.
- Fix critical errors not detected during the beta cycle.
- Provide users with additional and updated documentation, tutorials, and sample code.
- Deliver frequent updates and bug fixes between releases of Visual Basic.

For a limited time, True DBGrid Standard Edition is available free of charge to all licensed Visual Basic and Visual C++ users. Thereafter, upgrades to True DBGrid Standard Edition will only be available to premium support subscribers. To download a copy of True DBGrid Standard Edition, visit the APEX web site at http://www.apexsc.com.

True DBGrid Pro Edition

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscTrueDBGridProEditionC;vbproBooksOnlineJumpTopic"}

True DBGrid Pro Edition is the enhanced, full-featured version of DBGrid and True DBGrid Standard Edition. True DBGrid Pro adds scores of powerful data presentation features that let you render your data in a wide variety of formats and styles.

Advanced features

- Style object model similar to Excel
- Styles can be applied automatically based on cell contents
- Per-cell control of fonts, colors, text formatting, and indentation
- Automatic translation from underlying data to displayed values
- In-cell graphics, combo box, check box, and radio buttons
- Dropdown editor for large fields
- Enhanced unbound modes
- Easy-to-use storage mode
- Multiple line records
- Input masking
- and much more…(see our web site for a complete list)

Create easy-to-use, visually appealing front-end applications

Why deliver an application that looks like this:

Customer Name	Company	CustType	ContactType	Callback
Greg Daryll	Networks Internation	3	0	0
Jane Lambert	Advanced Software	4	1	0
Allen Clark	Microcomputer Consu	2	0	-1
David Elkins	Software Specialists	1	2	0
Carl Ziegler	Micro Mechanics	4	1	-1

When you can deliver one that looks like this?

Customer Name	Company	CustType	How	Call?	ContactDate
Greg Daryll	Networks International	Buyer		☐	6/22/96
Jane Lambert	Advanced Software	Distributor		☐	9/5/95
Allen Clark	Microcomputer Consultants	Normal		☑	7/30/96
David Elkins	Software Specialists	Prospective		☐	6/12/96
Carl Ziegler	Micro Mechanics	Distributor		☑	11/17/93

With True DBGrid, you can easily implement intuitive displays like the one above, with very little or no coding required!

Free technical support

True DBGrid Pro users receive free technical support (e-mail, phone, and fax). The APEX technical support staff is widely regarded as one of the most responsive and knowledgeable within the ISV community. Our international distributors also provide excellent technical support for local users. Please visit our web site at http://www.apexsc.com for an up-to-date list of our U.S. and international re-sellers and distributors.

Free enhanced MSFlexGrid

For a limited time, you can receive two grids for the price of one! When you buy True DBGrid at the manufacturer's suggested retail price, you'll also receive Videosoft's VSFlex, the enhanced version of the MSFlexGrid control that ships with Visual Basic.

No hassle money back guarantee

All APEX products come with a 30-day no hassle money back guarantee.

International support and distribution

True DBGrid Pro Edition is available in English, Japanese, and German versions. For sales and distribution in Japan and Germany, please contact:

Bunka Orient Company
1-3-1 Kamo, Izumi-Ku
Sendai-City 981-31
Japan
Tel: 81-22-378-7106
Fax: 81-22-378-7170

Zoschke Data GmbH
Bahnhofstrasse 3
Schoenberg/Holstein 24217
Germany
Tel: 49-4344-6166
Fax: 49-4344-6162

Support for other languages will be available shortly. Please visit the APEX web site at http://www.apexsc.com for up-to-date information on new language versions, as well as a complete

list of U.S. and worldwide re-sellers and distributors.

Other Products from APEX

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscOtherProductsFromAPEXC;vbproBooksOnlineJumpTopic"}

MyData Control

Now you can use data-bound controls with ANY data source! MyData Control fires special Visual Basic events that let you provide **your own** data. Includes plenty of examples for the most popular third-party database libraries.

VBA Companion

More than just a better object browser, VBA Companion prints sharp-looking reference documentation for any ActiveX control or automation server, including those developed in Visual Basic. It's like having an Explorer for the objects in your system registry!

Visit APEX on the World Wide Web

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscVisitAPEXOnWorldWideWebC;vbproBooksOnlineJumpTopic"}

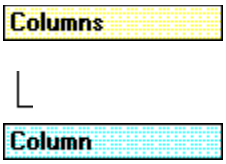
Be sure to visit the APEX web site at http://www.apexsc.com for up-to-date product information, special offers, demos, sample programs, and U.S. and international re-seller and distributor information.

APEX product specifications, support policies, prices, and special offers are subject to change without notice. Please visit our web site for the latest information, or call us at (412) 681-4343.

Column Object

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daobjColumnC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daobjColumnX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"daobjColumnP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"daobjColumnM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"daobjColumnE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daobjColumnS"}

A **Column** object represents a column within a **DBGrid** control.



Remarks

You manipulate a column in a **DBGrid** control by using a **Column** object's methods and properties. With a **Column** object, you can modify attributes of the column header as well as the column itself.

To use a **Column** object, you can either use the **Columns** property of the **DBGrid** control directly or assign each column to a separate variable dimensioned as a **Column** object. The following demonstrates the latter:

```
Dim Col1, Col2 as Column
Set Col1 = DBGrid1.Columns(0)
Set Col2 = DBGrid1.Columns(1)
Col1.Caption = "Column 1"
Col2.Caption = "Column 2"
```

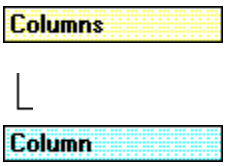
If often referring to the columns in a **DBGrid** control, you will increase performance by using the above method to assign values to columns rather than using the **Columns** property as in:

```
DBGrid1.Columns(0).Caption = "Column 1"
```

Columns Collection

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcolColumnsCollectionC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbcolColumnsCollectionX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"vaproCount;vbcolColumnsCollectionP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"vamthItem;vbcolColumnsCollectionM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"vbcolColumnsCollectionE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbcolColumnsCollectionS"}

The **Columns** collection contains all stored **Column** objects of a **DBGrid** control.



Syntax

Columns(*index***)**
Columns.Item(*index***)**

Remarks

You can use the properties and methods of the **Columns** collection to add and remove **Column** objects, count the number of columns in the **Columns** collection, and address individual columns of the **Columns** collection.

The **Columns** collection can be accessed through the **Columns** property of the **DBGrid** control.

DBGrid Control

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daobjDBGridC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daobjDBGridX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"daobjDBGridP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"daobjDBGridM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"daobjDBGridE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daobjDBGridS"}

Displays and enables data manipulation of a series of rows and columns representing records and fields from a **Recordset** object.

Syntax

DBGrid

Remarks

The data-aware **DBGrid** control appears similar to the **Grid** control; however, you can set the **DBGrid** control's **DataSource** property to a **Data** control so that the control is automatically filled and its column headers set automatically from a **Data** control's **Recordset** object. The **DBGrid** control is really a fixed collection of columns, each with an indeterminate number of rows.

Each cell of a **DBGrid** control can hold text values, but not linked or embedded objects. You can specify the current cell in code, or the user can change it at run time using the mouse or the arrow keys. Cells can be edited interactively, by typing into the cell, or programmatically. Cells can be selected individually or by row.

If a cell's text is too long to be displayed in the cell, the text wraps to the next line within the same cell. To display the wrapped text, you must increase the cell's **Column** object's **Width** property and/or the **DBGrid** control's **RowHeight** property. At design time, you can change the column width interactively by resizing the column or by changing the column's width in the **Column** object's property page.

Use the **DBGrid** control's **Columns** collection's **Count** property and the **Recordset** object's **RecordCount** property to determine the number of columns and rows in the control. A **DBGrid** control can have as many rows as the system resources can support and about 1700 columns.

When you select a cell, the **ColIndex** property is set, thus selecting one of the **Column** objects in the **DBGrid** object's **Columns** collection. The **Text** and **Value** properties of the **Column** object reference the contents of the current cell. The data in the current row can be accessed using the **Bookmark** property, which provides access to the underlying **Recordset** object's record. Each column of the **DBGrid** control has its own font, border, word wrap, color and other attributes that can be set without regard to other columns. At design time, you can set the column width and row height and establish columns that are not visible to the user. You can also prevent users from changing the formatting at run time.

Note If you set any of the **DBGrid** column properties at design time, you will need to set all of them in order to maintain the current settings.

Note If you use the **Move** method to position the **DBGrid** control, you may need to use the **Refresh** method to force it to repaint.

SelBookmarks Collection

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daobjSelBookmarksC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daobjSelBookmarksX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"daobjSelBookmarksP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"daobjSelBookmarksM;vamthItem"} {ewc HLP95EN.DLL,DYNALINK,"Events":"daobjSelBookmarksE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daobjSelBookmarksS"}

A **SelBookmarks** collection contains a bookmark for each row selected in a **DBGrid** control.

Syntax

SelBookmarks

Remarks

Use the **SelBookmarks** property of the **DBGrid** control to return the **SelBookmarks** collection. Bookmarks are added to the **SelBookmarks** collection in the order they're selected. You can reposition the **DBGrid** control's current record pointer by setting the **Bookmark** property to one of the selected bookmarks in the **SelBookmarks** collection.

Use the **Add** method to add bookmarks to the **SelBookmarks** collection. Once a bookmark is appended to the **SelBookmarks** collection, it appears selected in the **DBGrid** control.

To remove a bookmark from the **SelBookmarks** collection, use the **Remove** method. Once a bookmark is removed from the **SelBookmarks** collection, it no longer appears selected in the **DBGrid** control.

The **SelBookmarks** collection supports the **Add** and **Remove** methods as well as the **Count** property. Using these methods and properties, you can manipulate the list of selected items in the **DBGrid** control. For example, you can programmatically select additional items by using the **Add** method, or determine the total number of selected items using the **Count** property.

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproCaptionPropertyX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":""} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproCaptionPropertyS"}

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":""} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproEnabledPropertyDBGridDummyTopicS"}

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":""} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFontPropertyDBGridDummyTopicS"}

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":""} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproBackColorForeColorPropertiesDBGridControlDummyTopicS"}

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":""} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproForeColorPropertyDBGridDummyTopicS"}

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":""} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproTextPropertyColumnObjectDummyTopicS"}

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":""} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproLockedPropertyColumnObjectDummyTopicS"}

AfterColUpdate Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtAfterColUpdateC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtAfterColUpdateX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtAfterColUpdateA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtAfterColUpdateS"}

Occurs after data is moved from a cell in the **DBGrid** control to the control's copy buffer.

Syntax

Private Sub *object*_**AfterColUpdate (**[*index* **As Integer**,] *colindex* **As Integer)**

The AfterColUpdate event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
colindex	An integer that identifies the column in the control.

Remarks

When a user completes editing within a **DBGrid** control cell, as when tabbing to another column in the same row, pressing ENTER, or when the control loses focus, the BeforeColUpdate event is executed, and unless canceled, data from the cell is moved to the control's copy buffer. Once moved, the AfterColUpdate event is executed.

The AfterColUpdate event occurs after the BeforeColUpdate event, and only if the *cancel* argument in the BeforeColUpdate event is not set to **True**.

Once the AfterColUpdate event procedure begins, the cell data has already been moved to the control's copy buffer and can't be canceled, but other updates can occur before the data is committed to the **Recordset**.

AfterColUpdate Event Example

This example does a lookup when one column is updated and places the result in another column.

```
Private Sub DataGrid1_AfterColUpdate (ColIndex As Integer)
   If ColIndex = 1 Then
      Data1.Recordset.FindFirst "PubId = " _
       & DataGrid1.Columns(1).Value
      If Not Data1.Recordset.NoMatch Then
         DataGrid1.Columns(2).Value = _
          Data1.Recordset.Fields("Publisher")
      Else
         DataGrid1.Columns(2).Value = "No Match"
      End If
   End If
End Sub
```

AfterDelete Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtAfterDeleteC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtAfterDeleteX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtAfterDeleteA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtAfterDeleteS"}

Occurs after the user deletes a selected record in the **DBGrid** control.

Syntax

Private Sub *object*_**AfterDelete (**[*index* **As Integer**,] *colindex* **As Integer)**

The AfterDelete event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
colindex	An integer that identifies the column.

Remarks

When the user selects a record selector in the **DBGrid** control and presses DEL or CTRL+X, the selected row is deleted. Before the record is deleted, the BeforeDelete event is triggered. Once the row is deleted, the AfterDelete event is triggered. The row selected for deletion is available in the collection provided by the **SelBookmarks** property.

AfterDelete Event Example

This example displays a message confirming that a record has successfully been deleted.

```
Private Sub DataGrid1_AfterDelete ()
   MsgBox "Record has successfully been deleted!"
End Sub
```

AfterInsert Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtAfterInsertC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtAfterInsertX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtAfterInsertA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtAfterInsertS"}

Occurs after the user inserts a new record into the **DBGrid** control.

Syntax

Private Sub *object*_**AfterInsert (***index* **As Integer)**

The AfterInsert event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.

Remarks

When the user selects the new record (at the bottom of the control) and enters a character in one of the cells, the BeforeInsert event is triggered, followed by the BeforeUpdate, AfterUpdate and AfterInsert events.

When the AfterInsert event is triggered, the record has already been added to the database. The **Bookmark** property can be used to access the new record.

The AfterInsert event can't be canceled.

The AfterInsert event procedure can be used to update other tables or to perform post-update cleanup of other controls.

AfterInsert Event Example

This example creates an entry in a related table if the user enters a value in a column in the grid.

```
Private SubDataGrid1_AfterInsert ()
   If DataGrid1.Columns(1).Value <> "" Then
      Data2.Recordset.AddNew
      Data2.Recordset.Fields("PubId") = DataGrid1.Columns(1).Value
      Data2.Recordset.Update
   End If
End Sub
```

AfterUpdate Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtAfterUpdateC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtAfterUpdateX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtAfterUpdateA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtAfterUpdateS"}

Occurs after changed data has been written to the database from a **DBGrid** control.

Syntax

Sub *object*_**AfterUpdate (***index* **As Integer)**

The AfterUpdate event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.

Remarks

When the user moves to another row, or the **Recordset** object's **Update** method is executed, data is moved from the control's copy buffer to the **Data** control's copy buffer and written to the database. Once the write is complete, the AfterUpdate event is triggered.

The updated record is available by using the **Bookmark** property of the **DBGrid** control.

The AfterUpdate event occurs after the BeforeUpdate event, but before the LostFocus event for the control (or GotFocus for the next control in the tab order). This event occurs in bound and unbound mode and can't be canceled.

Unlike the Change event, changing data in a control or record using code doesn't trigger this event.

AfterUpdate Event Example

This example updates a label when any change has been made in the grid.

```
Private Sub DataGrid1_AfterUpdate ()
   Label1.Caption = "Last modified: " & Format$(Now, "Long Date")
End Sub
```

BeforeColUpdate Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtBeforeColUpdateC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtBeforeColUpdateX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtBeforeColUpdateA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtBeforeColUpdateS"}

Occurs after editing is completed in a cell, but before data is moved from the cell to the **DBGrid** control's copy buffer.

Syntax

Private Sub *object*_**BeforeColUpdate ([** *index* **As Integer**,] *colindex* **As Integer**, *oldvalue* **As Variant**, *cancel* **As Integer)**

The BeforeColUpdate event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
colindex	An integer that identifies the column.
oldvalue	A value that contains the value contained in the cell prior to the change.
cancel	A Boolean expression expression that specifies whether the change occurs, as described in Settings.

Settings

The settings for *cancel* are:

Setting	Description
True	Cancels the change, restores cell to *oldvalue*, and restores focus to the control.
False	(Default) Continues with change and permits change of focus.

Remarks

The data specified by the o*ldvalue* argument moves from the cell to the control's copy buffer when a user completes editing within a cell, as when tabbing to another column in the same row, pressing ENTER, or changing focus away from the cell. Before the data has been moved from the cell into the control's copy buffer, the BeforeColUpdate event is triggered. This event gives your application an opportunity to check the individual grid cells before they are committed to the control's copy buffer.

If your event procedures set the *cancel* argument to **True**, the previous value is restored in the cell and focus remains on the control and the AfterColUpdate event is not triggered.

To restore *oldvalue* in the cell and permit the user to move focus off of the cell, set *cancel* to **False** and set the cell to *oldvalue* as follows:

```
Cancel = False
DBGrid1.Columns(ColIndex).Value = OldValue
```

The AfterColUpdate event occurs after the BeforeColUpdate event.

By setting the *cancel* argument to **True**, the user can not move the focus from the control until the application determines that the data can be safely moved back to the control's copy buffer.

BeforeColUpdate Event Example

This example checks to make sure that the value the user has typed in is within a certain range; otherwise it disables the update.

```
Private Sub DataGrid1.BeforeColUpdate (ColIndex As Long, OldValue As
Variant, Cancel As Integer)
   If ColIndex = 1 Then
      If DataGrid1.Columns(1).Value < Now Then
         Cancel = True
         MsgBox "You must enter a date that is later than today."
      End If
   End If
End Sub
```

BeforeDelete Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtBeforeDeleteC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtBeforeDeleteX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtBeforeDeleteA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtBeforeDeleteS"}

Occurs before a selected record is deleted in a **DBGrid** control.

Syntax

Private Sub *object*_**BeforeDelete (**[*index* **As Integer**,] *cancel* **As Integer)**

The BeforeDelete event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
cancel	A Boolean expression that determines whether a record is deleted, as described in Settings.

Settings

The settings for *cancel* are:

Setting	Description
True	Leaves focus on control and doesn't delete the record.
False	(Default) Continues with delete operation and enables change of focus.

Remarks

When the user selects a record selector in the control and presses DEL or CTL+X, the BeforeDelete event is triggered before the selected row is deleted.

Once the row is deleted, the AfterDelete event is triggered. The row selected for deletion is available in the collection provided by the **SelBookmarks** property.

If your event procedure sets the *cancel* argument to **True**, the row isn't deleted.

If more than one row is selected, the error message `Multiple rows cannot be deleted` is displayed.

BeforeDelete Event Example

This example displays a message that asks the user to confirm a deletion in a grid.

```
Private Sub DataGrid1_BeforeDelete (Cancel As Integer)
   Dim mResult As Integer
   mResult = MsgBox("Are you sure that you want to delete " &
DataGrid1.SeletedRows &  " record?", _
    vbYesNo And vbQuestion, "Delete Confirmation")
   If mResult = vbNo Then Cancel = True
End Sub
```

BeforeInsert Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtBeforeInsertC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtBeforeInsertX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtBeforeInsertA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtBeforeInsertS"}

Occurs before new records are inserted into a **DBGrid** control.

Syntax

Private Sub *object*_**BeforeInsert (**[*index* **As Integer**,] *cancel* **As Integer)**

The BeforeInsert event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
cancel	A Boolean expression that determines if a record is added, as described in Settings.

Settings

The settings for *cancel* are:

Setting	Description
True	Leaves focus on control and doesn't add a new record
False	(Default) Continues with copy and enables change of focus

Remarks

When the user selects the new record (at the bottom of the **DBGrid** control) and enters a character in one of the cells, the BeforeInsert event is triggered, followed by the BeforeUpdate, AfterUpdate and AfterInsert events.

If your event procedure sets the *cancel* argument to **True**, the row isn't inserted and the cell is cleared.

When the BeforeInsert event is triggered, the record has not been added to the database. The new record exists in the **DBGrid** control's copy buffer until this event procedure ends.

After the AfterInsert event is finished, the new record row in the **DBGrid** control is reinitialized and the edited record becomes the last row in the **DBGrid** control.

BeforeInsert Event Example

This example displays a message that asks the user to confirm the addition of a new record.

```
Private Sub DataGrid1_BeforeInsert (Cancel As Integer)
   Dim mResult As Integer
   mResult = MsgBox("Confirm: Add a new record?", _
    vbYesNo And vbQuestion, "Confirmation")
   If mResult = vbNo Then Cancel = True
End Sub
```

BeforeUpdate Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtBeforeUpdateC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtBeforeUpdateX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtBeforeUpdateA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtBeforeUpdateS"}

Occurs before data is moved from a **DBGrid** control to the control's copy buffer.

Syntax

Private Sub *object*_**BeforeUpdate ([***index* **As Integer**,] *cancel* **As Integer)**

The BeforeUpdate event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
cancel	A Boolean expression that determines if data is copied, as described in Settings.

Settings

The settings for *cancel* are:

Setting	Description
True	Leaves focus on control and doesn't copy data.
False	(Default) Continues with copy operation and enables change of focus.

Remarks

When the user moves to another row or the **Recordset** object's **Update** method is executed, data is moved from the **DBGrid** control's copy buffer to the **Data** control's copy buffer and written to the database.

Just before the data is moved from the **DBGrid** control's copy buffer back into the **Data** control's copy buffer, the BeforeUpdate event is triggered. Unless the copy operation is canceled, the AfterUpdate event is triggered after the data has been moved back into the **Data** control's copy buffer and written to the database. The updated record is available by using the **Bookmark** property of the **DBGrid** control.

If you set the BeforeUpdate event *cancel* argument to **True**, focus remains on the control, neither the AfterUpdate or LostFocus event is triggered, and the record isn't saved to the database.

The BeforeUpdate event occurs before the AfterUpdate and LostFocus events for this control, or before the GotFocus event for the next control in the tab order.

This event occurs even if the control isn't bound.

Unlike the Change event, changing data in a control or record using code doesn't trigger this event.

You can use this event to validate data in a bound control record before permitting the user to commit the change to the **Data** control's copy buffer. By setting the *cancel* argument to **True**, the user can't move focus from the control until the application determines whether the data can be safely moved back to the **Data** control's copy buffer.

BeforeUpdate Event Example

This example displays a message that tells the user to enter a value in the first column before the grid can be updated.

```
Private Sub DataGrid1_BeforeUpdate (Cancel As Integer)
   If DataGrid1.Columns(1).Value = "" Then
      MsgBox "You must enter value in the first column!"
      Cancel = True
   End If
End Sub
```

ColResize Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtColResizeC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtColResizeX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtColResizeA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtColResizeS"}

Occurs when a user resizes a column of a **DBGrid** control.

Syntax

Private Sub *object*_**ColResize (**[*index* **As Integer**,] *colindex* **As Integer**, *cancel* **As Integer)**

The ColResize event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
colindex	An integer that identifies the column.
cancel	A Boolean expression that determines whether a column is resized, as described in Settings.

Settings

The settings for *cancel* are:

Setting	Description
True	Cancels the change, restores column to its original width.
False	(Default) Continues with width change.

Remarks

When the user resizes a column, the ColResize event is triggered. Your event procedure can accept the change, alter the degree of change, or cancel the change completely.

If you set the *cancel* argument to **True**, the column width is restored. To alter the degree of change, set the **Width** property of the **Column** object to the desired value.

Executing the **Refresh** method within the procedure causes the control to be repainted even if the *cancel* argument is **True**.

ColResize Event Example

This example resizes all the columns to the size of the first column if the user sizes the first column.

```
Private Sub DataGrid1_ColResize (ColIndex As Integer, Cancel As Integer)
   Dim nCol As Column
   If ColIndex = 1 Then
      For Each nCol In DataGrid1.Columns
         nCol.Width = DataGrid.Columns(1).Width
      Next
   End If
End Sub
```

HeadClick Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtHeadClickC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtHeadClickX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtHeadClickA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtHeadClickS"}

Occurs when the user clicks on the header for a particular column of a **DBGrid** control.

Syntax

Private Sub *object*_**HeadClick (**[*index* **As Integer**,] *colindex* **As Integer)**

The HeadClick event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
colindex	An integer that identifies the column.

Remarks

One possible use for this event is to resort the **Recordset** object based on the selected column.

HeadClick Event Example

This example sorts the record source of the **Data** control based on which column the user clicked.

```
Private Sub DataGrid1_HeadClick (ColIndex As Integer)
   Data1.RecordSource = "Select * From Publishers Order By " & _
    DataGrid1.Columns(ColIndex).DataField
   Data1.Refresh
End Sub
```

RowResize Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daevtRowResizeC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daevtRowResizeX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daevtRowResizeA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daevtRowResizeS"}

Occurs when a user resizes a row in a **DBGrid** control.

Syntax

Private Sub *object*_**RowResize (**[*index* **As Integer**,] *cancel* **As Integer)**

The RowResize event syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	An integer that identifies a control if it is in a control array.
cancel	A Boolean expression that determines if a change is canceled, as described in Settings.

Settings

The settings for *cancel* are:

Setting	Description
True	Cancels the change and restores row to its original height.
False	(Default) Continues with change of row height.

Remarks

The user can resize the **DBGrid** control rows using the mouse. When the user changes the height, the RowResize event is triggered. Your event procedure can accept the change, alter the degree of change, or cancel the change completely.

The **DBGrid** control's **RowHeight** property determines the height for all rows in the control.

If you set the *cancel* argument to **True**, the row height is restored. To alter the degree of change, set the **RowHeight** property to the desired value.

Executing the **Refresh** method within the procedure causes the control to be repainted even if the *cancel* argument is **True**.

RowResize Event Example

This example ensures that there are at least five visible rows in the grid.

```
Private Sub DataGrid1_RowResize (Cancel As Integer)
   If DataGrid1.VisibleRows < 5 Then Cancel = True
End Sub
```

CellText Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"damthCellTextC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"damthCellTextX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"damthCellTextA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"damthCellTextS"}

Returns a formatted text value from a **DBGrid** control cell. Doesn't support named arguments.

Syntax

*object***.CellText** *bookmark*

The **CellText** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
bookmark	Required. A string expression that represents a row in the **DBGrid** control.

Remarks

The **CellText** method returns a formatted string representation of the data in the current column for the row specified by the *bookmark* value. Using the **CellText** method is similar to accessing the **Text** property, except you can select a specific row from which to retrieve the value.

The value returned by the **CellText** method is derived from the **Text** property by applying the formatting as specified by the **NumberFormat** property of the **Column** object.

When using the **CellText** method, use the **Columns** collection to specify the specific column of the **DBGrid** control and set the *bookmark* argument to a specific row.

Using the **CellText** method to extract information from a cell doesn't affect the current selection.

CellText Method Example

This examples gets information from the top and bottom rows and displays it in a label.

```
Sub DBGrid1_Scroll (Cancel As Integer)
   Dim TopRow, BottomRow
   TopRow = DBGrid1.Columns(1).CellText(DBGrid1.FirstRow)
   BottomRow = DBGrid1.Columns(1).CellText(DBGrid1.RowBookmark _
    (DBGrid1.VisibleRows - 1))
   Label1.Caption = "Records " & TopRow & " to " & _
    BottomRow & " are currently displayed."
End Sub
```

CellValue Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"damthCellValueC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"damthCellValueX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"damthCellValueA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"damthCellValueS"}

Returns a raw data value in a column for a specified row in a **DBGrid** control. Doesn't support named arguments.

Syntax

object.**CellValue** *bookmark*

The **CellValue** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
bookmark	Required. A string expression that contains the unformatted data stored in a selected **DBGrid** control cell.

Remarks

When using the **CellValue** method, use the **Columns** collection to specify the specific column of the **DBGrid** control and set the *bookmark* argument to a specific row.

Using the **CellValue** method returns the same value as the **Value** property setting of the current **Column** object, except that you can specify a specific row in the **DBGrid** control to reference.

Using the **CellValue** method to extract information from a cell doesn't affect the current selection.

CellValue Method Example

This example retrieves all the values in a given column from the selected range of rows and loads them into an array for later use.

```
Sub Command1_Click ()
   Dim I
   ReDim CalcArray (0 to DBGrid1.SelBookmarks.Count - 1)
   For I = 0 to DBGrid1.SelBookmarks.Count -1
   ' Puts the value of the current row in the selected row
   ' array into corresponding CalcArray cell.
      CalcArray(I) = _
       DBGrid1.Columns(1).CellValue(DBGrid1.SelBookmarks(I))
   Next I
End Sub
```

ColContaining Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"damthColContainingC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"damthRowColContainingX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"damthColContainingA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"damthColContainingS"}

Returns the **ColIndex** value of the **DBGrid** control column containing the specified coordinate (X) value. Doesn't support named arguments.

Syntax

object.**ColContaining** *coordinate*

The **ColContaining** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
coordinate	Required. A single numeric expression that defines a horizontal coordinate (X value) based on the coordinate system of the container.

Remarks

The **ColContaining** method returns a number that corresponds to one of the column indexes of the control specified by *object*. This number ranges from 0 to 1 less than the setting of the **Count** property of the **Columns** collection (0 to **Columns.Count** - 1). This method is useful when working with mouse and drag events when you are trying to determine where the user clicked or dropped another control in terms of a column of the **DBGrid** control.

If *coordinate* is outside of the coordinate system of the container, a trappable error occurs.

Note The **ColContaining** method returns the **ColIndex** of the column indicated, not the visible column. If *coordinate* falls in the first visible column, but two columns have been scrolled off the left side of the control, the **ColContaining** method returns 2.

GetBookmark Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"damthGetBookmarkC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"damthGetBookmarkX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"damthGetBookmarkA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"damthGetBookmarkS"}

Returns a value containing a bookmark for a row relative to the current row in a **DBGrid** control. Doesn't support named arguments.

Syntax

object.**GetBookmark** *value*

The **GetBookmark** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	Required. A long numeric expression that addresses rows of the **DBGrid** control relative to the current row, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
0	Returns bookmark of the current row — the same as `DBGrid1.Bookmark`.
1	Returns bookmark of the row following current row.
-1	Returns bookmark of the row preceding current row.
n	Returns bookmark of the row relative to current row based on (`DBGrid1.Row + n`)

Remarks

The **GetBookmark** method may return values that are very different from the **RowBookmark** method because the current row may not be visible.

GetBookmark Method Example

This example checks the updated value of a particular column to make sure that the new value lies between the values of the previous and the next rows.

```
Sub DBGrid1_BeforeColUpdate (ColIndex As Integer, _
 OldValue as Variant, PrevVal, NextVal, CurVal, Cancel As Integer)
   If ColIndex = 1 Then
      PrevVal = DBGrid1.Columns(1).CellValue(_
       DBGrid1.GetBookmark(-1))
      NextVal = DBGrid1.Columns(1).CellValue(_
       DBGrid1.GetBookmark(1))
      CurVal = DBGrid1.Columns(1).Value
      If CurVal > PrevVal Or CurVal < NextVal  Then
         Cancel = True
         MsgBox "Value must be between" & PrevVal _
          & " and " & NextVal
      End If
   End If
End Sub
```

Add Method (Columns, SelBookmarks, Splits Collections)

Adds a new column to the **Columns** collection, a new bookmark to the **SelBookmarks** collection, or a new split to the **Splits** collection of the **DBGrid** control. Doesn't support named arguments.

Syntax

*object***.Add** *colindex*
*object***.Add** *bookmark*

The **Add** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
colindex	Required. An integer that specifies where the new **Column** or **Split** object is inserted in the **Columns** collection or **Splits** collection, as described in Settings.
bookmark	The bookmark to be added to the collection.

Settings

The settings for *colindex* are:

Setting	Description
0	Inserts new column as leftmost column.
Count	If the *colindex* argument is the same as the **Count** property setting, the new column is inserted as the rightmost column.
n	Inserts the new column to the left of the nth column in the **Columns** collection. The *n*th column and all subsequent columns are incremented accordingly.

Remarks

The **Add** method inserts a new **Column** object into the **Columns** collection based on the *colindex* argument. New columns are added with their **Visible** property set to **False** and all other properties set to their default values. Initially, new columns are unbound because the **DataField** property is set to a zero-length string ("""). The **Count** property of the **Columns** collection is incremented to reflect the new column.

Important If you have previously deleted a column using the **Remove** method, after adding new columns, you may need to refresh the display with the **Rebind** and **Refresh** methods. This instructs the **DBGrid** control to rebuild its internal column layout matrix to correctly reflect the true status of the control.

Use the **Add** method to add bookmarks to the **SelBookmarks** collection. Once a bookmark is appended to the **SelBookmarks** collection, it appears selected in the **DBGrid** control.

Rebind Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"damthRebindC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"damthRebindX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"damthRebindA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"damthRebindS"}

Re-creates the **DBGrid** control properties and columns. Doesn't support named arguments.

Syntax

object.**Rebind**

Remarks

The **Rebind** method causes the **DBGrid** control to perform the same operations that occur when you set the **DataSource** property. The **DBGrid** control resets the columns, headings and other properties based on the current **Data** control properties.

If you have not modified the grid columns at design time, then executing the **ReBind** method will reset the columns, headings, and other properties based on the current data source.

However, if you have altered the columns in any way at design time (even if you leave the **DataField** properties blank), then the grid will assume that you wish to maintain the modified grid layout and will not automatically reset the columns.

For an unbound grid (one with its **DataMode** property set to 1), this method is similar to the **Refresh** method except that the grid attempts to restore the current and topmost rows.

Note To force the grid to reset the column bindings even if the columns were modified at design time, invoke the **ClearFields** method immediately before **ReBind**. Conversely, to cancel the grid's automatic layout response and force the grid to use the current column/field layout, invoke the **HoldFields** method immediately before **ReBind**.

Rebind Method Example

This example checks a global variable to see if the user changed the table layout and reconfigures using the original table information.

```
Sub CheckForRebind_Click ()
   If UserChangedLayout Then
      DBGrid1.Rebind
   End If
End Sub
```

Remove Method (DBGrid)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"damthRemoveC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"damthRemoveX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"damthRemoveA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"damthRemoveS"}

Removes the specified row from the **SelBookmarks** collection, or the specified **Column** object from the **Columns** collection of a **DBGrid** control.

Syntax

object.**Remove** *index*

The **Remove** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
index	Required. An integer in the range of 0 to the **Count** property setting of the collection – 1.

Remarks

For the **SelBookmarks** collection, the **Remove** method removes the row specified by the *index* argument, then decrements the **SelBookmarks.Count** property by 1. If the row removed from the **SelBookmarks** collection is visible, it will be deselected in the **DBGrid** control.

For the **Columns** collection, the **Remove** method removes the column specified by the *index* argument, then decrements the **Columns.Count** property by 1.

If you specify a row that isn't in the **SelBookmarks** collection or a **Column** object that isn't in the **Columns** collection, a trappable error occurs.

RowBookmark Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"damthRowBookmarkC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"damthRowBookmarkX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"damthRowBookmarkA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"damthRowBookmarkS"}

Returns a value containing a bookmark for a visible row in the **DBGrid** control. Doesn't support named arguments.

Syntax

object.**RowBookmark** *value*

The **RowBookmark** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	Required. An integer in the range of 0 to the setting of the **DBGrid** control's **VisibleRows** property minus 1.

Remarks

RowBookmark(0) returns the same bookmark as the **FirstRow** property of the **DBGrid** control. The current row, as determined by the **DBGrid** control's **Bookmark** property, may not be returned by this method if the current row isn't visible.

Note The bookmarks returned by **RowBookmark** should not be saved because their values change as soon as rows visible in the **DBGrid** control change.

RowBookmark Method Example

This example selects all the rows that are currently visible on the grid.

```
Sub SelectAllVisible_Click ()
   Dim I
   For I = 0 To DBGrid1.VisibleRows - 1
      DBGrid1.SelBookmarks.Add DBGrid1.RowBookmark(I)
   Next I
End Sub
```

RowContaining Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"damthRowContainingC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"damthRowColContainingX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"damthRowContainingA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"damthRowContainingS"}

Returns a value corresponding to the row number of the specified vertical (Y) coordinate of the **DBGrid** control. Doesn't support named arguments.

Syntax

object.**RowContaining** *coordinate*

The **RowContaining** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
coordinate	Required. A single numeric expression that specifies a vertical coordinate (Y value) based on the coordinate system of the container.

Remarks

The **RowContaining** method returns a value that corresponds to one of the column indexes of the control specified by *object*. This value ranges from 0 to the setting of the **VisibleRows** property -1. This method is useful when working with mouse and drag events when you are trying to determine where the user clicked or dropped another control in terms of a row of the **DBGrid** control.

If *coordinate* is outside of the coordinate system of the container, a trappable error occurs.

RowContaining, ColContaining Method Example

This example saves the value of the cell where the user began a drag method.

```
Dim SaveValue
Sub DBGrid1_MouseDown (Button As Integer, Shift As Integer, _
 X As Single, Y As Single)
   Dim RowValue, ColValue
   ' Get the value of the row and column that the mouse is over
   RowValue = DBGrid1.RowContaining(Y)
   ColValue = DBGrid1.ColContaining(X)
   ' If the values are both valid, save the text of the cell and
   ' begin dragging.
   If RowValue > 0 And RowValue < DBGrid1.VisibleRows And _
    ColValue > 0 And ColValue < DBGrid1.VisibleCols Then
      SaveValue = DBGrid1.Columns(ColValue). _
      CellValue(DBGrid1.RowBookmark(RowValue))
      DBGrid1.Drag 1
   End If
End Sub
```

RowTop Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"damthRowTopC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"damthRowTopX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"damthRowTopA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"damthRowTopS"}

Returns a value containing the Y coordinate of the top of a specified row of a **DBGrid** control. Doesn't support named arguments.

Syntax

object.**RowTop** *value*

The **RowTop** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	Required. An integer that specifies a row in the range of 0 to the setting of the **VisibleRows** property -1.

Remarks

The **RowTop** method returns a value that corresponds to the Y coordinate of the top of the row specified by *value*. The return value is based on the **ScaleMode** property of the container.

You can use the **RowTop** method with the **RowHeight**, **Left**, and **Width** properties of the **Column** object to determine the exact location and dimension of a chosen cell in the **DBGrid** control.

RowTop Method Example

This example begins a drag operation in the grid. Using the grid cell location and size properties, a **Label** control the size of the cell is used as the drag object.

```
Sub DBGrid1_MouseDown (Button As Integer, _
Shift As Integer, X As Single, Y As Single)
   ' Declare variables.
   Dim DY, DX, RowValue, ColValue, CellLeft, CellTop
   ColValue = DBGrid1.ColContaining(X)
   RowValue = DBGrid1.RowContaining(Y)
   ' Get the height of the cell.
   DY = DBGrid1.RowHeight
   ' Get the width of the cell.
   DX = DBGrid1.Columns(ColValue).Width
   CellLeft = DBGrid1.Columns(ColValue).Left
   CellTop = DBGrid1.RowTop(RowValue)
   Label1.Caption = DBGrid1.Columns(ColValue). _
      CellValue(DBGrid1.RowBookmark(RowValue))
   Label1.Move CellLeft, CellTop, DX, DY
   Label1.Drag ' Drag label outline.
End Sub
```

Scroll Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"damthScrollC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"damthScrollX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"damthScrollA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"damthScrollS"}

Scrolls the **DBGrid** control horizontally and vertically in a single operation. Doesn't support named arguments.

Syntax

object.**Scroll** *colvalue, rowvalue*

The **Scroll** method syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
colvalue	Required. A long numeric expression that specifies a column in the control.
rowvalue	Required. A long numeric expression that specifies a row in the control.

Remarks

Positive values scroll right and down. Negative values scroll left and up. Values that are out of range don't cause an error — the **DBGrid** control scrolls to the maximum degree possible. The same effect can be achieved by setting the **FirstRow** and **LeftCol** properties, but these must be set independently, causing two separate Paint events.

Scroll Method Example

This example creates two buttons that enable you to scroll diagonally, one to move down and to the right and the other to move up and to the left.

```
Sub ScrollDownRight_Click
   ' Scroll down and to the right.
   DBGrid1.Scroll DBGrid1.VisibleCols, DBGrid1.VisibleRows
End Sub

Sub ScrollUpLeft_Click
   ' Scroll up and to the left.
   DBGrid1.Scroll -DBGrid1.VisibleCols, -DBGrid1.VisibleRows
End Sub
```

AllowAddNew Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproAllowAddNewC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproAllowAddNewX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproAllowAddNewA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproAllowAddNewS"}

Returns or sets a value indicating whether the user can add new records to the **Recordset** object underlying a **DBGrid** control.

Syntax

object.**AllowAddNew** [**=** *value*]

The **AllowAddNew** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines whether a user can add new records, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	Users can add records to the **Recordset** object underlying the **DBGrid** control.
False	Users can't add records to the **Recordset** underlying the **DBGrid** control.

Remarks

If the **AllowAddnew** property is **True**, the last row displayed in the **DBGrid** control is left blank to permit users to enter new records. If the **AllowAddNew** property is **False**, no blank line is displayed.

The underlying **Recordset** may not enable insertions even if the **AllowAddNew** property is **True**. In this case, an error occurs when the user tries to add a record.

AllowAddNew, AllowDelete, AllowUpdate Properties Example

This example checks the value of a check box. If it is **False**, the user can't make changes to the grid.

```
Private Sub Form_Load ()
   If Check1.Value = 0 Then
      DBGrid1.AllowDelete = False
      DBGrid1.AllowAddNew = False
      DBGrid1.AllowUpdate = False
   End If
End Sub
```

AllowDelete Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproAllowDeleteC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproAllowDeleteX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproAllowDeleteA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproAllowDeleteS"}

Returns or sets a value indicating whether the user can delete records from the **Recordset** object underlying a **DBGrid** control.

Syntax

object.**AllowDelete** [**=** *value*]

The **AllowDelete** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines whether a user can delete records, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	Users can delete records from the **Recordset** object underlying the **DBGrid** control.
False	Users can't delete records from the **Recordset** underlying the **DBGrid** control.

Remarks

Use the **AllowDelete** property to prevent the user from deleting records from the **Recordset** through interaction with the **DBGrid** control.

The underlying **Recordset** may not enable deletions even if the **AllowDelete** property is **True** for the **DBGrid** control. In this case, an error occurs when the user tries to delete a record.

AllowSizing Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproAllowSizingC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproAllowSizingX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproAllowSizingA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproAllowSizingS"}

Returns or sets a value indicating whether a user can resize columns or splits in the **DBGrid** control at run-time.

Syntax

object.**AllowSizing** [**=** *value*]

The **AllowSizing** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines whether a column or split can be resized, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	(Default for **Column**) User can resize column or split.
False	(Default for **Split**) User can't resize column or split.

Remarks

If the **AllowSizing** property is **True**, the mouse pointer turns into a double-headed (Size W E) arrow when positioned over the divider of the specified column, and the user can resize the column by dragging. Any change in column size causes a ColResize event.

For the leftmost split with **AllowSizing** set to **True**, the mouse pointer turns into a pair of vertical lines with a downward arrow when positioned over that split's size box (at the lower left corner), and the user can create a new split by dragging. The creation of a new split causes a SplitChange event.

If **AllowSizing** is **True** for any other split, the mouse pointer turns into a pair of vertical lines with a double-headed arrow when positioned over that split's size box, and the user can resize the split by dragging. No event is fired in this case (except for the standard mouse events).

AllowSizing Property Example

This example prevents the user from resizing or editing the first three columns of the grid.

```
Private Sub Form_Load ()
   Dim I
   For I = 0 to 2
      DBGrid1.Columns(I).AllowSizing = False
      DBGrid1.Columns(I).Locked = True
   Next I
End Sub
```

AllowRowSizing Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproAllowRowSizingC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproAllowRowSizingX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproAllowRowSizingA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproAllowRowSizingS"}

Returns or sets a value indicating whether a user can resize the rows of the **DBGrid** control or **Split** object at run-time.

Syntax

object.**AllowRowSizing** [**=** *value*]

The **AllowRowSizing** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines whether a user can resize rows, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	Rows can be sized by the user.
False	Rows can't be sized by the user.

Remarks

If the **AllowRowSizing** property is **True**, the mouse pointer turns into a double-headed (Size N S) arrow when positioned over the row divider between any record selectors, and the user can resize the rows by dragging. Any change in row size causes a RowResize event.

All rows of the **DBGrid** control are always the same height, which is determined by the **RowHeight** property.

Note Even if the **AllowRowSizing** property is **False**, the height of the rows can still be changed programmatically with the **RowHeight** property.

AllowRowSizing Property Example

This example checks the database to see if it has any memo fields; if not, row resizing is disabled.

```
Sub CheckForMemoField()
   Dim Fld As Field
   DBGrid1.AllowRowSizing = False
   For Each Fld in Data1.Recordset.Fields
      If Fld.Type = dbMemo Then
         DBGrid1.AllowRowSizing = True
         DBGrid1.RowHeight = DBGrid1.RowHeight * 2
         Exit For
      End If
   Next
End Sub
```

AllowUpdate Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproAllowUpdateC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproAllowUpdateX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproAllowUpdateA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproAllowUpdateS"}

Returns or sets a value indicating whether a user can modify any data in the **DBGrid** control.

Syntax

object.**AllowUpdate** [**=** *value*]

The **AllowUpdate** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines whether the user can change data, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	The user can modify data in the **DBGrid** control
False	The user can't modify data in the **DBGrid** control

Remarks

When the **AllowUpdate** property is **False**, the user can still scroll through the **DBGrid** control and select data, but can't change any of the values; any attempt to change the data in the grid is ignored.

You can also use the **Column** object properties to make individual columns of the **DBGrid** control read-only, but the **AllowUpdate** property setting takes precedence over the column settings (without changing the column settings).

Note The **Recordset** object may not enable updates even if **AllowUpdate** is **True** for the **DBGrid** control; in this case a trappable error occurs when the user tries to change the record.

ColIndex Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproColIndexC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproColIndexX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproColIndexA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproColIndexS"}

Returns a value indicating the position of the column in the **Columns** collection of the **DBGrid** control and the visible position (left-to-right) of the column in the **DBGrid** control. This property is read-only at run time and not available at design time.

Syntax

object.**ColIndex** [**=** *value*]

The **ColIndex** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	An integer ranging from 0 (the column furthest to the left) to the setting of the **Count** property.

Remarks

This property returns the zero-based index of a column within the **Columns** collection.

ColumnHeaders Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproColumnHeadersC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproColumnHeadersX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproColumnHeadersA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproColumnHeadersS"}

Returns or sets a value indicating whether the column headers are displayed in a **DBGrid** control.

Syntax

object.**ColumnHeaders** [**=** *value*]

The **ColumnHeaders** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines whether column headers are displayed, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	**DBGrid** control's column headers are displayed
False	**DBGrid** control's column headers aren't displayed

Columns Property (DBGrid)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproColumnsC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproColumnsX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproColumnsA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproColumnsS"}

Returns a collection of **Column** objects.

Syntax

object.**Columns**

The *object* placeholder represents an object expression that evaluates to an object in the Applies To list.

Remarks

The **Columns** property returns a collection of **Column** objects in a **Variant**.

You can manipulate most of a **DBGrid** control's attributes by changing the properties of **Column** objects. Choose a specific **Column** object with the **Col** property.

DataChanged Property (DBGrid)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproDataChangedPropertyDBGridC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDataChangedPropertyDBGridX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDataChangedPropertyDBGridA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDataChangedPropertyDBGridS"}

Returns or sets a value indicating that the data in the bound control has been changed by some process other than that of retrieving data from the current record. Not available at design time.

Syntax

object.**DataChanged** [= *value*]

The **DataChanged** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that indicates whether data has changed, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	The data currently in the control isn't the same as in the current record.
False	(Default) The data currently in the control, if any, is the same as the data in the current record.

Remarks

When a **Data** control moves from record to record, it passes data from fields in the current record to controls bound to the specific field or the entire record. As data is displayed in the bound controls, the **DataChanged** property is set to **False**. If the user or any other operation changes the value in the bound control, the **DataChanged** property is set to **True**. Simply moving to another record doesn't affect the **DataChanged** property.

When the **Data** control starts to move to a different record, the Validate event occurs. If **DataChanged** is **True** for any bound control, the **Data** control automatically invokes the **Edit** and **Update** methods to post the changes to the database.

If you don't wish to save changes from a bound control to the database, you can set the **DataChanged** property to **False** in the Validate event.

Inspect the value of the **DataChanged** property in your code for a control's Change event to avoid a cascading event. This applies to both bound and unbound controls.

Data Type

Integer (Boolean)

DataField Property (DBGrid Control, Column Object)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproDataFieldPropertyDBGridControlColumnObjectC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDataFieldPropertyDBGridControlColumnObjectX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDataFieldPropertyDBGridControlColumnObjectA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDataFieldPropertyDBGridControlColumnObjectS"}

Returns or sets a value that binds a control to a field in the current record

Syntax

object.**DataField** [**=** *value*]

The **DataField** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A string expression that evaluates to the name of one of the fields in the **Recordset** object specified by a **Data** control's **RecordSource** and **DatabaseName** properties.

Remarks

Bound controls provide access to specific data in your database. Bound controls that manage a single field typically display the value of a specific field in the current record. The **DataSource** property of a bound control specifies a valid **Data** control name, and the **DataField** property specifies a valid field name in the **Recordset** object created by the **Data** control. Together, these properties specify what data appears in the bound control.

When you use a **QueryDef** object or SQL statement that returns the results of an expression, the field name is automatically generated by the Microsoft Jet database engine. For example, when you code an SQL aggregate function or an expression in your SQL query, unless you alias the aggregate fields using an AS clause, the field names are automatically generated. Generally, the expression field name is Expr1 followed by a three-character number starting with 000. The first expression returned would be named Expr1000.

It's recommended that you code your SQL queries to alias expression columns as shown below:

```
Data1.RecordSource = "Select AVG(Sales)  " _
   & " AS AverageSales From SalesTable"
Text1.DataField = "AverageSales"
Data1.Refresh
```

Note Make sure the **DataField** property setting is valid for each bound control. If you change the setting of a **Data** control's **RecordSource** property and then use **Refresh**, the **Recordset** identifies the new object. This may invalidate the **DataField** settings of bound controls and produce a trappable error.

Data Type

String

DefColWidth Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproDefColWidthC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproDefColWidthX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproDefColWidthA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproDefColWidthS"}

Returns or sets a value indicating the default column width for all columns in the **DBGrid** control.

Syntax

object.**DefColWidth** [**=** *value*]

The **DefColWidth** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	An integer based on the scale mode of the control.

Remarks

If you set the **DefColWidth** property to 0, the control automatically sizes all columns based on either the width of the column heading or the **Size** property setting of the underlying field, whichever is larger. For example, to set the default column width of all columns to the width of the first column:

```
DBGrid1.DefColWidth = DBGrid1.Columns(0).Width
```

DividerStyle Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproDividerStyleC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproDividerStyleX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproDividerStyleA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproDividerStyleS"}

Returns or sets a value specifying the style of border drawn on the right edge of the specified column of a **DBGrid** control.

Syntax

object.**DividerStyle** [**=** *value*]

The **DividerStyle** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	An integer that specifies the border style, as described in Settings.

Settings

The settings for *value* are:

Constant	Value	Description
dbgNoDividers	0	No divider
dbgBlackLine	1	Black line
dbgDarkGrayLine	2	(Default) Dark gray line
dbgRaised	3	Raised
dbgInset	4	Inset
dbgUserForeColor	5	Divider is drawn using the color set by the **ForeColor** property
dbgLightGrayLine	6	Light gray line

Remarks

The **DividerStyle** property doesn't affect whether the column can be resized by dragging it or not. When the border is either raised or inset, the colors used are set by Microsoft Windows.

DividerStyle Property Example

This example changes the column divider style when the user clicks a heading.

```
Private Sub DBGrid1_HeadClick (ColIndex As Long)
   If DBGrid1.Columns(ColIndex).DividerStyle <> 5 Then
      DBGrid1.Columns(ColIndex).DividerStyle = _
       DBGrid1.Columns(ColIndex).DividerStyle + 1
   Else
      DBGrid1.Columns(ColIndex).DividerStyle = 0
   End If
End Sub
```

FirstRow Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproFirstRowC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproFirstRowX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproFirstRowA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproFirstRowS"}

Returns or sets a value containing the bookmark for the first visible row in the **DBGrid** control or **Split** object. Not available at design time.

Syntax

object.**FirstRow** [**=** *value*]

The **FirstRow** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A string expression containing a bookmark corresponding to the first visible row in the **DBGrid** control.

Remarks

For a **DBGrid** control, setting the **FirstRow** property causes the grid to scroll so that the specified row becomes the topmost row. If a grid contains multiple splits, then the topmost row changes in each split, even if the splits have different **ScrollGroup** property settings.

For a **Split** object, setting the **FirstRow** property causes the specified row to become the topmost row for that split only.

HeadFont Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproHeadFontC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproHeadFontX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproHeadFontA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproHeadFontS"}

Returns or sets a value indicating the font used in column headers in a **DBGrid** control.

Syntax

object.**Type** [= *value*]

The **Type** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	An object expression that evaluates to a **Font** object. The default value is the column's current font set to bold.

Remarks

Changing the **HeadFont** property may resize the headers to accommodate the new font.

HeadLines Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproHeadLinesC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproHeadLinesX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproHeadLinesA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproHeadLinesS"}

Returns or sets a value indicating the number of lines of text displayed in the column headers of a **DBGrid** control.

Syntax

object.**HeadLines** [**=** *value*]

The **HeadLines** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A single from 0 to 10. The default value is 1, which causes the control to display the names of underlying fields for each column in the header. A setting of 0 removes the headings.

Remarks

The **HeadLines** property can be used to display more than one line of text in the column headers of the **DBGrid** control.

HeadLines Property Example

This example checks the value of a check box to determine whether or not to display headings in the grid.

```
Private Sub Check1_Click ()
   If Check1.Value = vbChecked Then
      DBGrid1.HeadLines = 2  ' If checked,two lines in
                             ' column headings.
   Else
      DBGrid1.HeadLines = 0  ' No headings.
   End If
End Sub
```

NumberFormat Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproNumberFormatC;vafctFormat;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproNumberFormatX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproNumberFormatA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproNumberFormatS"}

Returns or sets a value indicating the format string for the **Column** object of a **DBGrid** control.

Syntax

object.**NumberFormat** [**=** *value*]

The **NumberFormat** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A string expression that defines how the expression in the **Value** property is formatted. The default value is a zero-length string (""").

Remarks

The **Text** property of the **Column** object is derived by applying this format to the **Value** property of the **Column** object. If **NumberFormat** is set to an invalid string, data in the cells are displayed as #ERR# and the value set in the **Value** property remains unchanged. See the **Format** function for information about valid format strings.

NumberFormat Property Example

This example formats the second column in a **DBGrid** control as a *long date*:

```
Private Sub Command1_Click ()
   DBGrid1.Columns(1).NumberFormat = "long date"
End Sub
```

RecordSelectors Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproRecordSelectorsC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproRecordSelectorsX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproRecordSelectorsA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproRecordSelectorsS"}

Returns or sets a value indicating if record selectors are displayed in the **DBGrid** control or **Split** object.

Syntax

object.**RecordSelectors** [**=** *value*]

The **RecordSelectors** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Boolean expression that determines if record selectors are displayed, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
True	Record selectors are displayed.
False	Record selectors aren't displayed.

Remarks

Record selectors, when displayed, appear to the left of the rows in the grid or split.

If a grid contains multiple splits, then setting its **RecordSelectors** property has the same effect as setting the **RecordSelectors** property of each split individually.

Note When the user selects a row by clicking its record selector, the bookmark of the selected row is added to the **SelBookmarks** collection.

RowDividerStyle Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproRowDividerStyleC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproRowDividerStyleX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproRowDividerStyleA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproRowDividerStyleS"}

Returns or sets a value that specifies the style of border drawn between the rows of the selected **DBGrid** control.

Syntax

object.**RowDividerStyle** [**=** *value*]

The **RowDividerStyle** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	An integer that specifies the border style, as described in Settings.

Settings

The settings for *value* are:

Setting	Description
0	No divider
1	Black line
2	(Default) Dark gray line
3	Raised
4	Inset
5	Divider is drawn using the **ForeColor** property setting color

Remarks

The **RowDividerStyle** property doesn't affect whether the border can be dragged or not. When the border is raised or inset, Microsoft Windows sets the colors.

RowDividerStyle Property Example

This example toggles through the different row line styles when you click a command button.

```
Private Sub ChangeStyle_Click ()
   If DBGrid1.RowDividerStyle < 5 Then
      DBGrid1.RowDividerStyle = DBGrid1.RowDividerStyle + 1
   Else
      DBGrid1.RowDividerStyle = 0
   End If
End Sub
```

SelBookmarks Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproSelBookmarksC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproSelBookmarksX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproSelBookmarksA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproSelBookmarksS"}

Returns a collection of bookmarks for all selected records in the **DBGrid** control.

Syntax

object.**SelBookmarks**

The *object* placeholder represents an object expression that evaluates to an object in the Applies To list.

Remarks

When a record is selected in the **DBGrid** control, its bookmark is appended to the collection returned by the **SelBookmarks** property. For example, if you create a clone of the **Recordset** object created by the **DBGrid** control, you can process individual data records by repositioning the cloned **Recordset** with bookmarks taken from the **SelBookmarks** collection.

SelBookmarks Property Example

This example loops through the rows the user has selected and deletes them from the database.

```
Sub DeleteRows()
   Do While DBGrid1.SelBookmarks.Count <> 0
      Data1.Recordset.Bookmark = DBGrid1.SelBookmarks(0)
      Data1.Recordset.Delete
      Data1.Refresh
   Loop
End Sub
```

VisibleCols Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproVisibleColsC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproVisibleColsX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproVisibleColsA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproVisibleColsS"}

Returns a value indicating the number of visible columns in the **DBGrid** control. Not available at design time and read-only at run time.

Syntax

object.**VisibleCols**

The **VisibleCols** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.

Remarks

The **VisibleCols** property is an integer ranging from 0 to the total number of columns available, as determined by the **Count** property of the **Columns** collection.

This property returns the number of visible columns in the current split. The value returned includes both fully and partially displayed columns.

Use the **Split** property to determine the index of the current split.

VisibleCols Property Example

This example defines buttons to move the grid a whole page left or right.

```
Private Sub PageRight_Click ()
   'Page grid to the right.
   If DBGrid1.LeftCol + DBGrid1.VisibleCols < _
       DBGrid1.Columns.Count Then
      DBGrid1.LeftCol = DBGrid1.LeftCol + _
       DBGrid1.VisibleCols
   End If
End Sub

Private Sub PageLeft_Click ()
   'Page grid to the left.
   If DBGrid1.LeftCol - DBGrid1.VisibleCols >= 0 Then
      DBGrid1.LeftCol = DBGrid1.LeftCol - _
       DBGrid1.VisibleCols
   End If
End Sub
```

VisibleRows Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"daproVisibleRowsC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"daproVisibleRowsX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"daproVisibleRowsA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"daproVisibleRowsS"}

Returns a value indicating the number of visible rows in the **DBGrid** control. This property is read-only at run time.

Syntax

object.**VisibleRows**

The *object* placeholder represents an object expression that evaluates to an object in the Applies To list.

Remarks

The **VisibleRows** property returns an integer ranging from 0 to the number of rows in the **DBGrid** control. The **VisibleRows** property includes either fully or partially visible **DBGrid** control rows.

VisibleRows Property Example

This example selects all the rows that are currently visible on the grid.

```
Private Sub SelectVisible_Click ()
   Dim I
   For I = 0 To DBGrid1.VisibleRows - 1
      DBGrid1.SelBookmarks.Add DBGrid1.RowBookmark(I)
   Next I
End Sub
```

DefaultValue Property (DBGrid Control)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproDefaultValuePropertyDBGridC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDefaultValuePropertyDBGridX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDefaultValuePropertyDBGridA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDefaultValuePropertyDBGridS"}

Sets the default value for a **Column** object in an unbound **DBGrid** control.

Syntax

object.**DefaultValue** [= *value*]

The **DefaultValue** property syntax has these parts:

Part	Description
object	An object expression that evaluates to an object in the Applies To list.
value	A Variant expression containing the default value for the specified column.

Remarks

The **DBGrid** control does not use this property itself, but provides it as a placeholder for you to associate default values with columns in an unbound grid. In the UnboundAddData event, you can use this property to retrieve default values for columns that were not supplied by the end-user. Such columns will contain a **Null** variant in the corresponding **RowBuffer.Value** property array. This property can also be used as a tag for a column (whether it is bound or unbound). Arbitrary values can be stored and retrieved later

Change Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtChangeEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtChangeEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtChangeEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtChangeEventControlsPlaceholderS"}

Click Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtClickEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtClickEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtClickEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtClickEventControlsPlaceholderS"}

DblClick Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtDblClickEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtDblClickEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtDblClickEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtDblClickEventControlsPlaceholderS"}

DragDrop Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtDragDropEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtDragDropEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtDragDropEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtDragDropEventControlsPlaceholderS"}

DragOver Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtDragOverEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtDragOverEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtDragOverEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtDragOverEventControlsPlaceholderS"}

KeyDown Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtKeyDownEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtKeyDownEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtKeyDownEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtKeyDownEventControlsPlaceholderS"}

KeyPress Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtKeyPressEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtKeyPressEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtKeyPressEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtKeyPressEventControlsPlaceholderS"}

KeyUp Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtKeyUpEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtKeyUpEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtKeyUpEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtKeyUpEventControlsPlaceholderS"}

MouseDown Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtMouseDownEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtMouseDownEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtMouseDownEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtMouseDownEventControlsPlaceholderS"}

MouseMove Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtMouseMoveEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtMouseMoveEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtMouseMoveEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtMouseMoveEventControlsPlaceholderS"}

MouseUp Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtMouseUpEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtMouseUpEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtMouseUpEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtMouseUpEventControlsPlaceholderS"}

OLECompleteDragEvent (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtOLECompleteDragEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtOLECompleteDragEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtOLECompleteDragEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtOLECompleteDragEventControlsPlaceholderS"}

OLEDragDrop Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtOLEDragDropEventPHolderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtOLEDragDropEventPHolderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtOLEDragDropEventPHolderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtOLEDragDropEventPHolderS"}

OLEDragOver Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtOLEDragOverEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtOLEDragOverEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtOLEDragOverEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtOLEDragOverEventControlsPlaceholderS"}

OLEGiveFeedback Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtOLEGiveFeedbackEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtOLEGiveFeedbackEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtOLEGiveFeedbackEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtOLEGiveFeedbackEventControlsPlaceholderS"}

OLESetData Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtOLESetDataEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtOLESetDataEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtOLESetDataEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtOLESetDataEventControlsPlaceholderS"}

OLEStartDrag Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtOLEStartDragEventControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtOLEStartDragEventControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtOLEStartDragEventControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtOLEStartDragEventControlsPlaceholderS"}

Refresh Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthRefreshMethodControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthRefreshMethodControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthRefreshMethodControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthRefreshMethodControlsPlaceholderS"}

OLEDrag Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthOLEDragMethodControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthOLEDragMethodControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthOLEDragMethodControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthOLEDragMethodControlsPlaceholderS"}

Appearance Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproAppearancePropertyControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproAppearancePropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproAppearancePropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproAppearancePropertyControlsPlaceholderS"}

BackColor Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBackColorPropertyControlsPlaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproBackColorPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproBackColorPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproBackColorPropertyControlsPlaceholderS"}

Caption Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproCaptionPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproCaptionPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproCaptionPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproCaptionPropertyControlsPlaceholderS"}

Enabled Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproEnabledPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproEnabledPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproEnabledPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproEnabledPropertyControlsPlaceholderS"}

Font Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproFontPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFontPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFontPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFontPropertyControlsPlaceholderS"}

ForeColor Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproForeColorPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproForeColorPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproForeColorPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproForeColorPropertyControlsPlaceholderS"}

IntegralHeight Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproIntegralHeightPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproIntegralHeightPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproIntegralHeightPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproIntegralHeightPropertyControlsPlaceholderS"}

Locked Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproLockedPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproLockedPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproLockedPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproLockedPropertyControlsPlaceholderS"}

MouseIcon Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproMouseIconPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMouseIconPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMouseIconPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMouseIconPropertyControlsPlaceholderS"}

MousePointer Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproMousePointerPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMousePointerPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMousePointerPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMousePointerPropertyControlsPlaceholderS"}

OLEDragMode Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproOLEDragModePropertyControlsPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLEDragModePropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLEDragModePropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLEDragModePropertyControlsPlaceholderS"}

OLEDropMode Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproOLEDropModePropertyControlsPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLEDropModePropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLEDropModePropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLEDropModePropertyControlsPlaceholderS"}

SelLength Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproSelLengthPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproSelLengthPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproSelLengthPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproSelLengthPropertyControlsPlaceholderS"}

SelStart Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproSelStartPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproSelStartPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproSelStartPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproSelStartPropertyControlsPlaceholderS"}

SelText Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproSelTextPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproSelTextPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproSelTextPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproSelTextPropertyControlsPlaceholderS"}

Style Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproStylePropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproStylePropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproStylePropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproStylePropertyControlsPlaceholderS"}

Text Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproTextPropertyControlsPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproTextPropertyControlsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproTextPropertyControlsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproTextPropertyControlsPlaceholderS"}

Keyword Not Found

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic"}

The keyword you've selected can't be found in Visual Basic Help. It's possible that:

- The keyword is misspelled
- You selected too much or too little text
- You asked for help on a word that is not a valid Visual Basic keyword.

The easiest way to get help on a specific keyword is to position the insertion point anywhere within the keyword you want help on and press F1. You do not need to select the keyword. In fact, if you select only a portion of the keyword, or more than a single word, Help will not find what you're looking for.

To use the built-in Help search dialog box, press the **Help Topics** button on the toolbar.

One option to possibly find what you're looking for is to view the ReadMe file that comes with Visual Basic. This document contains information regarding last-minute changes, additions, and deletions that did not make it into the final documentation.

Left, Top Properties (Placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproLeftTopPropertiesPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproLeftTopPropertiesPlaceholderS"}

Height, Width Properties (Placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproHeightWidthPropertiesPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHeightWidthPropertiesPlaceholderS"}

Tag Property (Placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproTagPropertyPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproTagPropertyPlaceholderS"}

ToolTipText Property (Placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproToolTipTextPropertyPlaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproToolTipTextPropertyPlaceholderS"}

Count Property (Placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproCountPropertyPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproCountPropertyPlaceholderS"}

Item Method (Placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproItemMethodPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthItemMethodPlaceholderS;vbproItemMethodPlaceholderS"}

Hwnd Property (Placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproHwndPropertyPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHwndPropertyPlaceholderS"}

TabStop Property (Placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproTabStopPropertyPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproTabStopPropertyPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproTabStopPropertyPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproTabStopPropertyPlaceholderS"}

Visible Property (Placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproVisiblePropertyPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproVisiblePropertyPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproVisiblePropertyPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproVisiblePropertyPlaceholderS"}

Picture Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproPicturePropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproPicturePropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproPicturePropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproPicturePropertyplaceholderS"}

Max, Min Properties (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproMaxMinPropertiesplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMaxMinPropertiesplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMaxMinPropertiesplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMaxMinPropertiesplaceholderS"}

Add Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthAddMethodplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthAddMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthAddMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthAddMethodplaceholderS"}

Clear Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthClearMethodplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthClearMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthClearMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthClearMethodplaceholderS"}

Container Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproContainerPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproContainerPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproContainerPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproContainerPropertyplaceholderS"}

Controls Collection (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcolControlsCollectionplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbcolControlsCollectionplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"vbcolControlsCollectionplaceholderP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"vbcolControlsCollectionplaceholderM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"vbcolControlsCollectionplaceholderE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbcolControlsCollectionplaceholderS"}

Copies Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproCopiesPropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproCopiesPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproCopiesPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproCopiesPropertyplaceholderS"}

DataBinding Object (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbobjDataBindingObjectplaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbobjDataBindingObjectplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"vbobjDataBindingObjectplaceholderP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"vbobjDataBindingObjectplaceholderM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"vbobjDataBindingObjectplaceholderE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbobjDataBindingObjectplaceholderS"}

DataBindings Collection (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcolDataBindingsCollectionplaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbcolDataBindingsCollectionplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"vbcolDataBindingsCollectionplaceholderP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"vbcolDataBindingsCollectionplaceholderM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"vbcolDataBindingsCollectionplaceholderE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbcolDataBindingsCollectionplaceholderS"}

Filename Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproFilenamePropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFilenamePropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFilenamePropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFilenamePropertyplaceholderS"}

FontBold, FontItalic, FontStrikeThru Properties (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproFontBoldFontItalicFontStrikeThruPropertiesplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFontBoldFontItalicFontStrikeThruPropertiesplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFontBoldFontItalicFontStrikeThruPropertiesplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFontBoldFontItalicFontStrikeThruPropertiesplaceholderS"}

FontName Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproFontNamePropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFontNamePropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFontNamePropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFontNamePropertyplaceholderS"}

FontSize Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproFontSizePropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFontSizePropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFontSizePropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFontSizePropertyplaceholderS"}

GetData Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthGetDataMethodplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthGetDataMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthGetDataMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthGetDataMethodplaceholderS"}

GetFormat Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthGetFormatMethodplaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthGetFormatMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthGetFormatMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthGetFormatMethodplaceholderS"}

HelpFile Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproHelpFilePropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHelpFilePropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHelpFilePropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHelpFilePropertyplaceholderS"}

Item Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproItemPropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproItemPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproItemPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproItemPropertyplaceholderS"}

Remove Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthRemoveMethodplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthRemoveMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthRemoveMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthRemoveMethodplaceholderS"}

SetData Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthSetDataMethodplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthSetDataMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthSetDataMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthSetDataMethodplaceholderS"}

Align Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproAlignPropertyplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproAlignPropertyplaceholderS"}

DragIcon Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproDragIconPropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDragIconPropertyplaceholderS"}

DragMode Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproDragModePropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDragModePropertyplaceholderS"}

HelpContextID Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproHelpContextIDPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHelpContextIDPropertyplaceholderS"}

TabIndex Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproTabIndexPropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproTabIndexPropertyplaceholderS"}

Alignment Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstAlignmentConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Border Property Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstBorderPropertyConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

BorderStyle Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstBorderStyleConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

BorderStyle Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproBorderStylePropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproBorderStylePropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproBorderStylePropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproBorderStylePropertyActiveXControlsplaceholderS"}

Clear Method (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthClearMethodActiveXControlsplaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthClearMethodActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthClearMethodActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthClearMethodActiveXControlsplaceholderS"}

Clipboard Object Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstClipboardObjectConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Color Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstColorConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

CommonDialog Control Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstCommonDialogControlConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

CommonDialog Error Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstCommonDialogErrorConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Control Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstControlConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

DataBindings Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproDataBindingsPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDataBindingsPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDataBindingsPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDataBindingsPropertyplaceholderS"}

DDE Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstDDEConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Drag-and-Drop Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstDragandDropConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Drawing Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstDrawingConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

FetchVerbs Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthFetchVerbsMethodC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthFetchVerbsMethodX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthFetchVerbsMethodA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthFetchVerbsMethodS"}

Form Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstFormConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Graphics Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstGraphicsConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Grid Control Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstGridControlConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Help Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstHelpConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

HideSelection Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproHideSelectionPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHideSelectionPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHideSelectionPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHideSelectionPropertyActiveXControlsplaceholderS"}

HideSelection Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproHideSelectionPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHideSelectionPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHideSelectionPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHideSelectionPropertyplaceholderS"}

Image Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproImagePropertyActiveXControlsplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproImagePropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproImagePropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproImagePropertyActiveXControlsplaceholderS"}

ImageList Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproImageListPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproImageListPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproImageListPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproImageListPropertyActiveXControlsplaceholderS"}

Index Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproIndexPropertyActiveXControlsplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproIndexPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproIndexPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproIndexPropertyActiveXControlsplaceholderS"}

Index Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproIndexPropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproIndexPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproIndexPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproIndexPropertyplaceholderS"}

Key Code Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstKeyCodeConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Key Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproKeyPropertyActiveXControlsplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproKeyPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproKeyPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproKeyPropertyActiveXControlsplaceholderS"}

Menu Accelerator Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstMenuAcceleratorConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Menu Control Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstMenuControlConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Miscellaneous Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstMiscellaneousConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

MousePointer Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstMousePointerConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

OLE Container Control Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstOLEContainerControlConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Picture Object Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstPictureObjectConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Printer Object Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstPrinterObjectConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

RasterOp Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstRasterOpConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Remove Method (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthRemoveMethodActiveXControlsplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthRemoveMethodActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthRemoveMethodActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthRemoveMethodActiveXControlsplaceholderS"}

ShowInTaskbar Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstShowInTaskbarPropertyplaceholderC;vbproBooksOnlineJumpTopic"}

ShowTips Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproShowTipsPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproShowTipsPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproShowTipsPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproShowTipsPropertyActiveXControlsplaceholderS"}

ShowWhatsThis Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthShowWhatsThisMethodplaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthShowWhatsThisMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthShowWhatsThisMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthShowWhatsThisMethodplaceholderS"}

Text Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproTextPropertyActiveXControlsplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproTextPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproTextPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproTextPropertyActiveXControlsplaceholderS"}

Value Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproValuePropertyActiveXControlsplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproValuePropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproValuePropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproValuePropertyActiveXControlsplaceholderS"}

Variant Type Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstVariantTypeConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

Visual Basic Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstVisualBasicConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

WhatsThisButton Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproWhatsThisButtonPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproWhatsThisButtonPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproWhatsThisButtonPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproWhatsThisButtonPropertyplaceholderS"}

WhatsThisHelp Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproWhatsThisHelpPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproWhatsThisHelpPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproWhatsThisHelpPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproWhatsThisHelpPropertyplaceholderS"}

WhatsThisHelpID Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproWhatsThisHelpIDPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproWhatsThisHelpIDPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproWhatsThisHelpIDPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproWhatsThisHelpIDPropertyplaceholderS"}

WhatsThisMode Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthWhatsThisModeMethodplaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthWhatsThisModeMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthWhatsThisModeMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthWhatsThisModeMethodplaceholderS"}

Windows 95 Control Constants (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcstWindows95ControlConstantsplaceholderC;vbproBooksOnlineJumpTopic"}

ColIsVisible Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproColIsVisiblePropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproColIsVisiblePropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproColIsVisiblePropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproColIsVisiblePropertyplaceholderS"}

ColPos Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproColPosPropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproColPosPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproColPosPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproColPosPropertyplaceholderS"}

DataObject Object (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproDataObjectObjectplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDataObjectObjectplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDataObjectObjectplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbobjDataObjectObjectplaceholderS;vbproDataObjectObjectplaceholderS"}

DataObjectFiles Collection (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcolDataObjectFilesCollectionplaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbcolDataObjectFilesCollectionplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Properties":"vbcolDataObjectFilesCollectionplaceholderP"} {ewc HLP95EN.DLL,DYNALINK,"Methods":"vbcolDataObjectFilesCollectionplaceholderM"} {ewc HLP95EN.DLL,DYNALINK,"Events":"vbcolDataObjectFilesCollectionplaceholderE"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbcolDataObjectFilesCollectionplaceholderS"}

DataSource Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproDataSourcePropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDataSourcePropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDataSourcePropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDataSourcePropertyplaceholderS"}

Drag Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthDragMethodplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthDragMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthDragMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthDragMethodplaceholderS"}

DrawMode Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproDrawModePropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDrawModePropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDrawModePropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDrawModePropertyplaceholderS"}

FixedAlignment Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproFixedAlignmentPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFixedAlignmentPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFixedAlignmentPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFixedAlignmentPropertyplaceholderS"}

GotFocus Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtGotFocusEventplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtGotFocusEventplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtGotFocusEventplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtGotFocusEventplaceholderS"}

GridLineWidth Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproGridLineWidthPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproGridLineWidthPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproGridLineWidthPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproGridLineWidthPropertyplaceholderS"}

Index Property (Control Array) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproIndexPropertyControlArrayplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproIndexPropertyControlArrayplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproIndexPropertyControlArrayplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproIndexPropertyControlArrayplaceholderS"}

LostFocus Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtLostFocusEventplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtLostFocusEventplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtLostFocusEventplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtLostFocusEventplaceholderS"}

Move Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthMoveMethodplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthMoveMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthMoveMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthMoveMethodplaceholderS"}

Name Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproNamePropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproNamePropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproNamePropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproNamePropertyplaceholderS"}

Object Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproObjectPropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproObjectPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproObjectPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproObjectPropertyplaceholderS"}

Parent Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproParentPropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproParentPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproParentPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproParentPropertyplaceholderS"}

RowColChange Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtRowColChangeEventplaceholderC;vbproBooksOnlineJumpTopic"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtRowColChangeEventplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtRowColChangeEventplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtRowColChangeEventplaceholderS"}

RowIsVisible Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproRowIsVisiblePropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproRowIsVisiblePropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproRowIsVisiblePropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproRowIsVisiblePropertyplaceholderS"}

RowPos Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproRowPosPropertyplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproRowPosPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproRowPosPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproRowPosPropertyplaceholderS"}

SetFocus Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthSetFocusMethodplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthSetFocusMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthSetFocusMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthSetFocusMethodplaceholderS"}

ZOrder Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthZOrderMethodplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthZOrderMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthZOrderMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthZOrderMethodplaceholderS"}

Files Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthFilesMethodplaceholderC;vbproBooksOnlineJumpTopic"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthFilesMethodplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthFilesMethodplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthFilesMethodplaceholderS"}

Connect Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproConnectEventplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproConnectEventplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproConnectEventplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproConnectEventplaceholderS"}

HScrollSmallChange, VScrollSmallChange Properties (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproHScrollSmallChangeVScrollSmallChangePropertiesplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHScrollSmallChangeVScrollSmallChangePropertiesplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHScrollSmallChangeVScrollSmallChangePropertiesplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHScrollSmallChangeVScrollSmallChangePropertiesplaceholderS"}

MinHeight, MinWidth Properties (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproMinHeightMinWidthPropertiesplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMinHeightMinWidthPropertiesplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMinHeightMinWidthPropertiesplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMinHeightMinWidthPropertiesplaceholderS"}

Property Pages Dialog Box (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproPropertryPagesDialogBoxplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproPropertryPagesDialogBoxplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproPropertryPagesDialogBoxplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproPropertryPagesDialogBoxplaceholderS"}

RemoteHost Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproRemoteHostPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproRemoteHostPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproRemoteHostPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproRemoteHostPropertyplaceholderS"}

RemotePort Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproRemotePortPropertyplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproRemotePortPropertyplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproRemotePortPropertyplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproRemotePortPropertyplaceholderS"}

ViewportHeight, ViewportLeft, ViewportTop, ViewportWidth Properties (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBooksOnlineJumpTopic;vbproViewportHeightViewportLeftViewportTopViewportWidthPropertiesplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproViewportHeightViewportLeftViewportTopViewportWidthPropertiesplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproViewportHeightViewportLeftViewportTopViewportWidthPropertiesplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproViewportHeightViewportLeftViewportTopViewportWidthPropertiesplaceholderS"}

Resync Method (Remote Data) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthResyncMethodRemoteDataplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthResyncMethodRemoteDataplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthResyncMethodRemoteDataplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthResyncMethodRemoteDataplaceholderS"}

ContainedVBControls Collection (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbcolContainedVBControlsCollectionplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbcolContainedVBControlsCollectionplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbcolContainedVBControlsCollectionplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbcolContainedVBControlsCollectionplaceholderS"}

Member Object (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproMemberObjectplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMemberObjectplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMemberObjectplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMemberObjectplaceholderS"}

Members Collection (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproMembersCollectionplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMembersCollectionplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMembersCollectionplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMembersCollectionplaceholderS"}

VBComponentsEvents Object (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproVBComponentsEventsObjectplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproVBComponentsEventsObjectplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproVBComponentsEventsObjectplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproVBComponentsEventsObjectplaceholderS"}

VBProjectsEvents Object (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproVBProjectsEventsObjectplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproVBProjectsEventsObjectplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproVBProjectsEventsObjectplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproVBProjectsEventsObjectplaceholderS"}

Files Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscFilesPropertyplaceholderC"}

Alignment Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscAlignmentPropertyplaceholderC"}

AboutBox Method (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscAboutBoxMethodplaceholderC"}

Col, Row Properties (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscColRowPropertiesplaceholderC"}

RowHeight Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscRowHeightPropertyplaceholderC"}

Scroll Event (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscScrollEventplaceholderC"}

ScrollBars Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscScrollBarsPropertyplaceholderC"}

SelEndCol, SelStartCol, SelEndRow, SelStartRow Properties (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscSelEndColPropertiesplaceholderC"}

ScrollBars Property (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscScrollBarsPropertyplaceholderC"}

Appearance Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproAppearancePropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproAppearancePropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproAppearancePropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproAppearancePropertyActiveXControlsplaceholderS"}

BackColor, ForeColor Properties (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBackColorForeColorPropertiesActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproBackColorForeColorPropertiesActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproBackColorForeColorPropertiesActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproBackColorForeColorPropertiesActiveXControlsplaceholderS"}

BorderStyle Constants (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBorderStyleConstantsActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproBorderStyleConstantsActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproBorderStyleConstantsActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproBorderStyleConstantsActiveXControlsplaceholderS"}

Caption Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproCaptionPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproCaptionPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproCaptionPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproCaptionPropertyActiveXControlsplaceholderS"}

Change Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproChangeEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproChangeEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproChangeEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproChangeEventActiveXControlsplaceholderS"}

Clear Method (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproClearMethodActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproClearMethodActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproClearMethodActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproClearMethodActiveXControlsplaceholderS"}

Click Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproClickEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproClickEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproClickEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproClickEventActiveXControlsplaceholderS"}

Clipboard Object Constants (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproClipboardObjectConstantsActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproClipboardObjectConstantsActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproClipboardObjectConstantsActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproClipboardObjectConstantsActiveXControlsplaceholderS"}

Count Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproCountPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproCountPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproCountPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproCountPropertyActiveXControlsplaceholderS"}

DataObject Object (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproDataObjectObjectActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDataObjectObjectActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDataObjectObjectActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDataObjectObjectActiveXControlsplaceholderS"}

DataObjectFiles Collection (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproDataObjectFilesCollectionActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDataObjectFilesCollectionActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDataObjectFilesCollectionActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDataObjectFilesCollectionActiveXControlsplaceholderS"}

DataSource Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproDataSourcePropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDataSourcePropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDataSourcePropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDataSourcePropertyActiveXControlsplaceholderS"}

DblClick Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproDblClickEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDblClickEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDblClickEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDblClickEventActiveXControlsplaceholderS"}

Drag-and-Drop Constants (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproDragandDropConstantsActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproDragandDropConstantsActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproDragandDropConstantsActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproDragandDropConstantsActiveXControlsplaceholderS"}

Enabled Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproEnabledPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproEnabledPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproEnabledPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproEnabledPropertyActiveXControlsplaceholderS"}

FetchVerbs Method (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproFetchVerbsMethodActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFetchVerbsMethodActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFetchVerbsMethodActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFetchVerbsMethodActiveXControlsplaceholderS"}

Files Method (ActiveX Controls) (placeholders)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproFilesMethodActiveXControlsplaceholdersC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFilesMethodActiveXControlsplaceholdersX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFilesMethodActiveXControlsplaceholdersA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFilesMethodActiveXControlsplaceholdersS"}

Font Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproFontPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFontPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFontPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFontPropertyActiveXControlsplaceholderS"}

FontName Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproFontNamePropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFontNamePropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFontNamePropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFontNamePropertyActiveXControlsplaceholderS"}

FontSize Property (ActiveX placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproFontSizePropertyActiveXPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproFontSizePropertyActiveXPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproFontSizePropertyActiveXPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproFontSizePropertyActiveXPlaceholderS"}

GetData Method (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproGetDataMethodActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproGetDataMethodActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproGetDataMethodActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproGetDataMethodActiveXControlsplaceholderS"}

GetFormat Method (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproGetFormatMethodActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproGetFormatMethodActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproGetFormatMethodActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproGetFormatMethodActiveXControlsplaceholderS"}

HideSelection Property (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproHideSelectionPropertyActiveXControlscomplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHideSelectionPropertyActiveXControlscomplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHideSelectionPropertyActiveXControlscomplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHideSelectionPropertyActiveXControlscomplaceholderS"}

hWnd Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproHWndPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHWndPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHWndPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHWndPropertyActiveXControlsplaceholderS"}

Item Method (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproItemMethodActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproItemMethodActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproItemMethodActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproItemMethodActiveXControlsplaceholderS"}

Key Code Constants (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproKeyCodeConstantsActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproKeyCodeConstantsActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproKeyCodeConstantsActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproKeyCodeConstantsActiveXControlsplaceholderS"}

Key Property (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproKeyPropertyActiveXControlscomplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproKeyPropertyActiveXControlscomplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproKeyPropertyActiveXControlscomplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproKeyPropertyActiveXControlscomplaceholderS"}

KeyDown, KeyUp Events (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproKeyDownKeyUpEventsActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproKeyDownKeyUpEventsActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproKeyDownKeyUpEventsActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproKeyDownKeyUpEventsActiveXControlsplaceholderS"}

KeyPress Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproKeyPressEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproKeyPressEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproKeyPressEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproKeyPressEventActiveXControlsplaceholderS"}

Max, Min Properties (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproMaxMinPropertiesActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMaxMinPropertiesActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMaxMinPropertiesActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMaxMinPropertiesActiveXControlsplaceholderS"}

MouseDown, MouseUp Events (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproMouseDownMouseUpEventsActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMouseDownMouseUpEventsActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMouseDownMouseUpEventsActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMouseDownMouseUpEventsActiveXControlsplaceholderS"}

MouseIcon Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproMouseIconPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMouseIconPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMouseIconPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMouseIconPropertyActiveXControlsplaceholderS"}

MouseMove Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproMouseMoveEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMouseMoveEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMouseMoveEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMouseMoveEventActiveXControlsplaceholderS"}

MousePointer Constants (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproMousePointerConstantsActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMousePointerConstantsActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMousePointerConstantsActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMousePointerConstantsActiveXControlsplaceholderS"}

MousePointer Property (ActiveX Controls) (placeholder

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproMousePointerPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproMousePointerPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproMousePointerPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproMousePointerPropertyActiveXControlsplaceholderS"}

OLECompleteDrag Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproOLECompleteDragEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLECompleteDragEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLECompleteDragEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLECompleteDragEventActiveXControlsplaceholderS"}

OLEDrag Method (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproOLEDragMethodActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLEDragMethodActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLEDragMethodActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLEDragMethodActiveXControlsplaceholderS"}

OLEDragDrop Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproOLEDragDropEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLEDragDropEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLEDragDropEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLEDragDropEventActiveXControlsplaceholderS"}

OLEDragMode Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproOLEDragModePropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLEDragModePropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLEDragModePropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLEDragModePropertyActiveXControlsplaceholderS"}

OLEDragOver Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproOLEDragOverEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLEDragOverEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLEDragOverEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLEDragOverEventActiveXControlsplaceholderS"}

OLEDropMode Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproOLEDropModePropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLEDropModePropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLEDropModePropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLEDropModePropertyActiveXControlsplaceholderS"}

OLEGiveFeedback Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproOLEGiveFeedbackEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLEGiveFeedbackEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLEGiveFeedbackEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLEGiveFeedbackEventActiveXControlsplaceholderS"}

OLESetData Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproOLESetDataEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLESetDataEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLESetDataEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLESetDataEventActiveXControlsplaceholderS"}

OLEStartDrag Event (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproOLEStartDragEventActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproOLEStartDragEventActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproOLEStartDragEventActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproOLEStartDragEventActiveXControlsplaceholderS"}

Picture Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproPicturePropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproPicturePropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproPicturePropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproPicturePropertyActiveXControlsplaceholderS"}

Property Pages Dialog Box (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproPropertyPagesDialogBoxActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproPropertyPagesDialogBoxActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproPropertyPagesDialogBoxActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproPropertyPagesDialogBoxActiveXControlsplaceholderS"}

RemoteHost Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproRemoteHostPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproRemoteHostPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproRemoteHostPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproRemoteHostPropertyActiveXControlsplaceholderS"}

RemotePort Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproRemotePortPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproRemotePortPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproRemotePortPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproRemotePortPropertyActiveXControlsplaceholderS"}

SelLength, SelStart, SelText Properties (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproSelLengthSelStartSelTextPropertiesActiveXControlsplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproSelLengthSelStartSelTextPropertiesActiveXControlsplaceholderX":1}
{ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproSelLengthSelStartSelTextPropertiesActiveXControlsplaceholderA"}
{ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproSelLengthSelStartSelTextPropertiesActiveXControlsplaceholderS"}

SetData Method (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproSetDataMethodActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproSetDataMethodActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproSetDataMethodActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproSetDataMethodActiveXControlsplaceholderS"}

ShowTips Property (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproShowTipsPropertyActiveXControlscomplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproShowTipsPropertyActiveXControlscomplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproShowTipsPropertyActiveXControlscomplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproShowTipsPropertyActiveXControlscomplaceholderS"}

Text Property (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproTextPropertyActiveXControlscomplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproTextPropertyActiveXControlscomplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproTextPropertyActiveXControlscomplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproTextPropertyActiveXControlscomplaceholderS"}

Value Property (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproValuePropertyActiveXControlscomplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproValuePropertyActiveXControlscomplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproValuePropertyActiveXControlscomplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproValuePropertyActiveXControlscomplaceholderS"}

hDC Property (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproHDCPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHDCPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHDCPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHDCPropertyActiveXControlsplaceholderS"}

Height, Width Properties (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproHeightWidthPropertiesActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHeightWidthPropertiesActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHeightWidthPropertiesActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHeightWidthPropertiesActiveXControlsplaceholderS"}

Index Property (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthIndexPropertyActiveXControlscomplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthIndexPropertyActiveXControlscomplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthIndexPropertyActiveXControlscomplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthIndexPropertyActiveXControlscomplaceholderS"}

Left, Top Properties (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproLeftTopPropertiesActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproLeftTopPropertiesActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproLeftTopPropertiesActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproLeftTopPropertiesActiveXControlsplaceholderS"}

Tag Property (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproTagPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproTagPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproTagPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproTagPropertyActiveXControlsplaceholderS"}

Visible Property (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproVisiblePropertyActiveXControlscomplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproVisiblePropertyActiveXControlscomplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproVisiblePropertyActiveXControlscomplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproVisiblePropertyActiveXControlscomplaceholderS"}

Remove Method (ActiveX Controls) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthRemoveMethodActiveXControlscomplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthRemoveMethodActiveXControlscomplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthRemoveMethodActiveXControlscomplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthRemoveMethodActiveXControlscomplaceholderS"}

Object Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthObjectPropertyActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthObjectPropertyActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthObjectPropertyActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthObjectPropertyActiveXControlsplaceholderS"}

Property Pages (ActiveX controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproPropertyPagesActiveXControlsplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproPropertyPagesActiveXControlsplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproPropertyPagesActiveXControlsplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproPropertyPagesActiveXControlsplaceholderS"}

No fonts exist (Error 24574) (Common Dialog Control) (complaceholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproNoFontsExistError24574CommonDialogControlcomplaceholderC"}
{ewc HLP95EN.DLL,DYNALINK,"Example":"vbproNoFontsExistError24574CommonDialogControlcomplaceholderX":1}
{ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproNoFontsExistError24574CommonDialogControlcomplaceholderA"}
{ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproNoFontsExistError24574CommonDialogControlcomplaceholderS"}

Help Contents placeholder

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproHelpContentsPlaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproHelpContentsPlaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproHelpContentsPlaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproHelpContentsPlaceholderS"}

BorderStyle Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproBorderStyleplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproBorderStyleplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproBorderStyleplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproBorderStyleplaceholderS"}

RightToLeft Property (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproRightToLeftplaceholderC"} {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproRightToLeftplaceholderX":1} {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproRightToLeftplaceholderA"} {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproRightToLeftplaceholderS"}

Refresh Method (ActiveX Controls) (placeholder)

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscRefreshMethodActiveXControlsplaceholderC"}