NNTP Client Control

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbobjNNTPClientControlC"}   {ewc
HLP95EN.DLL,DYNALINK,"Example":"vbobjNNTPClientControlX":1}   {ewc
HLP95EN.DLL,DYNALINK,"Properties":"vbobjNNTPClientControlP"}   {ewc
HLP95EN.DLL,DYNALINK,"Methods":"vbobjNNTPClientControlM"}   {ewc
HLP95EN.DLL,DYNALINK,"Events":"vbobjNNTPClientControlE"}   {ewc
HLP95EN.DLL,DYNALINK,"Specifics":"vbobjNNTPClientControlS"}

The **NNTP** control provides a reusable component that allows applications to access NNTP news servers (such as Newsreader). It furnishes news reading and posting capabilities.

Remarks

This control allows the Microsoft Access, Visual Basic, Visual C++, or Visual FoxPro developer to write applications that communicate with NNTP servers to view news groups, and retrieve and post news articles.

NNTP Commands

The following table summarizes the NNTP Client commands as specified in RFC977, and the NNTP extension commands as specified in Internet-Draft on Common NNTP Extensions. The extended commands make it possible to retrieve news groups with wildcard matching, access article threading information, and protocol commands that are intended specifically to support newsreader applications.

NNTP Client Commands	NNTP Extension Commands
ARTICLE	AUTHINFO
GROUP	LISTGROUP
LIST	LIST OVERVIEW.FMT
NEWSGROUP	XHDR
POST	XMOTD
QUIT	XOVER

ArticleNumbersSupported Property

Returns a value that determines if the **GetArticleNumbers** method can be used to retrieve a list of article numbers for a newsgroup. This property has no meaning before the connection to the server has been established. Read-only and unavailable at design time.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**ArticleNumbersSupported**
Visual FoxPro	*Object*.ArticleNumbersSupported
Visual C++	**BOOL GetArticleNumbersSupported**( );

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

Settings

The **ArticleNumbersSupported** property settings are:

Setting	Description
True	The **GetArticleNumbers** method can be used to retrieve a list of article numbers for a newsgroup.
False	Default. The **GetArticleNumbers** method can't be used to retrieve a list of article numbers for a newsgroup.

Data Type

Boolean

Remarks

To determine if the server supports the LISTGROUP, examine this property after a connection is established.

LastUpdate Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproLastUpdatePropertyC"}   {ewc HLP95EN.DLL,DYNALINK,"Example":"vbproLastUpdatePropertyX":1}   {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbproLastUpdatePropertyA"}   {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbproLastUpdatePropertyS"}

Returns or sets the default value used by the **GetAdministrationFile** and **ListNewGroups** methods.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**LastUpdate** [= *Date*]
Visual FoxPro	*Object*.LastUpdate[= *tCreationDate*]
Visual C++	**DATE GetLastUpdate**( ); **void SetLastUpdate**(**DATE** *NewValue*);

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

Data Type

Date

Default

Date of the control's creation

OverviewSupported Property

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbproOverviewSupportedPropertyC"}　　　{ewc
HLP95EN.DLL,DYNALINK,"Example":"vbproOverviewSupportedPropertyX":1}　　　{ewc
HLP95EN.DLL,DYNALINK,"Applies To":"vbproOverviewSupportedPropertyA"}　　　{ewc
HLP95EN.DLL,DYNALINK,"Specifics":"vbproOverviewSupportedPropertyS"}

Returns or sets a value that determines if the **GetOverviewFormat** and **GetOverview** methods can be used to retrieve header information stored in the server's overview database. This property has no meaning before the connection to the server has been established. Read-only and unavailable at design time.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**OverviewSupported**
Visual FoxPro	*Object*.OverviewSupported
Visual C++	**BOOL GetOverviewSupported**( );

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

Settings

The **OverviewSupported** property settings are:

Setting	Description
True	The **GetOverviewFormat** and **GetOverview** methods can be used to retrieve header information.
False	Default. The **GetOverviewFormat** and **GetOverview** methods can't be used to retrieve header information.

Data Type

Boolean

Remarks

To determine if the server supports the OVERVIEW.FMT command, examine this property after a connection is established.

AuthenticateRequest Event

Occurs when the connected NNTP server requests authentication.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*_**AuthenticateRequest** (*UserID* **As String,** *Password* **As String**)
Visual FoxPro	PROCEDURE *Object.*AuthenticateRequest LPARAMETERS c*UserID*, *cPassword*
Visual C++	**void** *dialogclass*::OnAuthenticateRequest*Control*(**BSTR FAR*** *UserId*, **BSTR FAR*** *Password*);

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

Arguments

Development Tool	Argument	Description
Microsoft Access and Visual Basic	*UserID*	Optional. User identification string to use for authentication.
Microsoft Access and Visual Basic	*Password*	Optional. Password to use for authentication.
Visual FoxPro	c*UserID*	Optional. User identification character expression to use for authentication.
Visual FoxPro	*cPassword*	Optional. Password character expression to use for authentication.
Visual C++	*UserID*	Optional. Pointer to a length-prefixed string containing the user identification string to use for authentication.
Visual C++	*Password*	Optional. Pointer to a length-prefixed string containing the password to use for authentication.

Remarks

If the *UserID* and *Password* arguments are specified, their values are used instead of the **UserID** and **Password** properties.

AuthenticateResponse Event

Occurs when an authentication response is received from the server.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*_**AuthenticateResponse** (**ByVal** *authenticated* **As Boolean**)
Visual FoxPro	PROCEDURE *Object.*AuthenticateResponse LPARAMETERS *lAuthenticated*
Visual C++	**void** *dialogclass*::OnAuthenticateResponse*Control*(**BOOL** *Authenticated*);

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

Arguments

Development Tool	Argument	Description
Microsoft Access, Visual Basic, and Visual C++	*authenticated*	Indicates if the authentication is successful. If this argument is **True**, the authentication has succeeded.
Visual FoxPro	*lAuthenticated*	Indicates if the authentication is successful. If this argument is true (.T.), the authentication has succeeded.

Data Type

Boolean

Banner Event

Occurs when the server responds with its sign-on banner after a connection is established.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*_**Banne**r (**ByVal** *Banner* **As String**)
Visual FoxPro	PROCEDURE *Object.*Banner LPARAMETERS *cBanner*
Visual C++	**void** *dialogclass*::OnBanner*Control*(**LPCTSTR** *Banner*);

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

Arguments

Development Tool	Argument	Description
Microsoft Access and Visual Basic	*Banner*	The sign-on message returned by the news server.
Visual FoxPro	*cBanner*	The sign-on message returned by the news server.
Visual C++	*Banner*	A pointer to the string containing the sign-on message returned by the news server.

GetAdministrationFile Method

Sends the NNTP XMOTD command to the server. This command retrieves the news server administrator's information if the information is newer than the value of *lastUpdate*.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetAdministrationFile** [*lastUpdate*]
Visual FoxPro	*Object*.GetAdministrationFile([*tLastUpdate*])
Visual C++	**void GetAdministrationFile**(**const VARIANT&** *LastUpdate*);

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

Arguments

Development Tool	Argument	Description
Microsoft Access and Visual Basic	*lastUpdate*	Optional. Indicates time of last update from the server. If the *lastUpdate* argument is not given, the Control uses the value of the **LastUpdate** property.
Visual FoxPro	*tLastUpdate*	Optional. Indicates time of last update from the server. If the *tLastUpdate* argument is not given, the Control uses the value of the **LastUpdate** property.
Visual C++	*LastUpdate*	Optional. Indicates time of last update from the server. If the *LastUpdate* argument is not given, the Control uses the value of the **GetLastUpdate** function.

Data Type

Date (**VARIANT** for Visual C++)

GetArticleByArticleNumber Method

Sends the NNTP ARTICLE command with *articleNumber* to the NNTP server. When this method reaches a successful completion, the DocOutput event occurs.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetArticleByArticleNumber** [*articleNumber*]
Visual FoxPro	*Object*.GetArticleByArticleNumber([*nArticleNumber*])
Visual C++	**void GetArticleByArticleNumber**(**const VARIANT&** *articleNumber*);

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

Arguments

Development Tool	Argument	Description
Microsoft Access, Visual Basic, and Visual C++	*articleNumber*	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
Visual FoxPro	*nArticleNumber*	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.

Data Type

Variant

Remarks

Valid Variant types for *articleNumber* are String and Integer.

GetArticleByMessageID Method

Sends the NNTP ARTICLE command with *messageID* to the server. When this method reaches a successful completion, the DocOutput event occurs.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetArticleByMessageID** *messageID*
Visual FoxPro	*Object*.GetArticleByMessageID(*cMessageID)*
Visual C++	**void GetArticleByMessageID**(**LPCTSTR** *messageID*);

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

Arguments

Development Tool	Argument	Description
Microsoft Access, Visual Basic, and Visual C++	*messageID*	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Visual FoxPro	*cMessageID*	Specifies the article's unique *cMessageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.

Data Type

String

GetArticleHeaders Method

Sends the NNTP XHDR command to the server. When this method reaches a successful completion, the DocOutput event occurs.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetArticleHeaders** *header*, [*firstArticle*,] [*lastArticle*]
Visual FoxPro	*Object*.GetArticleHeaders(*cHeader* [, *nFirstArticle*] [, *nLastArticle*])
Visual C++	**void GetArticleHeaders**(**LPCTSTR** *header*, **const VARIANT&** *firstArticle*, **const VARIANT&** *lastArticle*);

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

Arguments

Microsoft Access and Visual Basic

Argument	Data Type	Description
header	String	The name of a header line (e.g.,"subject") in a news group article. This parameter is required. See RFC-1036 for a list of valid header lines.
firstArticle	Long	Optional. If *firstArticle* and *lastArticle* are given, they indicate a range of article numbers.
lastArticle	Long	Optional. If *firstArticle* and *lastArticle* are given, they indicate a range of article numbers. If lastArticle is 0, the range is all headers following *firstArticle*.

Visual FoxPro

Argument	Data Type	Description
cHeader	Character expression	The name of a header line (e.g.,"subject") in a news group article. This parameter is required. See RFC-1036 for a list of valid header lines.
nFirstArticle	Numeric	Optional. If *nFirstArticle* and *nLastArticle* are given, they indicate a range of article numbers.
nLastArticle	Numeric	Optional. If *nFirstArticle* and *nLastArticle* are given, they indicate a range of article numbers. If *nLastArticle* is 0, the range is all headers following *nFirstArticle*.

Visual C++ Argument	Data Type	Description
header	**LPCTSTR**	Pointer to a string containing the name of a header line (e.g.,"subject") in a news group article. This parameter is required. See RFC-1036 for a list of valid header lines.The name of a header line (e.g.,"subject") in a news group article. This parameter is required. See RFC-1036 for a list of valid header lines.
firstArticle	**VARIANT**	Optional. If *firstArticle* and *lastArticle* are given, they indicate a range of article numbers.
lastArticle	**VARIANT**	Optional. If *firstArticle* and *lastArticle* are given, they indicate a range of article numbers. If *lastArticle* is 0, the range is all headers following *firstArticle*.

GetArticleNumbers Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthGetArticleNumbersMethodC"}   {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthGetArticleNumbersMethodX":1}   {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthGetArticleNumbersMethodA"}   {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthGetArticleNumbersMethodS"}

Sends the NNTP command LISTGROUP to the server. When this method reaches a successful completion, the DocOutput event occurs.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetArticleNumbers** [*groupName*]
Visual FoxPro	*Object*.GetArticleNumbers([*cGroupName*])
Visual C++	**void GetArticleNumbers(const VARIANT&** *groupName*);

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

Arguments

Development Tool	Argument	Description
Microsoft Access, Visual Basic, and Visual C++	*groupName*	Optional. If the *groupName* argument is given, a list of article numbers for that group is retrieved and the group becomes the selected group. If the *groupName* argument is not given, a list of article numbers for the selected news group is retrieved.

Visual FoxPro	*cGroupName*	Optional. If the *cGroupName* argument is included, a list of article numbers for that group is retrieved and the group becomes the selected group. If the *cGroupName* argument is omitted, a list of article numbers for the selected news group is retrieved.

Remarks

To determine if the current NNTP server supports the LISTGROUP command, use the **ArticleNumbersSupported** property after connection.

GetOverView Method

Sends the XOVER command to the server. Use the **OverviewSupported** property after connection to determine if the current NNTP server supports this method. When this method reaches a successful completion, the DocOutput event occurs.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetOverview** [*firstArticle*,] [*lastArticle*]
Visual FoxPro	*Object*.GetOverview([*nFirstArticle*] [, *nLastArticle*])
Visual C++	**void GetOverview(const VARIANT&** *firstArticle*, **const VARIANT&** *lastArticle*);

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

Arguments

Development Tool	Argument	Description
Microsoft Access, Visual Basic, and Visual C++	*firstArticle* *lastArticle*	Optional. If *firstArticle* and *lastArticle* are given, they indicate a range of article numbers. If *lastArticle* is 0, the range is all headers following *firstArticle*. If neither *firstArticle* nor *lastArticle* is given then information for the current article is retrieved.
Visual FoxPro	*nFirstArticle* *nLastArticle*	Optional. If *nFirstArticle* and *nLastArticle* are included, they indicate a range of article numbers. If *nLastArticle* is 0, the range is all headers following *nFirstArticle*. If *nFirstArticle* and *nLastArticle* are not included, information for the current article is retrieved.

Remarks

The XOVER command returns information from the overview database for the article(s) specified.

GetOverViewFormat Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthGetOverViewFormatMethodC"}   {ewc
HLP95EN.DLL,DYNALINK,"Example":"vbmthGetOverViewFormatMethodX":1}   {ewc
HLP95EN.DLL,DYNALINK,"Applies To":"vbmthGetOverViewFormatMethodA"}   {ewc
HLP95EN.DLL,DYNALINK,"Specifics":"vbmthGetOverViewFormatMethodS"}

Sends the LIST OVERVIEW.FMT command to the server. Use the **OverviewSupported** property
after connection to determine if the current NNTP sever supports this command. When this method
reaches a successful completion, the DocOutput event occurs.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetOverviewFormat**
Visual FoxPro	*Object*.GetOverviewFormat( )
Visual C++	**void GetOverviewFormat**( );

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

Arguments

None.

Remarks

The LIST OVERVIEW.FMT command is used to retrieve a list of headers in the order they appear in
the server's overview database.

ListGroupDescriptions Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthListGroupDescriptionsMethodC"}   {ewc
HLP95EN.DLL,DYNALINK,"Example":"vbmthListGroupDescriptionsMethodX":1}   {ewc
HLP95EN.DLL,DYNALINK,"Applies To":"vbmthListGroupDescriptionsMethodA"}   {ewc
HLP95EN.DLL,DYNALINK,"Specifics":"vbmthListGroupDescriptionsMethodS"}

Sends the NNTP LIST NEWSGROUPS command to the server. When this method reaches a successful completion, the DocOutput event occurs.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**ListGroupDescriptions**
Visual FoxPro	*Object*.ListGroupDescriptions( )
Visual C++	**void ListGroupDescriptions**( );

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

Arguments

None.

ListGroups Method

Sends NNTP LIST command to the server. The server responds with a list of all news groups. When this method reaches a successful completion, the DocOutput event occurs.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**ListGroups**
Visual FoxPro	*Object*.ListGroups( )
Visual C++	**void ListGroups**( );

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

Arguments

None.

ListNewGroups Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthListNewGroupsMethodC"}   {ewc HLP95EN.DLL,DYNALINK,"Example":"vbmthListNewGroupsMethodX":1}   {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbmthListNewGroupsMethodA"}   {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbmthListNewGroupsMethodS"}

Sends NNTP NEWGROUPS command to server. When this method reaches a successful completion, the DocOutput event occurs.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**ListNewGroups** [*LastUpdate* **As Variant**]
Visual FoxPro	*Object*.ListNewGroups([*tLastTime*])
Visual C++	**void ListNewGroups(const VARIANT&** *LastUpdate*);

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

Arguments

Development Tool	Argument	Description
Microsoft Access, Visual Basic, Visual C++	*LastUpdate*	Optional. Indicates the last time articles were retrieved by the client. If the *lastTime* parameter is not given, the Control uses the value of the **LastUpdate** property.
Visual FoxPro	*tLastTime*	Optional. Indicates the last time articles were retrieved by the client. If *tLastTime* is omitted, the Control uses the value of the **LastUpdate** property.

Data Type

Date (**VARIANT** for Visual C++)

Default Value

LastUpdate property

PostArticle Event

Occurs when an article is posted.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*_**PostArticle**()
Visual FoxPro	*Object*.**PostArticle**()

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

Arguments

None

PostingAllowed Property

Returns a value that indicates if the current NNTP server allows posting of news articles. This property has no meaning before the connection to the server has been established. Read-only and unavailable at design time.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**PostingAllowed**
Visual FoxPro	*Object*.PostingAllowed
Visual C++	**BOOL GetPostingAllowed**( );

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

Settings

The **PostingAllowed** property settings are:

Setting	Description
True	The current NNTP server allows posting of news articles.
False	The current NNTP server doesn't allow posting of news articles.

Remarks

To determine if the server supports posting, examine this property after a connection is established.

SelectGroup Event

Occurs after a successful completion of the **SelectGroup** method.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*_**SelectGroup**(**ByVal** *groupName* **As String, ByVal** *firstmessage* **As Long, ByVal** *lastMessage* **As Long, ByVal** *messageCount* **As Long**)
Visual FoxPro	PROCEDURE *Object.*SelectGroup LPARAMETERS *cGroupName*, *nFirstArticleNumber*, *nLastArticleNumber*, *nArticleCount*
Visual C++	**void** *dialogclass*::OnSelectGroup*Control*(**LPCTSTR** *groupName*, **long** *firstMessage*, **long** *lastMessage*, **long** *msgCount*);

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

Arguments

Microsoft Access and Visual Basic

Argument	Data Type	Description
groupName	String	The name of the group of articles to be selected.
firstMessage	Long	The number of the first article in the selected news group.
lastMessage	Long	The number of the last article in the selected news group.
messageCount	Long	An estimate by the server of the number of articles on file in the group.

Microsoft Access and Visual Basic

Argument	Data Type	Description
cGroupName	Character	The name of the group of articles to be selected.
nFirstArticleNumber	Numeric	The number of the first article in the selected news group.
nLastArticleNumber	Numeric	The number of the last article in the selected news group.
nArticleCount	Numeric	An estimate by the server of the number of articles on file in the group.

Microsoft Visual C++

Argument	Data Type	Description
groupName	**LPCTSTR**	Pointer to a string containing the name of the group of articles to be selected.
firstMessage	**long**	The number of the first article in the selected news group.
lastMessage	**long**	The number of the last article in the

selected news group.

msgCount	**long**	An estimate by the server of the number of articles on file in the group.

SelectGroup Method

Sends NNTP GROUP command to the server. On successful completion, the SelectGroup event occurs.

Return Value

Void

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**SelectGroup** *groupName*
Visual FoxPro	*Object*.SelectGroup(*cGroupName*)
Visual C++	**void SelectGroup**(**LPCTSTR** *groupName*);

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

Arguments

Development Tool	Argument	Description
Microsoft Access and Visual Basic	*groupName*	The name of the group of articles to be selected.
Visual FoxPro	*cGroupName*	The name of the group of articles to be selected.

Data Type

String

LastArticle Event

Occurs when the NNTP control is set to the last article in a message list.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*_**lastArticle**(**ByVa**l *articleNumber* **As Long**, **ByVal** *messageID* **As String**)
Visual FoxPro	PROCEDURE *Object.*LastArticle LPARAMETERS [*nArticleNumber,*] *cMessageID*
Visual C++	**void** *dialogclass*::OnLastArticle*Control*(**long** *articleNumber*, **LPCTSTR** *messageID*);

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

Arguments

Development Tool	Argument	Data Type	Description
Microsoft Access and Visual Basic	*articleNumber*	Long	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
	messageID	String	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Visual FoxPro	*nArticleNumber*		Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
	cMessageID		Specifies the article's unique *cMessageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from

			the message-id provided in the response to some other commands.
Visual C++	*articleNumber*	**long**	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
	messageID	**LPCTSTR**	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.

NextArticle Event

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbevtNextArticleEventC"}   {ewc HLP95EN.DLL,DYNALINK,"Example":"vbevtNextArticleEventX":1}   {ewc HLP95EN.DLL,DYNALINK,"Applies To":"vbevtNextArticleEventA"}   {ewc HLP95EN.DLL,DYNALINK,"Specifics":"vbevtNextArticleEventS"}

Occurs when the NNTP control is set to the next article in a message list.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*_**NextArticle**(**ByVal** *articleNumber* **As Long**, **ByVal** *messageID* **As String**)
Visual FoxPro	PROCEDURE *Object.*NextArticle LPARAMETERS [*nArticleNumber*,] *cMessageID*
Visual C++	**void** *dialogclass*::OnNextArticle*Control*(**long** *articleNumber*, **LPCTSTR** *messageID*);

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

Arguments

Development Tool	Argument	Data Type	Description
Microsoft Access and Visual Basic	*articleNumber*	Long	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
	messageID	String	Specifies the article's unique *messageID* for the current NNTP

			server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Visual FoxPro	*nArticleNumber*		Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
	cMessageID		Specifies the article's unique *cMessageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Visual C++	*articleNumber*	**long**	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
	messageID	**LPCTSTR**	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.

StatArticle Event

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*_**StatArticle**(**ByVal** *articleNumber* **As Long**, **ByVal** *messageID* **As String**)
Visual FoxPro	PROCEDURE *Object.*StatArticle LPARAMETERS [*nArticleNumber,*] *cMessageID*
Visual C++	**void** *dialogclass*::OnStatArticle*Control*(**long** *articleNumber*, **LPCTSTR** *messageID*);

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

Arguments

Development Tool	Argument	Data Type	Description
Microsoft Access and Visual Basic	*articleNumber*	Long	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
	messageID	String	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Visual FoxPro	*nArticleNumber*		Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
	cMessageID		Specifies the article's unique *cMessageID* for the current NNTP server.

			The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Visual C++	*articleNumber*	**long**	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
	messageID	**LPCTSTR**	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.

GetBodyByArticleNumber Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthGetBodyByArticleNumberMethodC"}   {ewc
HLP95EN.DLL,DYNALINK,"Example":"vbmthGetBodyByArticleNumberMethodX":1}   {ewc
HLP95EN.DLL,DYNALINK,"Applies To":"vbmthGetBodyByArticleNumberMethodA"}   {ewc
HLP95EN.DLL,DYNALINK,"Specifics":"vbmthGetBodyByArticleNumberMethodS"}

Requests the body of an article by its article number. A successful response from the server causes the DocOutput event to occur.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetBodyByArticleNumber** *articleNumber* **As Variant**
Visual FoxPro	*Object*.GetBodyByArticleNumber([*nArticleNumber*])
Visual C++	**void GetBodyByArticleNumber**(**const VARIANT&** *articleNumber*);

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

Arguments

Development Tool	Argument	Data Type	Description
Microsoft Access, Visual Basic, and Visual C++	*articleNumber*	Variant	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
Visual FoxPro	*nArticleNumber*	Numeric	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.

Return Value

Void

GetBodyByMessageID Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthGetBodyByMessageIDMethodC"}   {ewc
HLP95EN.DLL,DYNALINK,"Example":"vbmthGetBodyByMessageIDMethodX":1}   {ewc
HLP95EN.DLL,DYNALINK,"Applies To":"vbmthGetBodyByMessageIDMethodA"}   {ewc
HLP95EN.DLL,DYNALINK,"Specifics":"vbmthGetBodyByMessageIDMethodS"}

Requests the body of an article by its message ID. A successful response from the server causes the DocOutput event to occur.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object.***GetBodyByMessageID** *messageID* **As String**
Visual FoxPro	*Object.*GetBodyByMessageID(*cMessageID*)
Visual C++	**void GetBodyByMessageID**(**LPCTSTR** *messageID*);

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

Arguments

Development Tool	Argument	Data Type	Description
Microsoft Access and Visual Basic	*messageID*	String	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Visual FoxPro	*cMessageID*	Numeric	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Visual C++	*messageID*	**LPCTSTR**	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.

Return Value

Void

GetHeaderByArticleNumber Method

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmthGetHeaderByArticleNumberMethodC"}   {ewc
HLP95EN.DLL,DYNALINK,"Example":"vbmthGetHeaderByArticleNumberMethodX":1}   {ewc
HLP95EN.DLL,DYNALINK,"Applies To":"vbmthGetHeaderByArticleNumberMethodA"}   {ewc
HLP95EN.DLL,DYNALINK,"Specifics":"vbmthGetHeaderByArticleNumberMethodS"}

Requests the header of an article by its article number. A successful response from the server causes the DocOutput event to occur.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetHeaderByArticleNumber** *articleNumber* **As Variant**
Visual FoxPro	*Object*.GetHeaderByArticleNumber([*nArticleNumber*])
Visual C++	**void GetHeaderByArticleNumber**(**const VARIANT&** *articleNumber*);

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

Arguments

Development Tool	Argument	Data Type	Description
Microsoft Access, Visual Basic, and Visual C++	*articleNumber*	Variant	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
Visual FoxPro	*nArticleNumber*	Numeric	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.

Return Value

Void

GetHeaderByMessageID Method

Requests the header of an article by the article's message ID. A successful response from the server causes the DocOutput event to occur.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetHeaderByMessageID** *messageID* **As String**
Visual FoxPro	*Object*.GetHeaderByMessageID(*nMessageID*)
Visual C++	**void GetHeaderByMessageID**(**LPCTSTR** *messageID*);

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

Arguments

Development Tool	Argument	Data Type	Description
Microsoft Access and Visual Basic	*messageID*	String	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Visual FoxPro	*nMessageID*	Numeric	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Visual C++	*messageID*	**LPCTSTR**	Specifies the article's unique *messageID* for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.

Return Value

Void

GetStatByArticleNumber Method

Requests the stat of an article by its article number. A successful response from the server causes the
DocOutput event to occur.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**GetStatByArticleNumber** *articleNumber* **As Variant**
Visual FoxPro	*Object*.GetStatByArticleNumber([*nArticleNumber*])
Visual C++	**void GetStatByArticleNumber**(**const VARIANT&** *articleNumber*);

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

Arguments

Development Tool	Argument	Data Type	Description
Microsoft Access, Visual Basic, and Visual C++	*articleNumber*	Variant	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
Visual FoxPro	*nArticleNumber*	Numeric	Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.

Return Value

Void

SetLastArticle Method

Selects the last article of a news group.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**SetLastArticle**
Visual FoxPro	*Object*.SetLastArticle( )
Visual C++	**void SetLastArticle**( );

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

Arguments

None

Return Value

Void

SetNextArticle Method

Selects the next article of a news group.

Syntax

Development Tool	Syntax
Microsoft Access and Visual Basic	*object*.**SetNextArticle**
Visual FoxPro	*Object*.SetNextArticle( )
Visual C++	**void SetNextArticle**( );

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

Arguments

None

Return Value

Void

Using the NNTP Client Control

{ewc HLP95EN.DLL,DYNALINK,"See Also":"vbmscNNTPOverviewC"}

The **NNTP** Client control allows you to interact with Internet newsgroups. With this control, you can connect to a news server, retrieve a list of available newsgroups and their descriptions, enter a newsgroup, get lists of articles, and get any article.

The Networking News Transfer Protocol (NNTP) Client control implements the basic client NNTP Protocol as specified by RFC977, *Network News Transfer Protocol*. The NNTP control also implements NNTP extension commands as documented in the Internet-Draft on Common NNTP Extensions. For questions on Internet-Drafts, contact Internet-Drafts@CNRI.Reston.VA.US.

Possible Uses

The **NNTP** Client control can be used in the following scenarios:

- To create an application that automatically browses selected newsgroups.
- To create an application that searches messages for selected keywords allowing the user to retrieve only those messages that interest him or her.

Scenario: a Custom News Reader

A typical use of the **NNTP** Client control might be to develop a custom news reader. The basic steps necessary to implement a news reader include the following:

1. Use a **RichTextBox** control and a global event flag.
2. Connect to a remote server using the **Connect** method.
3. Authenticate the user with the AuthenticateRequest event.
4. Get a list of newsgroups using the **ListGroups** method and the **GetData** method in the DocOutput event.
5. Parse the returned ListGroups data.
6. Enter a Newsgroup using the **SelectGroup** method.
7. Get a list of articles using the **GetArticleHeaders** method in the SelectGroup event.
8. Retrieve a message's text using the **GetArticleByArticleNumber** method.
9. Disconnect with the **Quit** method.

Setup

In the Visual Basic example below, the following objects are used:

- **NNTP** control named "NNTP1"
- **RichTextBox** control named "rtfList"
- **TextBox** control named "txtUserID"
- **TextBox** control named "txtPassword"
- **TextBox** control named "txtRemoteHost"
- **TextBox** control named "txtRemotePort"
- **TextBox** control named "txtSelectGroup"
- **TextBox** control named "txtArticleHeaders"
- **TextBox** control named "txtNumber"
- **TextBox** control named "txtMessage"
- **CommandButton** named "cmdListGroups"
- **CommandButton** named "cmdSelectGroup"
- **CommandButton** named "cmdGetArticleHeaders"
- **CommandButton** named "cmdGetArticleByArticleNumber"

- **CommandButton** named "cmdQuit"

Use a RichTextBox Control and a Global Event Flag

The list of available newsgroups on any given server is usually extremely large. When using the **NNTP** control to retrieve such a list, it's possible to quickly reach the limits of a standard **TextBox** control. To avoid this, you should use a "bottomless" **TextBox** control such as the **RichTextBox** control to receive the results of a **ListGroups** method.

The bulk of the methods of the **NNTP** control cause the DocOutput event to occur. The DocOutput event passes a reference to the **DocOutput** object, which automatically parses the data stream and notifies you of the data type or the state of the transaction. For more information, see "Using **DocInput** and **DocOutput** Objects".

However, the DocOutput event does not tell you what method has caused the event to occur. This information is critical, since it determines where the results are to be stored. For example, using the **ListGroups** method will return a long list of all available newsgroups. You will want to store this data in either a database or a **RichTextBox** control. On the other hand, a **GetArticlesByArticleNumber** method will return a single message, which will go into a different text box.

To stream the data into the appropriate text box, set up a global event flag in your program. This flag notifies the DocOutput event of the type of method invoked, and consequently which text box the returned data should go into. In Visual Basic, you simply declare a global variable in the Declarations section of the code window:

```
Public EventFlag as Integer
```

In the Declarations section, you should also create a list of constants in your program that correspond to the methods which your program will invoke. A short example is shown below:

```
Public Const LISTGROUPS = 1
Public Const GETARTICLEHEADERS = 2
Public Const GETARTICLENUMBERS = 3
Public Const GETARTICLESUBJECT = 4
Public Const SELECTGROUP = 5
```

As each of the methods you implement is invoked, your program will set the event flag accordingly. Thereafter, in the DocOutput event code, examine the event flag to determine where to route the data.

Connect to a News Server Using the Connect Method

The first step is to use the **Connect** method to connect to a remote machine which hosts the news server. A **TextBox** control contains the name of the remote server, which is supplied as an argument to the **Connect** method:

```
Private Sub cmdConnect()
    ' txtRemoteHost contains news server name.
    NNTP1.Connect txtRemoteHost
End Sub
```

Authenticate the User with the AuthenticateRequest Event

Some news servers require user authentication in the form of a user ID and a password. For such servers, you can implement the AuthenticateRequest event into your application. This event passes the parameters *UserID* and *Password* to the news server. The code below passes the control's **UserID** and **Password** properties, assuming the end user has supplied this information in the **TextBox** controls:

```
Sub NNTP1_AuthenticateRequest(UserID As String, Password As String)
    UserID = txtUserID.Text
```

```
      Password = txtPassword.Text
End Sub
```

Get a List of Newsgroups Using the ListGroups Method and the GetData Method in the DocOutput Event

Once you have connected to a news server, you must get a list of the available newsgroups. The **ListGroups** method requires no arguments. However, invoking the **ListGroups** method causes the DocOutput event to occur. In the code below, the variable EventFlag is first set to the predefined constant (LISTGROUPS) that signifies the **ListGroups** method. Consequently, the DocOutput event code retrieves the data using the **GetData** method. The code then checks the variable EventFlag to route the data to a **RichTextBox** control called "rtfList".

```
Private Sub cmdListGroups_Click()
    EventFlag = LISTGROUPS  ' Set the event flag.
    NNTP1.ListGroups        ' Invoke the method.
End Sub

Sub NNTP1_DocOutput(ByVal DocOutput As DocOutput)
    Dim vtData As Variant ' Variable for data.

    ' Use GetData if the State = icDocData.
    Select Case DocOutput.State
    Case icDocData
       DocOutput.GetData vtData
    Case icDocError
       ' Handle errors here.
    End Select

    ' Use Select Case with the event flag.
    Select Case EventFlag
    Case LISTGROUPS
       ' Route the data into a RichTextBox control.
       rtfList.Text = rtfList.Text & vtData & _
       vbCrLF
    Case GETARTICLEHEADERS
       ' Handle GetArticleHeaders output here.
    .
    .
    End Select
End Sub
```

Parse the Returned ListGroups Data

There is one addition that may be made to the code above. When the **ListGroups** method is invoked, the DocOutput event returns a block of data (vtData) that includes the name, description, and other information about the news group. You may wish to parse out the unneeded data using a user-defined function. For an example, see "User-Defined ParseData Function."

```
rtfList.Text = rtfList.Text & ParseData(vtData)
```

Enter a Newsgroup Using the SelectGroup Method

Once you have retrieved a list of all available newsgroups, you can select one to enter. Use the **SelectGroup** method with the name of the group you wish to enter as argument. The code below invokes the **SelectGroup** method using a **TextBox** control to supply the argument.

```
Sub cmdSelectGroup_Click()
    ' The SelectGroup method causes the DocOutput
```

```
   ' event to occur.
   NNTP1.SelectGroup txtSelectGroup
End Sub
```

Get a List of Articles Using the GetArticleHeaders Method in the SelectGroup Event

The **SelectGroup** method causes the SelectGroup event to occur. This event passes three arguments, the GroupName, first message, and last message. It's most convenient to pass the values of these arguments to the **GetArticleHeaders** method. This method retrieves the available article headers from the server for the group that the user has selected.

The **GetArticleHeaders** method requires at least one argument which specifies which header to return. There are only two, well-defined headers. These are "Subject" and "Author."

The example below invokes the **SelectGroup** method and specifies a group to enter. The SelectGroup event occurs, which then invokes the **GetArticleHeaders** method. In turn the DocOutput event occurs and the data is sent to a **TextBox** control.

```
Sub NNTP1_SelectGroup(ByVal groupname As String, _
ByVal firstMessage As Long, ByVal msgCount As Long)
   EventFlag = SELECTGROUP ' Set the event flag.
   ' Invoke the GetArticleHeaders method,
   ' specifying the "Subject" line to retrieve,
   ' as well as the first and last message
   ' numbers as arguments.
   NNTP1.GetArticleHeaders "subject", _
   Trim(Str(firstMessage)), Trim(Str(lastMessage))
End Sub

Sub NNTP1_DocOutput(DocOutput As DocOutput)
   ' Use the Select Case statement to
   ' specify how to process the data.
   Dim vData As Variant ' Variant for data.

   Select Case Document.State
   Case icDocData
      ' Retrieve the data from the DocOutput object.
      DocOutput.GetData vtData
      '
      ' Handle other cases.
   End Select

   ' Use a Select Case with the event flag.
   Select Case EventFlag
      Case GETARTICLEHEADERS
         ' Route the data into a RichTextBox control
         ' named "rtfHeaders."
         txtArticleHeaders.Text = _
         txtArticleHeaders.Text & vtData & vbCrLF
      '
      ' Handle other cases
   End Select
End Sub
```

Retrieve a Message's text Using the GetArticleByArticleNumber Method

Finally, to retrieve the body of a message, you can use the article number which is included in the data returned by the **GetArticleHeader** method. The **GetArticleByArticleNumber** method requires only one argument, the number of the article to be retrieved.

The code below assumes the article number has been placed into a **TextBox** control named "txtNumber." As with the other methods, the event flag notifies the DocOutput event how to process the returned data.

```
Private Sub cmdGetByNumber_Click()
    EventFlag = GETARTICLEBYNUMBER
    NNTP1.GetArticleByArticleNumber CStr(txtNumber)
End Sub

Sub NNTP1_DocOutput(DocOutput As DocOutput)
    Dim vData As Variant ' Variant for data.

    ' Use the Select Case statement to
    ' specify how to process the data.
    Select Case Document.State
    Case icDocData
       ' Retrieve the data from the DocOutput object.
       DocOutput.GetData vtData, vbString
       '
       ' Handle other cases.
    End Select

    Select Case EventFlag   ' Use a Select Case with
                            ' the event flag.
       Case GETARTICLEBYNUMBER
          ' Route the data into a TextBox control
          ' named "txtMessage."
          txtMessage.Text = vtData
          '
          ' Handle other cases.
    End Select
End Sub
```

Disconnect with the Quit Method

When your user has finished browsing newsgroups, the program should exit gracefully from the news server. The **Quit** method disconnects your application, as shown in the following code. This method takes no arguments:

```
Private Sub cmdQuit_Click()
    NNTP1.Quit
End Sub
```

User-Defined ParseData Function Example

This function is designed to be used with the NNTP Client control. After invoking the ListGroups method, this function parses the returned data which contains several group listings as well as the start and end message numbers. The function returns only the names of news groups after parsing out the start and end message numbers.

```
Public Function ParseData(NewsGroupInfo As String)
    Dim Group As String
    Dim FullGroup As String
    Dim GroupDesc As String
    Dim grpPos As Long
    Dim LenSTR As Long
    Dim LenGRP As Long
    Dim ExtraData As String
    '-------------------------------------------------
    Screen.MousePointer = vbHourglass
    grpPos = 1 ' Position of first news group name.
    On Error Resume Next
    Do
        If (ExtraData <> "") Then
        NewsGroupInfo = ExtraData & NewsGroupInfo
        ExtraData = ""
        End If

        ' Get length of group info.
        LenGRP = InStr(grpPos, NewsGroupInfo, _
        vbCr) - grpPos
        If LenGRP < 1 Then
        ExtraData = Mid(NewsGroupInfo, grpPos)
        Exit Do
        End If

        ' Get group info.
        FullGroup = Mid(NewsGroupInfo, grpPos, LenGRP)
        If ((FullGroup = vbNullString) Or _
        (FullGroup = ".")) Then
        Exit Do    ' Validate group data.
        End If

        ' Search for end of group name
        LenSTR = InStr(1, FullGroup, vbTab) -
        If LenSTR < 1 Then LenSTR = _
        InStr(1, FullGroup, " ") - 1
        If LenSTR < 1 Then LenSTR = Len(FullGroup)

        ' Extract group name only.
        Group = Mid(FullGroup, 1, LenSTR)
        GroupDesc = " [" & _
        Trim(Mid(FullGroup, LenSTR + 1)) & "]"

        ' Return the parsed string.
        ParseData = ParseData & Group & vbCrLf

        ' Reset group position.
```

```
            grpPos = grpPos + LenGRP + 2
        Loop Until (FullGroup = vbNullString)
        Screen.MousePointer = vbDefault
End Function
```