



## Object reference

### **jmail.smtpmail**

- ◆ **Body** : *String*
- ◆ **Charset** : *String*
- ◆ **ContentTransferEncoding** : *String*
- ◆ **ContentType** : *String*
- ◆ **DeferredDelivery** : *Date*
- ◆ **Encoding** : *String*
- ◆ **ErrorCode** : *Integer*
- ◆ **ErrorMessage** : *String*
- ◆ **ErrorSource** : *String*
- ◆ **ISOEncodeHeaders** : *Boolean*
- ◆ **Lazysend** : *Boolean*
- ◆ **Log** : *String*
- ◆ **Logging** : *Boolean*
- ◆ **MailDomain** : *String*
- ◆ **MimeVersion** : *String*
- ◆ **Priority** : *Integer*
- ◆ **Recipients** : *String*
- ◆ **ReplyTo** : *String*
- ◆ **ReturnReceipt** : *Boolean*
- ◆ **Sender** : *String*
- ◆ **SenderName** : *String*
- ◆ **ServerAddress** : *String*
- ◆ **Silent** : *Boolean*
- ◆ **SimpleLayout** : *Boolean*
- ◆ **Subject** : *String*
- ◆ **UsePipelining** : *Boolean*
- ◆ **AddAttachment**( *FileName*, [*ContentType*])
- ◆ **AddCustomAttachment**( *FileName*, *Data* )
- ◆ **AddHeader**( *XHeader*, *Value* )
- ◆ **AddNativeHeader**( *Header*, *Value* )
- ◆ **AddRecipient**( *Email* )
- ◆ **AddRecipientBCC**( *Email* )
- ◆ **AddRecipientCC**( *Email* )
- ◆ **AddRecipientEx**( *Email*, *Name* )
- ◆ **AddURLAttachment**( *bstrURL*, *bstrAttachAs*, [*bstrAuth*])
- ◆ **AppendBodyFromFile**( *FileName* )
- ◆ **AppendText**( *Text* )
- ◆ **ClearAttachments**()
- ◆ **ClearCustomHeaders**()
- ◆ **ClearRecipients**()
- ◆ **Close**()
- ◆ **Execute**()
- ◆ **ExtractEmailAddressesFromURL**( *bstrURL*, [*bstrAuth*])

◆ **GetMessageBodyFromURL( *bstrURL*, [*bstrAuth*] )**

◆ **LogCustomMessage( *Message* )**

## Properties

◆ **Body** : *String*

(v3.0)

The body of the message. To append text to the body use AppendText

◆ **Charset** : *String*

(v3.0)

This is the charset of the message. The default is "US-ASCII"

◆ **ContentTransferEncoding** : *String*

(v3.0)

Specifies the content transfer encoding. The default is "Quoted-Printable"

◆ **ContentType** : *String*

(v3.0)

This is the contenttype of the message. It defaults to "text/plain" but can be set to whatever you need. If you want to send HTML in your messages, change this to "text/html"

◆ **DeferredDelivery** : *Date*

(v3.0)

Sets defered delivery of messages. If the mailservier supports it the message wont be delivered before this date and time.

◆ **Encoding** : *String*

(v3.0)

This can be used to change the default Attachment encoding from base64. Valid options are "base64", "uuencode" or "quoted-printable"

◆ **ErrorCode** : *Integer*

(v3.0)

Contains the error code if JMail.silent is set to true

◆ **ErrorMessage** : *String*

(v3.0)

Contains the error message if JMail.silent is set to true

◆ **ErrorSource** : *String*

(v3.0)

Contains the error source if JMail.silent is set to true

◆ **ISOEncodeHeaders** : *Boolean*

(v3.0)

Encodes header strings according to iso-8859-1 character sets. The default is true.

◆ **Lazysend** : *Boolean*

(v3.0)

This property specifies if JMail is to wait until the mail is sent and then return or if it is to buffer the message and send it in the background. By setting this you do however don get control of error messages etc.<BR><BR><B> NOTE:</B> If you use this option, ServerAddress have NO function. The lazysend function will resolve the Mailserver with DNS queries. This can be a problem in some configurations.

◆ **Log** : *String*

(v3.0)

This is the log created by JMail when logging is set to true

◆ **Logging** : *Boolean*

(v3.0)

Enables/Disables logging in JMail

◆ **MailDomain** : *String*

(v3.0)

This can be used to override the EHLO/HELO statement to your mailserver

◆ **MimeVersion** : *String*

(v3.0)

Specifies the mime version. The default is "1.0"

◆ **Priority** : *Integer*

(v3.0)

This is the priority of the message. The range of priorities can be from 1 to 5.<ol><li>This means that the message is High Priority. Some mailers prefer to call this level "Urgent".</li><li>This is also high priority.</li><li>This is normal priority.</li><li>This is low priority.</li><li>This is the lowest priority.</li></ol>

◆ **Recipients** : *String*

(v3.0)

Readonly property of all recipients of this message

◆ **ReplyTo** : *String*

(v3.0)

Specifies a optional reply address

◆ **ReturnReceipt** : *Boolean*  
(v3.0)

Specifies wether or not the sender requires a return receipt. The default value of the property is "false"

◆ **Sender** : *String*  
(v3.0)

Specifies the sender address of the message

◆ **SenderName** : *String*  
(v3.0)

Specifies the sender name of the message

◆ **ServerAddress** : *String*  
(v3.0)

Specifies the address of the server. There can be more than one server specified by separating the list with a semicolon. If a port other than 25 is used then specify this by adding a colon after the servername. If the serverAddress is left blank JMail will try to resolv the remote Mail server and sen the message directly to that server.

◆ **Silent** : *Boolean*  
(v3.0)

Set to true, JMail will not trow exceptions. Instead JMail.execute() will return true or false depending on the success of the operation

◆ **SimpleLayout** : *Boolean*  
(v3.0)

Set to true to reduce the number of headers JMail produces.

◆ **Subject** : *String*  
(v3.0)

Specifies the subject of the message

◆ **UsePipelining** : *Boolean*  
(v3.0)

Overrides if JMail should use pipelining on a server that supports it.

## Methods

### ◆ **AddAttachment**(*FileName*, [*ContentType*])

(v3.0)

Adds a file attachment to the message

### ◆ **AddCustomAttachment**(*FileName*, *Data*)

(v3.0)

Adds a custom attachment. This can be used to attach "virtual files" like a generated text string or certificate etc.

### ◆ **AddHeader**(*XHeader*, *Value*)

(v3.0)

Adds a user defined X-header to the message

### ◆ **AddNativeHeader**(*Header*, *Value*)

(v3.0)

Adds a header to the message

### ◆ **AddRecipient**(*Email*)

(v3.0)

Adds a recipient to the message

### ◆ **AddRecipientBCC**(*Email*)

(v3.0)

Adds a Blind Carbon Copy recipient to the message

### ◆ **AddRecipientCC**(*Email*)

(v3.0)

Adds a Carbon Copy recipient to the message

### ◆ **AddRecipientEx**(*Email*, *Name*)

(v3.0)

Adds a recipient with a name to the message

### ◆ **AddURLAttachment**(*bstrURL*, *bstrAttachAs*, [*bstrAuth*])

(v3.0)

Downloads and adds an attachment based on an URL. A seconds argument, "AttachAs", is used for specifying the filename that the attachment will receive in the message. A third and optional argument is used for optional WWW-Authentication.

### ◆ **AppendBodyFromFile(*FileName*)**

(v3.0)

Appends body text from a file

### ◆ **AppendText(*Text*)**

(v3.0)

Append "text" to body

### ◆ **ClearAttachments()**

(v3.0)

Clears the list of attachments

### ◆ **ClearCustomHeaders()**

(v3.0)

Clears all custom headers

### ◆ **ClearRecipients()**

(v3.0)

Clear the recipient list

### ◆ **Close()**

(v3.0)

Force JMail to close an cached connection to a mailserver.

### ◆ **Execute()**

(v3.0)

Executes and sends the message to the server

### ◆ **ExtractEmailAddressesFromURL(*bstrURL*, [*bstrAuth*])**

(v3.0)

Downloads and adds email addresses from an URL.

### ◆ **GetMessageBodyFromURL(*bstrURL*, [*bstrAuth*])**

(v3.0)

Clears the body of the message and replaces it with the contents of the URL. The contenttype is automaticly set to match the contenttype of the URL. The second argument (login and password) is optional

### ◆ **LogCustomMessage(*Message*)**

(v3.0)

Logs a custom user message to the JMail log. This function works ONLY if logging is set to true