

mail

Using GMail

To connect to GMail the following settings supplied in the properties Array:

```
var userName = 'xxx';
var passWord = 'yyy';
var properties = new Array();
properties[0] = 'mail.smtp.host=smtp.gmail.com';
properties[1] = 'mail.smtp.auth=true';
properties[2] = 'mail.smtp.username=' + userName;
properties[3] = 'mail.smtp.password=' + passWord;
properties[4] = 'mail.smtp.port=587';
properties[5] = 'mail.smtp.starttls.enable=true';
```

Return Types

[Attachment MailMessage](#)

Method Summary

[Attachment](#) [#createBinaryAttachment](#)(filename, binarydata, [mimeType])
Creates a binary attachment object.

[Attachment](#) [#createTextAttachment](#)(filename, textdata, [mimeType])
Creates a text based attachment object.

[String](#) [#getLastSendMailExceptionMsg](#)()
Get the exception that occurred in the last sendMail attempt (null if no exception occurred).

[MailMessage](#) [#getMailMessage](#)(binaryblob/string)
Helper method, returns MailMessage object from binary or 7bits string.

[String\[\]](#) [#getPlainMailAddresses](#)(addressesString)
Helper method to only get the plain addresses.

[Boolean](#) [#isValidEmailAddress](#)(email)
Checks whether the given e-mail address is valid or not.

[MailMessage](#) [#receiveMail](#)(userName, password, leaveMsgsOnServer, [receiveMode], [onlyreceiveMsgWithSentDate], [overridePreferencePOP3Host
[properties array])
Receive mails from pop3 account.

[Boolean](#) [#sendMail](#)([to[,to2,toN], [from[,reply], subject, msgText, [cc,cc2,ccN], [bcc,bcc2,bccN], [attachment/attachments array],
[overridePreferenceSMTPHost/properties array])
Send a mail, if you make the msgText start with <html> the message will be sent in html (and you can use all html formatting).

Method Details

[createBinaryAttachment](#)

[Attachment](#) [createBinaryAttachment](#)(filename, binarydata, [mimeType])
Creates a binary attachment object.

Parameters

filename
binarydata
[mimeType]

Returns

[Attachment](#)

Sample

```
var attachment1 = plugins.mail.createBinaryAttachment('logo1.gif',plugins.file.readFile('c:/temp/a_logo.gif'));
var attachment2 = plugins.mail.createBinaryAttachment('logo2.gif',plugins.file.readFile('c:/temp/another_logo.gif'));
var success = plugins.mail.sendMail('to_someone@example.com', 'John Cobb <from_me@example.org>', 'subject',
'msgText',null,null,new Array(attachment1,attachment2));
if (!success)
{
    plugins.dialogs.showWarningDialog('Alert','Failed to send mail','OK');
}
```

[createTextAttachment](#)

[Attachment](#) **createTextAttachment**(filename, textdata, [mimeType])

Creates a text based attachment object.

Parameters

filename

textdata

[mimeType]

Returns

[Attachment](#)

Sample

```
var attachment = plugins.mail.createTextAttachment('readme.html','<html>bla bla bla', 'text/html');
var success = plugins.mail.sendMail('to_someone@example.com', 'John Cobb <from_me@example.com>', 'subject',
'msgText', null, null, attachment);
if (!success)
{
    plugins.dialogs.showWarningDialog('Alert','Failed to send mail','OK');
}
```

[getLastSendMailExceptionMsg](#)

[String](#) **getLastSendMailExceptionMsg**()

Get the exception that occurred in the last sendMail attempt (null if no exception occurred).

Returns

[String](#)

Sample

```
var success = plugins.mail.sendMail('to_someone@example.com,to_someone_else@example.org', 'John Cobb
<from_me@example.com>', 'subject', 'my message', null, 'unnamed@example.com');
if (!success)
{
    plugins.dialogs.showWarningDialog('Alert',plugins.mail.getLastSendMailExceptionMsg(),'OK');
}
```

[getMailMessage](#)

[MailMessage](#) **getMailMessage**(binaryblob/string)

Helper method, returns MailMessage object from binary or 7bits string.

Parameters

binaryblob/string

Returns

[MailMessage](#)

Sample

```
var msg = plugins.mail.getMailMessage(myColumn);
if (msg != null) //if is null error occurred!
{
    application.output(msg.getFromAddresses())
}
```

[getPlainMailAddresses](#)

[String\[\]](#) **getPlainMailAddresses**(addressesString)

Helper method to only get the plain addresses.

Parameters

addressesString

Returns

[String\[\]](#)

Sample

```
var plainArray = plugins.mail.getPlainMailAddresses('John Cobb <from_me@example.com>;Pete
Cobb<from_pete@example.com>');
application.output(plainArray[0]) //will return 'from_me@example.com'
```

[isValidEmailAddress](#)

[Boolean](#) **isValidEmailAddress**(email)

Checks whether the given e-mail address is valid or not.

Parameters

email

Returns

Boolean

Sample

```
plugins.mail.isValidEmailAddress("me@example.com");
```

receiveMail

[MailMessage\[\]](#) **receiveMail**

(userName, password, leaveMsgsOnServer, [receiveMode], [onlyreceiveMsgWithSentDate], [overridePreferencePOP3Host/properties array])

Receive mails from pop3 account.

Parameters

userName

password

leaveMsgsOnServer

[receiveMode]

[onlyreceiveMsgWithSentDate]

[overridePreferencePOP3Host/properties array]

Returns

[MailMessage\[\]](#)

Sample

```
var msgs = plugins.mail.receiveMail('me', 'test', true);
if (msgs != null) //if is null error occurred!
{
    for (var i = 0 ; i < msgs.length ; i++)
    {
        var msg = msgs[i]
        application.output(msg.getFromAddresses())
        application.output(msg.getRecipientAddresses())
        application.output(msg.getReplyAddresses())
        application.output(msg.getReceivedDate())
        application.output(msg.getSentDate())
        application.output(msg.getHeaders())
        application.output(msg.getSubject())
        application.output(msg.getHtmlMsg())
        application.output(msg.getPlainMsg())
        var attachments = msg.getAttachments()
        if (attachments != null)
        {
            for (var j = 0 ; j < attachments.length ; j++)
            {
                var attachment = attachments[j]
                application.output(attachment.getName())
                var attachmentDataByteArray = attachment.getData()
                //write attachmentDataByteArray to a file...
            }
        }
    }
}

//it is also possible to first receive the headers and later receive a full message with particular 'sentdate'
//var receiveMode = 1;//0=FULL,1=HEADERS_ONLY,2=NO_ATTACHMENTS
//var msgs = plugins.mail.receiveMail('me', 'test', true ,receiveMode);

//when first did receive the headers(=all_field+subject,no body and no attachemnt), get a msg with a specific
sentdate
//var msgs = plugins.mail.receiveMail('me', 'test', true , 1 , theSentDateObjectFormPreviousHeaderLoading);

//it is possbile to set all kind of pop3 properties
//var properties = new Array()
//properties[0] = 'mail.pop3.host=myserver.com'
//properties specification can be found at:http://java.sun.com/products/javamail/javadocs/com/sun/mail/pop3
/package-summary.html
//var msgs = plugins.mail.receiveMail('me', 'test', true , 0 , null, properties);
```

sendMail

Boolean sendMail

([to[,to2,toN], [from[,reply], subject, msgText, [cc,cc2,ccN], [bcc,bcc2,bccN], [attachment/attachments array], [overridePreferenceSMTPHost/properties array])

Send a mail, if you make the msgText start with <html> the message will be sent in html (and you can use all html formatting).

Parameters

[to[,to2,toN]

[from[,reply]

subject

msgText

[cc,cc2,ccN]

[bcc,bcc2,bccN]

[attachment/attachments array]

[overridePreferenceSMTPHost/properties array]

Returns

Boolean

Sample

```
var attachment1 = plugins.mail.createBinaryAttachment('embedded',plugins.file.readFile('c:/temp/a_logo.gif'));
var msgText = 'plain msg<html>styled html msg</html>';
var success = plugins.mail.sendMail('to_someone@example.com,to_someone_else@example.net', 'John Cobb
<from_me@example.com>', 'subject', msgText,null,'unnamed@example.com',[attachment1]);
if (!success)
{
    plugins.dialogs.showWarningDialog('Alert','Failed to send mail','OK');
}

//it is possible to set all kind of smtp properties
//var properties = new Array()
//properties[0] = 'mail.smtp.host=myserver.com'
//properties specification can be found at:http://java.sun.com/products/javamail/javadocs/com/sun/mail/smtp
//package-summary.html
//var msgs = plugins.mail.sendMail('to_someone@example.com,to_someone_else@example.net', 'John Cobb
<from_me@example.com>', 'subject', msgText,null,'unnamed@example.com',null, properties);
```