

# MailMessage

## Method Summary

<a href="#">Attachment</a> []	<a href="#">#getAttachments()</a> Returns an array of Attachment instances corresponding to the attachments of this message.
<a href="#">String</a>	<a href="#">#getCCAddresses()</a> Returns a String with all addresses present in the CC field of this message.
<a href="#">String</a>	<a href="#">#getFromAddresses()</a> Returns a String with all addresses present in the From field of this message.
<a href="#">String</a>	<a href="#">#getHeaders()</a> Returns a String with all headers of this message.
<a href="#">String</a>	<a href="#">#getHtmlMsg()</a> Returns a String with the HTML content of this message.
<a href="#">String</a>	<a href="#">#getPlainMsg()</a> Returns a String with the plain content of this message.
<a href="#">Date</a>	<a href="#">#getReceivedDate()</a> Returns a Date instace corresponding to the moment when the message was received.
<a href="#">String</a>	<a href="#">#getRecipientAddresses()</a> Returns a String with all addresses in the To field of this message.
<a href="#">String</a>	<a href="#">#getReplyAddresses()</a> Returns a String with all addresses in the Reply-To field of this message.
<a href="#">Date</a>	<a href="#">#getSentDate()</a> Returns a Date instance corresponding to the moment when this message was sent.
<a href="#">String</a>	<a href="#">#getSubject()</a> Returns a String with the subject of this message.

## Method Details

[getAttachments](#)

[Attachment](#)[] **getAttachments()**

Returns an array of Attachment instances corresponding to the attachments of this message.

### Returns

[Attachment](#)[]

## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '        Name: ' + attachment.getName() + '\n';
                str += '        Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

[getCCAddresses](#)

**String** [getCCAddresses\(\)](#)

Returns a String with all addresses present in the CC field of this message.

**Returns**

[String](#)

## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '        Name: ' + attachment.getName() + '\n';
                str += '        Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

[getFromAddresses](#)

[String](#) **getFromAddresses()**

Returns a String with all addresses present in the From field of this message.

**Returns**

[String](#)

## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '    Name: ' + attachment.getName() + '\n';
                str += '    Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

getHeaders

[String](#) **getHeaders()**

Returns a String with all headers of this message.

**Returns**

[String](#)

## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '        Name: ' + attachment.getName() + '\n';
                str += '        Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

`getHtmlMsg`

[String](#) `getHtmlMsg()`

Returns a String with the HTML content of this message.

### Returns

[String](#)

## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '    Name: ' + attachment.getName() + '\n';
                str += '    Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

[getPlainMsg](#)

[String](#) **getPlainMsg()**

Returns a String with the plain content of this message.

**Returns**

[String](#)

## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '        Name: ' + attachment.getName() + '\n';
                str += '        Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

[getReceivedDate](#)

[Date](#) **getReceivedDate()**

Returns a Date instance corresponding to the moment when the message was received.

**Returns**

[Date](#)

## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '        Name: ' + attachment.getName() + '\n';
                str += '        Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

[getRecipientAddresses](#)

**String** [getRecipientAddresses\(\)](#)

Returns a String with all addresses in the To field of this message.

**Returns**

[String](#)



## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '        Name: ' + attachment.getName() + '\n';
                str += '        Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

[getReplyAddresses](#)

**String** [getReplyAddresses\(\)](#)

Returns a String with all addresses in the Reply-To field of this message.

**Returns**

[String](#)

## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '        Name: ' + attachment.getName() + '\n';
                str += '        Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

[getSentDate](#)

[Date](#) **getSentDate()**

Returns a Date instance corresponding to the moment when this message was sent.

**Returns**

[Date](#)

## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '    Name: ' + attachment.getName() + '\n';
                str += '    Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

[getSubject](#)

[String](#) **getSubject()**

Returns a String with the subject of this message.

**Returns**

[String](#)

## Sample

```
var msgs = plugins.mail.receiveMail(username, password, true, 0, null, properties);
if (msgs != null)
{
    for (var i=0; i < msgs.length; i++)
    {
        var msg = msgs[i];
        var str = '';
        str += 'From: ' + msg.getFromAddresses() + '\n';
        str += 'To: ' + msg.getRecipientAddresses() + '\n';
        str += 'CC: ' + msg.getCCAddresses() + '\n';
        str += 'Reply to: ' + msg.getReplyAddresses() + '\n';
        str += 'Received on: ' + msg.getReceivedDate() + '\n';
        str += 'Sent on: ' + msg.getSentDate() + '\n\n';
        str += 'Subject: ' + msg.getSubject() + '\n\n';
        str += 'Plain message: ' + msg.getPlainMsg() + '\n\n';
        str += 'HTML message: ' + msg.getHtmlMsg() + '\n\n';
        str += 'Headers: ' + msg.getHeaders() + '\n\n';
        var attachments = msg.getAttachments();
        if (attachments != null) {
            str += 'Number of attachments: ' + attachments.length + '\n\n';
            for (var j=0; j < attachments.length; j++)
            {
                var attachment = attachments[j];
                str += 'Attachment ' + j + '\n';
                str += '    Name: ' + attachment.getName() + '\n';
                str += '    Size: ' + attachment.getData().length + '\n\n';
            }
        }
        plugins.file.writeTXTFile('msg' + i + '.txt', str);
        application.output('Message ' + i + ' retrieved.');
```

```
    }
}
else
{
    application.output("Failed to retrieve messages.");
}
```