

# Attachment

## Method Summery

`byte[]` [#getData\(\)](#)  
Returns a byte array with the content of this attachment.

`String` [#getMimeType\(\)](#)  
Returns the Mime type of this attachment.

`String` [#getName\(\)](#)  
Returns the name of this attachment.

`Boolean` [#isEmbedded\(\)](#)  
Returns true if this attachment is embedded, false otherwise.

## Method Details

`getData`

`byte[]` **`getData()`**

Returns a byte array with the content of this attachment.

**Returns**

`byte[]`

**Sample**

```
var logo = plugins.mail.createBinaryAttachment('logo.jpg', plugins.file.readFile('d:/logo.jpg'));
var invoice = plugins.mail.createTextAttachment('invoice.txt', plugins.file.readTXTFile('d:/invoice.txt'));
var attachments = new Array(logo, invoice);
var success = plugins.mail.sendMail(toAddress, fromAddress, 'subject line', 'message text', null, null,
attachments, properties);
if (!success)
{
    plugins.dialogs.showWarningDialog('Alert', 'Failed to send mail', 'OK');
}
else
{
    plugins.dialogs.showInfoDialog('Success', 'Mail sent', 'OK');
    application.output('logo attachment name: ' + logo.getName());
    application.output('logo attachment mime type: ' + logo.getMimeType());
    application.output('logo attachment size: ' + logo.getData().length);
    application.output('logo attachment embedded state: ' + logo.isEmbedded());
    application.output('invoice attachment name: ' + invoice.getName());
    application.output('invoice attachment mime type: ' + invoice.getMimeType());
    application.output('invoice attachment size: ' + invoice.getData().length);
    application.output('invoice attachment embedded state: ' + invoice.isEmbedded());
}
```

`getMimeType`

`String` **`getMimeType()`**

Returns the Mime type of this attachment.

**Returns**

`String`

## Sample

```
var logo = plugins.mail.createBinaryAttachment('logo.jpg', plugins.file.readFile('d:/logo.jpg'));
var invoice = plugins.mail.createTextAttachment('invoice.txt', plugins.file.readTXTFile('d:/invoice.txt'));
var attachments = new Array(logo, invoice);
var success = plugins.mail.sendMail(toAddress, fromAddress, 'subject line', 'message text', null, null,
attachments, properties);
if (!success)
{
    plugins.dialogs.showWarningDialog('Alert', 'Failed to send mail', 'OK');
}
else
{
    plugins.dialogs.showInfoDialog('Success', 'Mail sent', 'OK');
    application.output('logo attachment name: ' + logo.getName());
    application.output('logo attachment mime type: ' + logo.getMimeType());
    application.output('logo attachment size: ' + logo.getData().length);
    application.output('logo attachment embedded state: ' + logo.isEmbedded());
    application.output('invoice attachment name: ' + invoice.getName());
    application.output('invoice attachment mime type: ' + invoice.getMimeType());
    application.output('invoice attachment size: ' + invoice.getData().length);
    application.output('invoice attachment embedded state: ' + invoice.isEmbedded());
}
```

## getName

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

Returns the name of this attachment.

## Returns

**String**

## Sample

```
var logo = plugins.mail.createBinaryAttachment('logo.jpg', plugins.file.readFile('d:/logo.jpg'));
var invoice = plugins.mail.createTextAttachment('invoice.txt', plugins.file.readTXTFile('d:/invoice.txt'));
var attachments = new Array(logo, invoice);
var success = plugins.mail.sendMail(toAddress, fromAddress, 'subject line', 'message text', null, null,
attachments, properties);
if (!success)
{
    plugins.dialogs.showWarningDialog('Alert', 'Failed to send mail', 'OK');
}
else
{
    plugins.dialogs.showInfoDialog('Success', 'Mail sent', 'OK');
    application.output('logo attachment name: ' + logo.getName());
    application.output('logo attachment mime type: ' + logo.getMimeType());
    application.output('logo attachment size: ' + logo.getData().length);
    application.output('logo attachment embedded state: ' + logo.isEmbedded());
    application.output('invoice attachment name: ' + invoice.getName());
    application.output('invoice attachment mime type: ' + invoice.getMimeType());
    application.output('invoice attachment size: ' + invoice.getData().length);
    application.output('invoice attachment embedded state: ' + invoice.isEmbedded());
}
```

## isEmbedded

**Boolean** [isEmbedded\(\)](#)

Returns true if this attachment is embedded, false otherwise. Attachments become embedded if they are references through tags from the body text of the message.

## Returns

**Boolean**

## Sample

```
var logo = plugins.mail.createBinaryAttachment('logo.jpg', plugins.file.readFile('d:/logo.jpg'));
var invoice = plugins.mail.createTextAttachment('invoice.txt', plugins.file.readTXTFile('d:/invoice.txt'));
var attachments = new Array(logo, invoice);
var success = plugins.mail.sendMail(toAddress, fromAddress, 'subject line', 'message text', null, null,
attachments, properties);
if (!success)
{
    plugins.dialogs.showWarningDialog('Alert', 'Failed to send mail', 'OK');
}
else
{
    plugins.dialogs.showInfoDialog('Success', 'Mail sent', 'OK');
    application.output('logo attachment name: ' + logo.getName());
    application.output('logo attachment mime type: ' + logo.getMimeType());
    application.output('logo attachment size: ' + logo.getData().length);
    application.output('logo attachment embedded state: ' + logo.isEmbedded());
    application.output('invoice attachment name: ' + invoice.getName());
    application.output('invoice attachment mime type: ' + invoice.getMimeType());
    application.output('invoice attachment size: ' + invoice.getData().length);
    application.output('invoice attachment embedded state: ' + invoice.isEmbedded());
}
```