

## ASP-Mail Kurzanleitung

### ASPMail Methoden

[SendMail](#)

[ClearAttachments](#)

[AddRecipient](#)

[ClearBodyText](#)

[ClearRecipients](#)

[AddExtraHeader](#)

[ClearCCs](#)

[ClearExtraHeaders](#)

[AddBCC](#)

[GetBodyTextFromFile](#)

[ClearBCCs](#)

[EncodeHeader](#)

[ClearAllRecipients](#)

[GetTempPath](#)

[AddAttachment](#)

---

#### SendMail

Mailer.SendMail Ergebnis: True / False

Die SendMail Method versucht, eine Email zu versenden. SendMail benötigt keine Parameter. Sie erzeugen die Nachricht mit den Properties FromAddress, Subject, BodyText und RemoteHost sowie der Methode AddRecipient.

```
Beispiel:  
if Mailer.SendMail then  
  ' Nachricht wurde gesendet  
else  
  Response.Write ("Fehler: " & Mailer.Response)  
end if
```

---

#### AddRecipient

AddRecipient (Name, Adresse) Ergebnis: True / False

Fügt einen neuen Empfänger der To-Liste hinzu. Das Feld 'Name' muß nicht ausgefüllt werden.

```
Beispiel:  
returnValue = Mailer.AddRecipient("Jay Jones", "jay@somehost.net")  
if returnValue then  
  ' Jay Jones wurde erfolgreich hinzugefügt  
end if
```

---

#### ClearRecipients

Entfernt jeden Empfänger aus der Empfängerliste.

Ergebnis: Nothing

---

#### AddCC

AddCC (Name, Adresse) Ergebnis: True / False

Fügt einen neuen Empfänger der Nachricht der "CC"-Liste hinzu. Der Parameter 'name' muß nicht angegeben werden.

```
Beispiel:  
returnValue = Mailer.AddCC ("", "jay@somehost.net")  
if returnValue then
```

```
' jayj@somehost.net' wurde hinzugefügt  
end if
```

---

#### ClearCCs

Entfernt jeden Empfänger der "CC"-Liste.

Ergebnis: Nothing

---

#### AddBCC

AddBCC (Name, Adresse) Ergebnis: True / False

Fügt einen neuen "unsichtbaren" (BCC) Empfänger hinzu. BCC Empfänger werden nicht in der Nachrichten-Empfängerliste aufgeführt.

Beispiel:  
returnValue = Mailer.AddBCC ("Jay Jones", "jayj@somehost.net")  
if returnValue then  
' Jay Jones wurde in die BCC Liste eingefügt  
end if

---

#### ClearBCCs

Entfernt jeden Empfänger von der "BCC"-Liste.

Ergebnis: Nothing

---

#### ClearAllRecipients

Entfernt alle Empfänger der "To"-, "CC"- und "BCC"-Listen.

Ergebnis: Nothing

---

#### AddAttachment

AddAttachment (Dateipfad) Ergebnis: Nothing

Hängt ein Attachment an die aktuelle Email an. Es ist nicht notwendig, den absoluten Pfad zu diesem Attachment anzugeben.

Beispiele:  
Mailer.AddAttachment ("D:\webs\mysite\downloads\myfile.zip")  
Mailer.AddAttachment (Server.MapPath("/downloads/myFile.zip"))

---

#### ClearAttachments

Löscht alle angehängenen Attachments.

Ergebnis: Nothing

---

#### ClearBodyText

Entfernt jeglichen Text, welcher zuvor durch den BodyText Property hinzugefügt wurde.

Ergebnis: Nothing

---

#### AddExtraHeader

AddExtraHeader (Header) Ergebnis: True / False

Fügt einen extra X-Header der Email zu. 'Header' ist eine Codezeile, welche einen echten SMTP X-Header generiert.

Beispiel:  
returnValue = Mailer.AddExtraHeader("X-HeaderName: XHdrValue")

---

#### ClearExtraHeaders

Entfernt jeglichen X-Headers, welcher durch AddExtraHeader hinzugefügt wurde.

---

### GetBodyTextFromFile

GetBodyTextFromFile (Dateipfad)

Fügt den Nachrichtentext aus einer Textdatei ein. Der Pfad der Datei muß absolut referenziert werden.

Beispiele:

Mailer.GetBodyTextFromFile ("d:\webs\mysite\data\msgbody.txt")  
Mailer.GetBodyTextFromFile (Server.MapPath("/data/msgbody.txt"))

---

### EncodeHeader

EncodeHeader (strValue) Ergebnis: Encoded string

Verschlüsselt ein String im RFC1522-Format, um einen 8bit Mail Header zu unterstützen, wie den 8bit Betreff-Headers. 'strValue' ist der String zum Verschlüsseln.

Beispiel:

Mailer.Subject = Mailer.EncodeHeader("Réponse de Service à la clientèle")

---

### GetTempPath

Gibt den Pfad für temporäre Mail-Dateien zurück, welcher über das Betriebssystem festgelegt wurde.

Ergebnis: Path string

Beispiel:

tempPath = Mailer.GetTempPath

## ASPMail Eigenschaften

[BodyText](#)

[CharSet](#)

[ConfirmRead](#)

[ContentType](#)

[CustomCharSet](#)

[DateTime](#)

[Encoding](#)

[FromName](#)

[FromAddress](#)

[Organization](#)

[Priority](#)

[RemoteHost](#)

[ReplyTo](#)

[Response](#)

[ReturnReceipt](#)

[SMTPLog](#)

[Subject](#)

[SuppressMsgBody](#)

[TimeOut](#)

[Urgent](#)

[UseMSMailHeaders](#)

[Version](#)

[WordWrap](#)

[WordWrapLen](#)

---

### BodyText

Der Nachrichtentext. Um den Inhalt zu entfernen, müssen Sie die Method ClearBodyText einsetzen.

Beispiel:

Mailer.BodyText = "Ihr Auftrag wird bearbeitet"

### CharSet

Der verwendete Zeichensatz. Grundeinstellung ist US ASCII.

Zulässige Werte sind:

- 1 = US Ascii
- 2 = ISO-8859-1

Beispiel:

Mailer.CharSet = 2

---

### ConfirmRead

Das Lesebestätigungs-Attribut.

Ist dieses Attribut auf TRUE gesetzt, das Emailprogramm des Empfängers unterstützt diese Funktion und der Empfänger hat diese Funktion eingeschaltet, dann wird das Emailprogramm des Empfängers eine Lesebestätigung an die FromAddress senden.

Beispiel:

Mailer.ConfirmRead = true

---

### ContentType

Das Option ContentType legt den Datentyp der Nachricht im ContentType header fest. Wollen Sie z.B. eine Nachricht im HTML-Format versenden, können Sie die Einstellung wie folgt treffen: ContentType="text/html". Unterstützt das Empfängerprogramm diesen Datentyp, wird Ihre Email als HTML-Seiteausgegeben.

Achtung: Der Datentyp wird ignoriert, wenn Sie eine Datei an die Email anhängen.

Beispiel:

Mailer.ContentType = "text/html"

---

### CustomCharSet

Wünschen Sie einen weiteren Zeichensatz neben den Standardzeichensätzen zu nutzen, können Sie mitCustomCharSet einen weiteren Zeichensatz einstellen.

Beispiel:

Mailer.CustomCharSet = "ISO-2022" or  
Mailer.CustomCharSet = "big5"

---

### DateTime

AspMail schreibt grundsätzlich eine Datum-/Zeitangabe auf Basis der GMT-Zeitzone in den Nachrichtenkopf. Wenn Sie diesen Eintrag überschreiben wollen, müssen Sie dazu mit dem DateTime Property eine gültige Zeitangabe auf Basis des Formates "RFC 822 & RFC 1123" einfügen.

Beispiel:

Mailer.DateTime = "Fri, 02 May 1997 10:53:49 -0500"

---

### Encoding

Der Verschlüsselungstyp für Attachments. Grundeinstellung ist MIME.

Zulässige Werte sind:

- 1 = UUEncoded
- 2 = MIME

Beispiel:

Mailer.Encoding = 1

---

### FromName

Der Absendername.

Beispiel:

Mailer.FromName = "Joe's Widget Shop"

---

#### FromAddress

Die Absender-Emailadresse.

Beispiel:

Mailer.FromAddress = "joe@widgets.com"

---

#### Organization

Setzt die Organisation-Variable in den Nachrichtenkopf.

Beispiel:

Mailer.Organization = "Your Company Name"

---

#### Priority

Setzt die Dringlichkeit der Nachricht. Die Prioritäten gehen von 1-5.

Zulässige Werte sind:

1 – High  
3 – Normal  
5 – Low

Beispiel:

Mailer.Priority = 1

---

#### RemoteHost

Gibt den entfernten SMTP-Server an, von dem aus die Nachricht versendet wird. Typischerweise ist das ein SMTP-Server bei Ihrem Zugangsprovider oder einem Internet SMTP-Server in Ihrem Unternehmensnetzwerk. Hier können bis zu 3 Serveradressen festgelegt werden, die durch ein Semikolon getrennt werden müssen. Ist der Hauptserver nicht erreichbar, wird der Versand über den zweiten Server der Liste durchgeführt. Ist dieser ebenfalls nicht erreichbar, tritt der dritte Server an diese Stelle.

Beispiel:

Mailer.RemoteHost = "mailhost.myisp.net"  
or  
Mailer.RemoteHost = "mailhost.myisp.net; mailhost.myotherisp.net"

---

#### ReplyTo

Die Option ReplyTo erlaubt es Ihnen, eine Antwortadresse zu definieren, die neben Ihrer eigentlichen Adresse in den Kopf eingetragen wird und an die eine Antwort auf Ihre Email gesendet wird. In der Grundeinstellung benutzt das Programm den "Reply-To:"-Header für die Antwort, wenn dieser spezifiziert wurde.

---

#### Response

Die Option Response liefert die Fehlermeldungen zurück, die unter Umständen auftreten.

---

#### ReturnReceipt

Das ReturnReceipt Attribut.  
Ist dieses auf TRUE gesetzt, der Empfänger-SMTP-Server unterstützt diese Funktion und diese Funktion ist auch eingeschaltet, so wird der SMTP-Server eine Zustellbestätigung an den Absender senden.

Beispiel:

Mailer.ReturnReceipt = false

---

#### SMTPLog

Sollten Sie wünschen, ein Programm zu überprüfen, so können Sie sich Fehlermeldungen in ein LogFile speichern lassen. Achten Sie dabei darauf, daß IUSR\_XYZ IIS - User die Schreibrechte besitzt.

Achtung: Nutzen Sie diese Einstellung nicht, wenn gleichzeitig mehrere Nutzer Zugriff auf dieses Programm haben. Diese Funktion ist ausschließlich für einen Nutzer gedacht.

Beispiel:

Mailer.SMTPLog = "c:\smtplog.txt"

---

### Subject

Der Betreff der Nachricht.

Beispiel:

Mailer.Subject = "Stock split announced!"

---

### SuppressMsgBody

Die Option SuppressMsgBody ist in der Grundeinstellung auf TRUE gesetzt und wird zusammen mit der Option SMTPLog eingesetzt. Ist unter SMTPLog eine bestimmte Datei festgelegt und SuppressMsgBody ist eingeschaltet, so wird neben dem Fehler auch noch eine Kopie der Nachricht gespeichert. Ist SuppressMsgBody ausgeschaltet, wird der Inhalt der Nachricht nicht an das LogFile übergeben.

---

### Timeout

Timeout legt die Zeit fest, die AspMail auf eine Antwort des Empfänger-Servers wartet. Grundeinstellung sind 30 Sekunden.

Beispiel:

Mailer.Timeout = 15

---

### Urgent

Das Urgent Attribut setzt den X-Urgent-Header in die zu versendende Nachricht. Nicht alle Empfängerprogramme unterstützen dieses Attribut.

Beispiel:

Mailer.Urgent = true

---

### UseMSMailHeaders

MS-Mail Priority Headers wird in der Grundeinstellung zusätzlich zu dem Standard SMTP Priority Headers gesendet. Mit diesem Befehl können Sie die Standardeinstellung ausstellen.

Beispiel:

Mailer.UseMSMailHeaders = false

---

### Version

Gibt die Versionsnummer aus.

Beispiel:

Response.Write "Component Version: " & Mailer.Version

---

### WordWrap

Die Option WordWrap ist in der Grundeinstellung ausgestellt. Setzen Sie das WordWrap auf TRUE, so wird der Nachrichtentext an der Stelle umgebrochen, die durch WordWrapLen definiert wird.

---

### WordWrapLen

Das WordWrapLen Property ist in der Grundeinstellung auf 70 Zeichen eingestellt. Sie können den Zeilenumbruch durch diesen Wert ändern.