

ASP: Send A Letter With SendLetter()

Instructions

1. Create a new ASP page.
2. Copy the entire code below and paste the code in the ASP page.
3. Replace *C:\MyFile.pdf* with a path to your test PDF file. Remember, for method SendLetter you must use a document where the recipient's address already appears so it will show through the envelope window. You can use one of our Sample Letter Documents.
4. Replace *MyUsername* and *MyPassword* with your PostalMethods user details.
5. Start your application. It will start and send a letter to the PostalMethods SendLetter Web Service method. If the response is a positive number, you will be able to see your letter in the PostalMethods Control Panel.

If the response is negative number, check the Web Service Status Codes section.

That's it - you have successfully sent a letter to PostalMethods. Congratulations.

```
<%
'
' This asp script demonstrates how to send a binary file
' (such as .PDF, .DOC) from an ASP script, using MS SOAP 3.0
' and the SendLetter method within PostalMethods. As asp has no
' built in provisions to reading binary files, we use an ADODB.Stream
' object to do so.
'

Option Explicit

Const Filename = "C:\MyFile.pdf"
Const MyUsername = "MyUsername"
Const MyPassword = "MyPassword"

Dim objSoap, lngResult
Dim B
'
' Read your binary document.
'
B = ReadBinaryFile(Filename)
'
'Create the SoapClient object      'MS SOAP v3.0
'
Set objSoap = Server.CreateObject("MSSOAP.SOAPClient30")
'
'Set to True when an Active Server Pages (ASP) application or an ISAPI DLL uses the
SoapClient object.
'
objSoap.ClientProperty("ServerHttpRequest") = True
'
'Initializes the SoapClient object. ASP initializes the WSDL on each call.
'For better performance, you may download the WSDL locally as an XML and refer to the
local file.
'
objSoap.mssoapinit("http://api.postalmethods.com/2009-02-26/PostalWS.asmx?WSDL")
'
'Set to True when a proxy server is to be detected automatically
'
```

```

objSoap.ConnectorProperty("EnableAutoProxy") = True
'
' Invoke the SendLetter method
'
lngResult = objSoap.SendLetter( _
    MyUsername, _
    MyPassword, _
    "Send a letter using ASP", _
    "pdf", _
    B, _
    "Default")

If CLng(lngResult) > 0 Then
    WriteLine("LetterID is: " & lngResult)
Else
    WriteLine("Error is:" & lngResult)
End If

'*****
' Utility methods
'*****

Function ReadBinaryFile(FileName)
    Const adTypeBinary = 1

    'Create Stream object
    Dim BinaryStream
    Set BinaryStream = CreateObject("ADODB.Stream")

    'Specify stream type - we want To get binary data.
    BinaryStream.Type = adTypeBinary

    'Open the stream
    BinaryStream.Open

    'Load the file data from disk To stream object
    BinaryStream.LoadFromFile FileName

    'Open the stream And get binary data from the object
    ReadBinaryFile = BinaryStream.Read

    'Clean up
    BinaryStream.Close
    Set BinaryStream = Nothing
End Function

Sub WriteLine(T)
    Response.Write T & "<BR>"
End Sub

'Note: If you are faxing from behind a proxy using MS SOAP v.3, you can use one of the
two following options:

'Option 1 - Use automatic detection by:
'    objSoap.ConnectorProperty("EnableAutoProxy") = True

'Option 2 - Specify a proxy server:
'    objSoap.ConnectorProperty("ProxyServer") = "192.168.0.100"
'and optionally, if the proxy server is password-protected specify:
'    objSoap.ConnectorProperty("ProxyUser") = "Proxy Username"
'    objSoap.ConnectorProperty("ProxyPassword") = "Proxy Password"

%>

```

ASP: Send A Letter with SendLetterAndAddress()

Instructions

1. Create a new ASP page.
2. Copy the entire code below and paste the code inside the ASP page.
3. Replace *C:\MyFile.pdf* with a path to your test PDF file. Remember, method `SendLetterAndAddressV2` will add the recipient's address so it would show through the envelope window. You can use one of our Sample Letter Documents.
4. Replace *MyUsername* and *MyPassword* with your PostalMethods user details.
5. Replace all the address parameters with real address values. Leave an empty string ("") where no value is needed.
6. Start your application. It will start and send a letter to the PostalMethods `SendLetterAndAddressV2` Web Service method. If the response is a positive number, you will be able to see your letter in the PostalMethods Control Panel. If the response is negative number, check the Web Service Status Codes section.

That's it - you have successfully sent a letter to PostalMethods. Congratulations.

```
<%
'
' This asp script demonstrates how to send a binary file
' (such as .PDF, .DOC) from an ASP script, using MS SOAP 3.0
' and the SendLetterAndAddressV2 method within PostalMethods. As asp has no
' built in provisions to reading binary files, we use an ADODB.Stream
' object to do so.
'
Option Explicit

Const Filename = "C:\MyFile.pdf"
Const MyUsername = "MyUsername"
Const MyPassword = "MyPassword"
Const Description = "Send a letter using ASP"
Const AttentionLine1 = "Mr. Danny Kay"
Const AttentionLine2 = "Director, Dancing Dept."
Const CompanyName = "Comedy Makers"
Const AddressLine1 = "1234 Main ST."
Const AddressLine2 = "1st FL."
Const City = "Brooklyn"
Const State = "NY"
Const PostalCodeZIP = "11201"
Const Country = "USA"

Dim objSoap, lngResult
Dim B
'
' Read your binary document.
'
B = ReadBinaryFile(Filename)
'
'Create the SoapClient object      'MS SOAP v3.0
'
Set objSoap = Server.CreateObject("MSSOAP.SOAPClient30")
'
'Set to True when an Active Server Pages (ASP) application or an ISAPI DLL uses the
SoapClient object.
'
objSoap.ClientProperty("ServerHttpRequest") = True
```

```

'
'Initializes the SoapClient object. ASP initializes the WSDL on each call.
'For better performance, you may download the WSDL locally as an XML and refer to the
local file.
'
objSoap.mssoapinit("http://api.postalmethods.com/2009-02-26/PostalWS.asmx?WSDL")
'
'Set to True when a proxy server is to be detected automatically
'
objSoap.ConnectorProperty("EnableAutoProxy") = True
'
' Invoke the SendLetterAndAddress() method
'
lngResult = objSoap.SendLetterAndAddress( _
    MyUsername, _
    MyPassword, _
    Description, _
    "pdf", _
    B, _
    "Default",
    AttentionLine1, _
    AttentionLine2, _
    CompanyName, _
    AddressLine1, _
    AddressLine2, _
    City, _
    State, _
    PostalCodeZIP, _
    Country)

If CLng(lngResult) > 0 Then
    WriteLine("LetterID is: " & lngResult)
Else
    WriteLine("Error is:" & lngResult)
End If

'*****
'    Utility methods
'*****

Function ReadBinaryFile(FileName)
    Const adTypeBinary = 1

    'Create Stream object
    Dim BinaryStream
    Set BinaryStream = CreateObject("ADODB.Stream")

    'Specify stream type - we want To get binary data.
    BinaryStream.Type = adTypeBinary

    'Open the stream
    BinaryStream.Open

    'Load the file data from disk To stream object
    BinaryStream.LoadFromFile FileName

    'Open the stream And get binary data from the object
    ReadBinaryFile = BinaryStream.Read

    'Clean up
    BinaryStream.Close
    Set BinaryStream = Nothing
End Function

```

```

Sub WriteLine(T)
    Response.Write T & "<BR>"
End Sub

'Note: If this script is running from behind a proxy using MS SOAP v.3, you can use
one of the two following options:

'Option 1 - Use automatic detection by:
'    objSoap.ConnectorProperty("EnableAutoProxy") = True

'Option 2 - Specify a proxy server:
'    objSoap.ConnectorProperty("ProxyServer") = "192.168.0.100"
'and optionally, if the proxy server is password-protected specify:
'    objSoap.ConnectorProperty("ProxyUser") = "Proxy Username"
'    objSoap.ConnectorProperty("ProxyPassword") = "Proxy Password"

%>

```

ASP: Cancel Delivery Of A Letter with CancelDelivery()

Instructions

1. Create a new ASP page.
2. Copy the entire code below and paste the code there.
3. Replace *MyUsername* and *MyPassword* with your PostalMethods user details.
4. Replace *ID* with the LetterID you would like to cancel.
5. Start your application. If the response is -3000, PostalMethods is trying to cancel your letter. You should query its status. If the response is any other number, check the Web Service Status Codes section.

```

<%
'
' This asp script demonstrates how to cancel a delivery of a Letter, using MS SOAP 3.0
' and the CancelDelivery method within PostalMethods.
'
Option Explicit

Const MyUsername = "MyUsername"
Const MyPassword = "MyPassword"
' Replace with the LetterID of the Letter, as returned by the sending method, such as
SendLetter()
Const LetterID="ID"

Dim objSoap, lngResult
'
'Create the SoapClient object      'MS SOAP v3.0
'
Set objSoap = Server.CreateObject("MSSOAP.SOAPClient30")
'
'Set to True when an Active Server Pages (ASP) application or an ISAPI DLL uses the
SoapClient object.
'
objSoap.ClientProperty("ServerHTTPRequest") = True
'
'Initializes the SoapClient object. ASP initializes the WSDL on each call.
'For better performance, you may download the WSDL locally as an XML and refer to the
local file.

```

```
'
objSoap.mssoapinit("http://api.postalmethods.com/2009-02-26/PostalWS.asmx?WSDL")
'
'Set to True when a proxy server is to be detected automatically
'
objSoap.ConnectorProperty("EnableAutoProxy") = True
'
' Invoke the SendLetter method
'
lngResult = objSoap.CancelDelivery( _
                                MyUsername, _
                                MyPassword, _
                                LetterID)

If CLng(lngResult) = -3000 Then
    '
    ' -3000 means the request was successfully marked for cancellation.
    '
    Response.Write "Letter was requested to be canceled"
Else
    '
    ' A value less than -3000 means an error occurred.
    ' See the PostalMethods Status Codes:
    http://www.postalmethods.com/statuscodes.
    '
    Response.Write "Error is:" & lngResult
End If

%>
```