Standard Operating Procedure (SOP) for Sending Mass Emails with Attachments Using VBA

<https://mailtrap.io/blog/how-to-send-mass-email-in-outlook/#How-to-send-personalized-bulk-emails-from-Outlook>

Purpose

This SOP outlines the steps to create a VBA script that sends personalized emails with attachments from an Excel spreadsheet.

Scope

This procedure applies to users who need to send bulk emails using Microsoft Excel and Outlook, specifically for the following data structure:

| **Customer Category** | **Customer Name** | **Contact Name** | **Contact Email** | **Attachment Path** |
| --- | --- | --- | --- | --- |

Prerequisites

* Microsoft Excel and Outlook installed and configured.
* Basic knowledge of navigating the Excel interface and using VBA.
* Ensure that macros are enabled in Excel.

Procedure

Step 1: Prepare Your Excel Sheet

1. **Create an Excel workbook** with the following columns:
   * **Column A**: Customer Name
   * **Column B**: Customer Mission
   * **Column C**: Contact First Name
   * **Column D**: Contact Last Name
   * **Column E**: Contact Email
   * **Column F**: Attachment Path (full path to the file you want to attach)
2. **Fill in the data** starting from row 2, leaving row 1 for headers.

Step 2: Open the VBA Editor

1. Press ALT + F11 in Excel to open the Visual Basic for Applications (VBA) editor.

Step 3: Insert a New Module

1. Click Insert > Module to create a new module.

Step 4: Write the VBA Code

Copy and paste the following code into the module:

text

Option Explicit

Private Declare PtrSafe Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" \_

(ByVal hwnd As LongPtr, ByVal lpOperation As String, \_

ByVal lpFile As String, ByVal lpParameters As String, \_

ByVal lpDirectory As String, ByVal nShowCmd As Long) As LongPtr

Sub SendMassEmails()

Dim ws As Worksheet

Dim lastRow As Long

Dim i As Long

Dim emailBody As String

Dim emailSubject As String

' Set the worksheet

Set ws = ThisWorkbook.ActiveSheet

' Find the last row with data

lastRow = ws.Cells(ws.Rows.Count, "A").End(xlUp).Row

' Ask user if they want to proceed

If MsgBox("This will open the new Outlook and create " & (lastRow - 1) & " emails." & vbNewLine & \_

"Please ensure you're logged into the new Outlook before proceeding." & vbNewLine & \_

"Continue?", vbQuestion + vbYesNo) = vbNo Then

Exit Sub

End If

' Open new Outlook

OpenNewOutlook

' Wait for Outlook to open

Application.Wait Now + TimeSerial(0, 0, 5)

' Loop through each row

For i = 2 To lastRow ' Assuming row 1 has headers

' Construct email body

emailBody = "Hi " & ws.Cells(i, 2).Value & "," & vbNewLine & vbNewLine & \_

"I'm thrilled to introduce myself as your new contact at Suniture!" & vbNewLine & vbNewLine & \_

"I'd love to catch up and hear about your progress on new projects since we last spoke about " & ws.Cells(i, 4).Value & "..." & vbNewLine & vbNewLine & \_

"How does a quick introduction/update meeting this week or next sound? This would be a great opportunity for us to connect and for you to share your insights on our latest offerings, visually stunning for the very best—high-end." & vbNewLine & vbNewLine & \_

"At Suniture, we specialize in creating beautiful outdoor spaces with our premium, weather-resistant furniture, sunbeds, sunbuns, gazebos and umbrellas. Warranty®" & vbNewLine & vbNewLine & \_

"\*\*Think chic aesthetics that hold up beautifully, no matter the weather.\*\*" & vbNewLine & vbNewLine & \_

"Looking forward to reconnecting!" & vbNewLine & vbNewLine & \_

"Best" & vbNewLine & \_

"Hugo"

' Construct email Subject

emailSubject = "Following up with " & ws.Cells(i, 1).Value & " and Suniture"

' Create new email

CreateNewEmail ws.Cells(i, 3).Value, \_

emailSubject, \_

emailBody

' Wait between emails

Application.Wait Now + TimeSerial(0, 0, 2)

Next i

MsgBox "Process completed! Please check your draft folder in the new Outlook.", vbInformation

End Sub

Private Sub OpenNewOutlook()

Dim result As LongPtr

' Try to open new Outlook using default mail protocol

result = ShellExecute(0, "open", "mailto:", "", "", 1)

If result <= 32 Then ' Error occurred

MsgBox "Could not open new Outlook. Please ensure it's set as your default email client.", vbCritical

End

Else

'Close the draft email that was opened

SendKeys "%{F4}", True 'Alt+F4 to close the window

End If

End Sub

Private Sub CreateNewEmail(RecipientEmail As String, emailSubject As String, emailBody As String)

' Construct mailto URL

Dim mailtoURL As String

mailtoURL = "mailto:" & RecipientEmail & "?subject=" & Replace(emailSubject, " ", "%20") & \_

"&body=" & Replace(Replace(emailBody, vbNewLine, "%0D%0A"), " ", "%20")

' Open new email

ShellExecute 0, "open", mailtoURL, "", "", 1

' Wait for email window to open

Application.Wait Now + TimeSerial(0, 0, 1)

' Send keyboard shortcuts to create the email

SendKeys "^{ENTER}", True ' Ctrl+Enter to send to drafts

End Sub

Step 5: Customize Your Email Content

* Modify MailSubject and MailBody in row 2 of your Excel sheet as needed.
* Ensure that any placeholder text (like «Contact\_Name») is replaced dynamically in the body of the email.

Step 6: Run the Macro

1. Close the VBA editor.
2. Press ALT + F8, select SendMassEmailsWithAttachments, and click Run.

Important Notes:

* **Test with a Small Dataset**: Before sending out mass emails, test with a small number of entries to ensure everything works correctly.
* **Check Security Settings**: Depending on your organization's security settings, you may need to adjust permissions for macros and programmatic access to Outlook.
* **Attachments**: Ensure that the file paths provided in your Excel sheet are correct and accessible.