

Parsing VCard file

Today we had a client who needed to read in VCard files. The following script shows an early version of the script which loops over the desktop folder to import VCF file. For testing there was only one file there, so we skipped creating record as we simply wrote always in the same record.



```
# Loop over files on desktop
Set Variable [ $folder ; Value: MBS( "Folders.UserDesktop" ) ]
Set Variable [ $files ; Value: MBS( "Files.List"; $folder; 1+4; ".vcf" ) ]
If [ MBS("IsError") = 0 ]
  Set Variable [ $count ; Value: ValueCount ( $files ) ]
  Set Variable [ $index ; Value: 1 ]
  If [ $count > 0 ]
    Loop
      # Read in vcard
      Set Variable [ $filename ; Value: GetValue($files; $index) ]
      Set Variable [ $filepath ; Value: MBS( "Path.AddPathComponent"; $folder; $filename ) ]
      Set Variable [ $text ; Value: MBS( "Text.ReadTextFile"; $FilePath; "UTF-8" ) ]
      If [ MBS("IsError") = 0 ]
        # Normalize end of line characters
        Set Variable [ $text ; Value: MBS( "Text.ReplaceNewline"; $text; 1) ]
        # Create new record here
        # Process all text lines
        Set Variable [ $LineCount ; Value: ValueCount ( $text ) ]
        Set Variable [ $LineIndex ; Value: 1 ]
        Loop
          Set Variable [ $Line ; Value: GetValue($text; $lineindex) ]
          If [ Left ( $line ; 2 ) = "N:" ]
            # Name
            Set Variable [ $z ; Value: Middle ( $line ; 3; Length ( $line ) ) ]
            Set Variable [ $list ; Value: MBS( "List.CSVSplit"; $z ) ]
            Set Variable [ $FirstName ; Value: GetValue($list; 1) ]
            Set Variable [ $SureName ; Value: GetValue($list; 2) ]
            Set Field [ CON::nameGiven ; $firstName ]
            Set Field [ CON::nameFamily ; $sureName ]
          Else If [ Left($line; 6) = "EMAIL;" ]
            # todo
          Else If [ Left($line; 27) = "PHOTO;ENCODING=b;TYPE=JPEG:" ]
            # Photo as base64 coded JPEG:
            Set Variable [ $data ; Value: Middle ( $line ; 28; Length ( $line ) ) ]
            Set Variable [ $LineIndex ; Value: $LineIndex + 1 ]
            Loop
              Set Variable [ $Line ; Value: GetValue($text; $lineindex) ]
              Exit Loop If [ Left ( $line ; 1 ) ≠ " " ]
              Set Variable [ $data ; Value: $data & ¶ & $line ]
            # next
            Set Variable [ $LineIndex ; Value: $LineIndex + 1 ]
            Exit Loop If [ $LineIndex > $LineCount ]
          End Loop
          Set Variable [ $LineIndex ; Value: $LineIndex - 1 ]
```

```
        Set Variable [ $image ; Value: Base64Decode ( $data; "image.jpg" ) ]
        Set Field [ CON::imageOrLogo ; $image ]
    End If
    # next
    Set Variable [ $LineIndex ; Value: $LineIndex + 1 ]
    Exit Loop If [ $LineIndex > $LineCount ]
End Loop
# Commit record
End If
# next
Set Variable [ $index ; Value: $index + 1 ]
Exit Loop If [ $index > $count ]
End Loop
End If
End If
```

To look in detail, we use [Files.List](#) to list the files. Mode 4 for visible items only and 1 to limit to files. We loop over all file names and use [Path.AddPathComponent](#) to add file name to the folder path. For the text we read it and replace all end of line characters with the mac version. When we hit the name entry in the lines, we get the first and last name. For a photo we need to collect all the lines for base64 decode and than display image in container.

The script is not yet ready and probably needs some more work to cover all keys, but it may be a good start for other people.