

証跡マクロ君_呼び出し.vbs

```

1 Option Explicit
2
3 Private Sub bt_DirSelect_Click()
4
5     Dim wk_str As String
6     Dim i_Count As Integer
7     'ディレクトリ選択画面を開く
8     With Application.FileDialog(msoFileDialogFolderPicker)
9         .AllowMultiSelect = False
10        .ButtonName = "フォルダ選択"
11        .Show
12
13        'ダイアログボックスにて選択されたディレクトリが
14        'あれば変数へ格納する。
15        wk_str = ""
16        If .SelectedItems.Count > 0 Then
17            ActiveSheet.Cells(3, 3) = .SelectedItems(1)
18        End If
19    End With
20
21
22
23 ' サンプル w
24 '   ファイル ダイアログを開きます。
25     Dim lngCount
26     With Application.FileDialog(msoFileDialogOpen)
27         .AllowMultiSelect = True
28         .Show
29
30         ' 選択された各ファイルのパスを表示します。
31         For lngCount = 1 To .SelectedItems.Count
32             MsgBox .SelectedItems(lngCount)
33         Next lngCount
34
35     End With
36
37 End Sub
38
39 Private Sub btCreate_Click()
40     Dim rcode As Integer
41
42     ' 基本情報の定義
43     Set fs = CreateObject("Scripting.FileSystemObject")
44     Set target_book = ThisWorkbook
45     book_path = target_book.Path
46
47
48     ' 証跡作成マクロの呼び出し
49     rcode = create_sheets
50
51 End Sub
52
53 Sub ガラ作成_Click()
54     Dim rcode As Integer
55
56     ' 基本情報の定義
57     Set fs = CreateObject("Scripting.FileSystemObject")
58     Set target_book = ThisWorkbook
59     book_path = target_book.Path
60
61
62     ' 証跡作成マクロの呼び出し
63     rcode = ガラ作成実行
64     ' Call ガラ作成
65 End Sub

```

取扱注意

```

1 Option Explicit
2
3 Public target_book As Workbook      '作業用ワークブック
4 Public book_path As String          '作業用ワークブックが格納されているディレクトリ
5 Public book_ebi As Workbook        '証跡用ワークブック
6 Public Target_dir As String         '証跡格納フォルダ
7 Public fs                            'ファイルシステムオブジェクトの格納
8
9 Public Const t_row As Integer = 3   '証跡フォルダ記載位置 (行)
10 Public Const t_col As Integer = 3  '証跡フォルダ記載位置 (列)
11 Public Const t_filetype As String = "*.JPG" '証跡対象ファイルタイプ (JPG)
12 Public Const t_outSet As String = "テキスト" '設定値の文言
13 Public Const t_ON As String = "ON"
14 Public Const t_numberLen As Integer = 7 '確認項目番号の長さ
15 Public Const t_p_num As String = "L3" '確認項目番号の記入位置
16 Public Const t_p_val As String = "L4" '確認項目内容の記入位置
17 Public Const t_p_out As String = "AD22" '確認項目内容の出力位置
18 Public Const t_p_str As String = "【確認項目内容】"
19 Public Const t_numberLen2 As Integer = 3 '項目番号の長さ
20 Public Const t_p_num2 As String = "L13" '項目番号の記入位置
21 Public Const t_p_val2 As String = "L14" '試験条件の記入位置
22 Public Const t_p_out2 As String = "AD3" '試験条件の出力位置
23 Public Const t_p_str2 As String = "【試験条件】"
24
25
26
27
28 Function create_sheets()
29     Dim t_count As Integer
30     Dim t_filename As String
31     Dim t_sheetname As String
32     Dim t_nameOut As String
33     Dim t_valueOut As String
34     Dim t_wide As Double
35     Dim t_wkData As String
36     Dim t_underCut As String
37     Dim t_sizebr As String
38
39     '-----
40     '設定値の取得
41     '-----
42     t_nameOut = target_book.Sheets(1).Range("H8")
43     t_valueOut = target_book.Sheets(1).Range("H10")
44     t_wide = target_book.Sheets(1).Range("H12")
45     t_underCut = target_book.Sheets(1).Range("H13")
46     t_sizebr = target_book.Sheets(1).Range("H14")
47
48     '-----
49     'ページ設定を切替える(設定が"テキスト"の場合か否か)
50     '-----
51     If t_nameOut = t_outSet Then
52         Call ヘッドOFF
53     Else
54         Call ヘッドON
55     End If
56
57
58     '-----
59     '証跡格納ディレクトリの特定
60     '-----
61     'エクセルシートにNewWorkbook証跡格納フォルダが記載されている場合、以下の処理を行う。
62     If target_book.Sheets(1).Cells(t_row, t_col) <> "" Then
63
64         '記載されているフォルダが存在するか確認する。
65         If fs.FolderExists(target_book.Sheets(1).Cells(t_row, t_col)) = True Then
66             'フォルダが存在した場合はTarget_dirにディレクトリを格納する
67             Target_dir = target_book.Sheets(1).Cells(t_row, t_col)
68
69         Else
70             'フォルダが存在しなかった場合、エラーメッセージを出力する。
71             MsgBox "指定した証跡フォルダ[" & target_book.Sheets(1).Cells(t_row, t_col) & "]" が見つかりませんでした。" _
72                 & "シートに書かれたフォルダを確認してください。" _
73                 & Chr(13) & "証跡がこのエクセルファイルと同じディレクトリに存在する場合は、" _
74                 & "シートにある「証跡フォルダ」を削除してください。", vbOKOnly + vbCritical
75             create_sheets = -1
76             Exit Function
77
78         End If
79     Else
80         'エクセルシートに証跡格納フォルダが記載されていない場合は、カレントディレクトリを格納する。
81         Target_dir = target_book.Path
82

```

証跡マクロ君.vbs

```

83 End If
84
85 MsgBox "デバッグメッセージ" & Chr(13) _
86 & "[ " & Target_dir & " ]"
87
88 ' 特定したディレクトリからファイルリストを取得する
89
90 t_filename = ""
91 t_filename = Dir(Target_dir & t_filetype, vbNormal)
92
93 ' 対象ファイルが存在しなかった場合
94 If t_filename = "" Then
95     ' 証跡ファイルが見つからない場合はそのまま終了
96     MsgBox "指定されたディレクトリ配下に証跡ファイルは見つけれませんでした。", _
97         vbOKOnly + vbExclamation
98     create_sheets = -2
99     Exit Function
100 End If
101
102 ' 以下、対象がひとつでも見つかった場合の処理
103
104 ' 新しいワークブックを開く
105
106 Excel.Application.SheetsInNewWorkbook = 1
107 Set book_ebi = Workbooks.Add
108
109
110 ' 新しいシートを作成し、証跡を貼り付けていく。
111
112 t_count = 1
113 Do While t_filename <> ""
114
115     ' シートを作成する。(1回目はシート1があるのでスキップさせる)
116     If t_count <> 1 Then
117         book_ebi.Sheets.Add After:=Worksheets(t_count - 1)
118     End If
119     target_book.Sheets(2).Copy After:=Worksheets(t_count)
120
121
122     ' 証跡を貼り付ける位置へフォーカスを移動する
123     book_ebi.Sheets(t_count).Cells(2, 2).Activate
124     book_ebi.Sheets(t_count + 1).Cells(2, 2).Activate
125
126     ' 証跡をシートへ貼り付ける
127     book_ebi.Sheets(t_count).Pictures.Insert(Target_dir & t_filename).Select
128     ActiveSheet.Pictures.Insert(Target_dir & t_filename).Select
129     ActiveSheet.Pictures.CopyPicture
130     ActiveSheet.Pictures.Delete
131     ActiveSheet.Paste
132
133     ' 自動縮小機能ONの場合、サイズを信用端末から取得したSSのサイズと同一サイズに横幅を調整する
134     If t_sizebr = "ON" Then
135
136         Selection.ShapeRange.LockAspectRatio = msoTrue
137
138         Selection.ShapeRange.Width = 767.96
139
140     End If
141
142
143     ' シート名を変更する。(ファイル名の拡張子無し部分をシート名にする)
144
145     t_sheetname = Left(t_filename, Len(t_filename) - 4)
146
147
148     ' ファイル名を左上に記入する。(設定が"テキスト"の場合)
149
150     If t_nameOut = t_outSet Then
151         ActiveSheet.Range("A1").Value = t_sheetname
152     End If
153
154
155     ' 確認項目内容と試験条件を追記する。(設定が"ON"の場合)
156
157     If t_valueOut = t_ON Then
158
159         ' 確認項目番号の取得
160         t_wkData = Left(t_sheetname, t_numberLen)
161
162
163         ' *** 確認内容の記入 ***
164         ' 確認項目番号を記入する => VLOOKUP関数の結果が出る

```

証跡マクロ君.vbs

```

165:         target_book.Sheets(1).Range(t_p_num).Value = t_wkData
166:         'コピー          => VLOOKUP関数の結果をコピー
167:         t_wkData = target_book.Sheets(1).Range(t_p_val).Cells.Text
168:         '値を貼付け
169:         ActiveSheet.Range(t_p_out).Value = t_wkData
170:         '文言を出力
171:         ActiveSheet.Range(t_p_out).Offset(-1, 0).Value = t_p_str
172:
173:
174:         '項目番号の取得
175:         t_wkData = Left(t_sheetname, t_numberLen2)
176:
177:         '**** 試験条件の記入 ****
178:         '確認項目番号を記入する => VLOOKUP関数の結果が出る
179:         target_book.Sheets(1).Range(t_p_num2).Value = t_wkData
180:         'コピー          => VLOOKUP関数の結果をコピー
181:         t_wkData = target_book.Sheets(1).Range(t_p_val2).Cells.Text
182:         '値を貼付け
183:         ActiveSheet.Range(t_p_out2).Value = t_wkData
184:         '文言を出力
185:         ActiveSheet.Range(t_p_out2).Offset(-1, 0).Value = t_p_str2
186:
187:     End If
188:
189:     '-----
190:     'シート名編集(アンダーカット)処理(追加開始)
191:     '-----
192:     Dim iStrPos As Integer
193:     iStrPos = InStrRev(t_sheetname, "_", -1, vbTextCompare)
194:     iStrPos = InStr(1, t_sheetname, "_", vbTextCompare)
195:     If (iStrPos <> 0) And (t_underCut = t_ON) Then
196:         t_sheetname = Left(t_sheetname, iStrPos - 1)
197:     End If
198:     '-----
199:     'シート名編集(アンダーカット)処理(追加終了)
200:     '-----
201:
202:     If Len(t_sheetname) > 31 Then
203:         'シート名は最大31bytesなので、31bytesを超える
204:         'ファイル名(拡張子抜きで)の場合は自動変更しない。
205:         MsgBox "ファイル名が31bytesを超えるため、シート名の変更ができませんでした。" & Chr(13) &
206:             & "ファイル名: " & t_filename, vbOKOnly + vbExclamation
207:     Else
208:         'ファイル名が31bytes以内の場合はシート名を変更する。
209:         book_ebi.Sheets(t_count).Name = Left(UCase(Left(t_sheetname, 3)) & Right(t_sheetname, Len(t_sheetname) - 3), 31)
210:     End If
211:     'シート名設定下限文字の解除(3文字未満でも処理可能な処理へ変更)
212:     '-----
213:     book_ebi.Sheets(t_count + 1).Name = Left(UCase(Left(t_sheetname, 3)) & Right(t_sheetname, Len(t_sheetname) - 3), 31)
214:     book_ebi.Sheets(t_count + 1).Name = UCase(t_sheetname)
215:     End If
216:
217:     '-----
218:     'シートの横幅調節(追加開始)
219:     '-----
220:     book_ebi.Sheets(t_count + 1).Columns("A:CQ").ColumnWidth = t_wide
221:     '-----
222:     'シートの横幅調節(追加終了)
223:     '-----
224:
225:     '-----
226:     'A1セルへ証跡番号設定(追加開始)
227:     '-----
228:     Dim sEbiNo As String
229:     sEbiNo = Mid(t_sheetname, 1, 15) & "-" & Mid(t_sheetname, 13, Len(t_sheetname) - 12)
230:     book_ebi.Sheets(t_count + 1).Range("A1").Value = t_sheetname
231:
232:     'A1セルへ証跡番号設定(追加終了)
233:     '-----
234:
235:
236:     '-----
237:     '印刷イメージを整える
238:     '-----
239:     'シートの印刷設定(用紙:横1ページに収める設定)
240:     With book_ebi.Sheets(t_count).PageSetup
241:         .LeftHeader = ""
242:         .CenterHeader = ""
243:         .RightHeader = ""
244:         .LeftFooter = ""
245:         .CenterFooter = ""
246:         .RightFooter = "&A"

```

証跡マクロ君.vbs

```

247 .LeftMargin = Application.InchesToPoints(0.787)
248 .RightMargin = Application.InchesToPoints(0.787)
249 .TopMargin = Application.InchesToPoints(0.393700787401575)
250 .BottomMargin = Application.InchesToPoints(0.984)
251 .HeaderMargin = Application.InchesToPoints(0.512)
252 .FooterMargin = Application.InchesToPoints(0.512)
253
254 .LeftHeader = "&A"
255 .CenterFooter = "&P / &N ページ"
256 .RightFooter = "&F"
257 .LeftMargin = Application.InchesToPoints(0.393700787401575)
258 .RightMargin = Application.InchesToPoints(0.393700787401575)
259 .TopMargin = Application.InchesToPoints(0.984251968503937)
260 .BottomMargin = Application.InchesToPoints(0.393700787401575)
261 .HeaderMargin = Application.InchesToPoints(0.590551181102362)
262 .FooterMargin = Application.InchesToPoints(0.196850393700787)
263
264 .PrintHeadings = False
265 .PrintGridlines = False
266 .PrintComments = xlPrintNoComments
267 .PrintQuality = 600
268
269 .PrintQuality = 1200
270
271 .CenterHorizontally = False
272 .CenterVertically = False
273 .Orientation = xlLandscape
274 .Draft = False
275 .PaperSize = xlPaperA4
276 .FirstPageNumber = xlAutomatic
277 .Order = xlDownThenOver
278 .BlackAndWhite = False
279 .Zoom = False
280 .FitToPagesWide = 1
281 .FitToPagesTall = 1
282 .PrintErrors = xlPrintErrorsDisplayed
283 End With
284
285 book_ebi.Sheets(t_count + 1).Cells(1, 1).Activate
286
287 ' 次のファイル名を取得する。
288 t_filename = ""
289 t_filename = Dir()
290 t_count = t_count + 1
291 Loop
292
293 Application.DisplayAlerts = False
294 book_ebi.Worksheets(1).Delete
295 Application.DisplayAlerts = True
296 book_ebi.Worksheets(1).Activate
297
298 create_sheets = 0
299 End Function
300
301 Sub ヘッドOFF()
302
303 ' ヘッドをOFFにする。
304 With target_book.Sheets(2).PageSetup
305     .LeftHeader = ""
306 End With
307
308 End Sub
309
310 Sub ヘッドON()
311
312 ' ヘッドをONにする。
313 With target_book.Sheets(2).PageSetup
314     .LeftHeader = "&A"
315 End With
316
317 End Sub
318
319
320 Function ガラ作成実行()
321     Dim t_count As Integer
322     Dim t_filename As String
323     Dim t_sheetname As String
324     Dim t_nameOut As String
325     Dim t_valueOut As String
326     Dim t_wide As Double
327     Dim t_wkData As String
328

```

取扱注意

証跡マクロ君.vbs

```

329 Dim シート名 As String
330 Dim 番号位置 As String
331
332 '-----
333 ' 設定値の取得
334 '-----
335 t_nameOut = target_book.Sheets(1).Range("H8")
336 t_valueOut = target_book.Sheets(1).Range("H10")
337 t_wide = target_book.Sheets(1).Range("H12")
338 シート名 = target_book.Sheets(1).Range("H11")
339 番号位置 = target_book.Sheets(1).Range("H18")
340
341 '-----
342 ' ページ設定を切替える(設定が"テキスト"の場合か否か)
343 '-----
344 If t_nameOut = t_outSet Then
345     Call ヘッダOFF
346 Else
347     Call ヘッダON
348 End If
349
350
351 '-----
352 ' 確認項目Noを取得する。
353 '-----
354 t_filename = ""
355 t_filename = Sheets(シート名).Range(番号位置).Value
356
357 ' 対象ファイルが存在しなかった場合
358 If t_filename = "" Then
359     Exit Function
360 End If
361
362
363
364 ' 以下、対象がひとつでも見つかった場合の処理
365 '-----
366 ' 新しいワークブックを開く
367 '-----
368 Excel.Application.SheetsInNewWorkbook = 1
369 Set book_ebi = Workbooks.Add
370
371 '-----
372 ' 新しいシートを作成し、証跡を貼り付けていく。
373 '-----
374 t_count = 1
375 Do While t_filename <> ""
376
377     target_book.Sheets(2).Copy After:=Worksheets(t_count)
378
379
380     '-----
381     ' シート名を変更する
382     '-----
383     t_sheetname = t_filename
384
385     '-----
386     ' ファイル名を左上に記入する。(設定が"テキスト"の場合)
387     '-----
388     If t_nameOut = t_outSet Then
389         ActiveSheet.Range("A1").Value = t_sheetname
390     End If
391
392     '-----
393     ' 確認項目内容を追記する。(設定が"ON"の場合)
394     '-----
395     If t_valueOut = t_ON Then
396         ' 確認項目番号の取得
397         t_wkData = Left(t_sheetname, t_numberLen)
398
399         '*** 確認内容の記入 ***
400         ' 確認項目番号を記入する ⇒ VLOOKUP関数の結果が出る
401         target_book.Sheets(1).Range(t_p_num).Value = t_wkData
402         ' コピー ⇒ VLOOKUP関数の結果をコピー
403         t_wkData = target_book.Sheets(1).Range(t_p_val).Cells.Text
404         ' 値を貼付け
405         ActiveSheet.Range(t_p_out).Value = t_wkData
406         ' 文言を出力
407         ActiveSheet.Range(t_p_out).Offset(-1, 0).Value = t_p_str
408
409
410

```

証跡マクロ君.vbs

```

411 '項目番号の取得
412 t_wkData = Left(t_sheetname, t_numberLen2)
413
414 '**** 試験条件の記入 ****
415 '確認項目番号を記入する ⇒ VLOOKUP関数の結果が出る
416 target_book.Sheets(1).Range(t_p_num2).Value = t_wkData
417 'コピー ⇒ VLOOKUP関数の結果をコピー
418 t_wkData = target_book.Sheets(1).Range(t_p_val2).Cells.Text
419 '値を貼付け
420 ActiveSheet.Range(t_p_out2).Value = t_wkData
421 '文言を出力
422 ActiveSheet.Range(t_p_out2).Offset(-1, 0).Value = t_p_str2
423 End If
424
425
426 '証跡を貼り付ける位置へフォーカスを移動する
427 book_ebi.Sheets(t_count + 1).Cells(2, 2).Activate
428
429
430 '-----
431 'シート名編集処理(追加開始)
432 '-----
433 Dim iStrPos As Integer
434 iStrPos = InStr(1, t_sheetname, "_", vbTextCompare)
435 If iStrPos <> 0 Then
436     t_sheetname = Left(t_sheetname, iStrPos - 1)
437 End If
438 '-----
439 'シート名編集処理(追加終了)
440 '-----
441
442 If Len(t_sheetname) > 31 Then
443     'シート名は最大31bytesなので、31bytesを超える
444     'ファイル名(拡張子抜きで)の場合は自動変更しない。
445     MsgBox "ファイル名が31bytesを超えるため、シート名の変更ができませんでした。" & Chr(13) _
446         & "ファイル名:" & t_filename, vbOKOnly + vbExclamation
447 Else
448
449 '-----
450 'シート名設定下限文字の解除(3文字未満でも処理可能な処理へ変更)
451 '-----
452 book_ebi.Sheets(t_count + 1).Name = UCase(t_sheetname)
453 End If
454
455 '-----
456 'シートの横幅調節
457 '-----
458 book_ebi.Sheets(t_count + 1).Columns("A:CQ").ColumnWidth = t_wide
459
460
461 '次のファイル名を取得する。
462 t_filename = ""
463 t_filename = target_book.Sheets(シート名).Range(番号位置).Offset(t_count, 0).Value
464 t_count = t_count + 1
465
466 Loop
467
468 Application.DisplayAlerts = False
469 book_ebi.Worksheets(1).Delete
470 Application.DisplayAlerts = True
471 book_ebi.Worksheets(1).Activate
472
473 End Function

```