

<http://www.patcatnats.fr/spip.php?article74>



Exécuter suivant extension

- Inform@tic - Access -



Date de mise en ligne : vendredi 17 novembre 2006

Copyright © PatCatNat's - Tous droits réservés

La fonction `OpenFileExtend` permet d'exécuter un fichier en fonction de son extension. Vous n'avez pas besoin de connaître l'applicatif pour lancer l'exécution, il suffit que cette extension soit répertoriée dans Windows. Pour ouvrir un fichier Excel il suffit de lancer le classeur. Cette même fonction permet soit d'exécuter le fichier soit de l'imprimer.

Sommaire

- [Syntaxe](#)
- [Appel de la fonction](#)
- [Fonction](#)



Access 97

Access 97 | Non testé |



Access 2003

Access 2003 | Compatible |

Syntaxe

= `OpenFileExtend` ([,] [, Opération]

Nom de document : tout nom de document répertorié dans windows comprenant l'extension.

Type de fenêtre : indiquer le type de fenêtre dans laquelle sera exécuté le document.

Hide, Normal, Minimized, Maximized ou Restore

Opération : indiquer le type d'opération à réaliser. open ou print (open par défaut). open exécutera le document et print imprimera le document seulement sur l'imprimante par défaut et sans mise en page.

Appel de la fonction

Exécuter suivant extension

```
Private Sub Commande0_Click()  
    Dim Réponse As Variant  
    Réponse = OpenfileExtend("classeur1.xls", Maximized, OpExecute)  
    If Not Réponse = True Then  
        MsgBox Réponse  
    End If  
End Sub
```

La variable Réponse contient l'état de l'exécution du fichier. Si l'exécution a été réalisée correctement, la réponse est True. Si l'exécution n'a pas pu être réalisée correctement, la réponse est False, suivie de la cause de la non exécution.

Fonction

Code à insérer dans un module standard

```
Option Compare Database
Option Explicit Private Declare Function ShellExec Lib "shell32.dll" _
    Alias "ShellExecuteA" _
    (ByVal Hwnd As Long, _
    ByVal lpOperation As String, _
    ByVal lpFile As String, _
    ByVal lpParameters As String, _
    ByVal lpDirectory As String, _
    ByVal nShowCmd As Long) _
    As Long

Public Enum TypeOpen
    Hide = 0
    Normal = 1
    Minimized = 2
    Maximized = 3
    Restore = 9
End Enum

Public Enum OP
    OpExecute = 1
    OpPrint = 2
End Enum

Private Const ERROR_SUCCESS = 32&
Private Const ERROR_NO_ASSOC = 31&
Private Const ERROR_OUT_OF_MEM = 0&
Private Const ERROR_FILE_NOT_FOUND = 2&
Private Const ERROR_PATH_NOT_FOUND = 3&
Private Const ERROR_BAD_FORMAT = 11&

Function OpenFileExtend(FileName As String, _
    Optional Window As TypeOpen = Minimized _
    , Optional Operation As OP = OpExecute _
    ) As Variant

    Dim lRet As Long
    Dim varTaskID As Variant
    Dim stRet As String
    lRet = ShellExec(hWndAccessApp, _
        IIf(Operation = OpPrint, "print", "open"), _
        FileName, vbNullString, vbNullString, _
        Window)
    If lRet > ERROR_SUCCESS Then
        stRet = vbNullString
    End If
End Function
```

```
lRet = -1
Else
  Select Case lRet
    Case ERROR_NO_ASSOC:
      varTaskID = Shell("rundll32.exe shell32.dll,OpenAs_RunDLL " _
        & FileName, 1)
      lRet = (varTaskID <> 0)
    Case ERROR_OUT_OF_MEM:
      stRet = "Erreur: Pas assez de mémoire pour exécuter"
    Case ERROR_FILE_NOT_FOUND:
      stRet = "Erreur : FileName non trouvé"
    Case ERROR_PATH_NOT_FOUND:
      stRet = "Erreur : chemin non trouvé"
    Case ERROR_BAD_FORMAT:
      stRet = "Erreur : Type de FileName inconnu"
    Case Else:
  End Select
End If
OpenFileExtend = lRet & _
  IIf(stRet = "", vbNullString, " " & stRet)
End Function
```

Post-scriptum :

Utilisé dans la Médiathèque.

Toutes Api de Raymond sur [son site](#)