

Les évènements dans la feuille de calcul Excel

par SilkyRoad (silkyroad.developpez.com)

Date de publication : 24.10.2006

Dernière mise à jour :

Ce tutoriel décrit les évènements associés à la feuille de calcul.

Les exemples proposés ont été testés avec Excel2002.

I - Introduction

II - Les événements

II-A - Worksheet_Activate

II-B - Worksheet_BeforeDoubleClick

II-C - Worksheet_BeforeRightClick

II-D - Worksheet_Calculate

II-E - Worksheet_Change

II-F - Worksheet_Deactivate

II-G - Worksheet_FollowHyperlink

II-H - Worksheet_PivotTableUpdate

II-I - Worksheet_SelectionChange

III - Téléchargement

I - Introduction

Les événements permettent l'interaction entre votre programme et l'utilisateur. Il s'agit de procédures qui se déclenchent automatiquement lorsqu'une action prédéfinie survient dans:

- * L'application Excel
- * **Le classeur**
- * La feuille de calcul
- * **Les graphiques**
- * **Les UserForm**
- * Les contrôles

Vous pouvez ainsi intercepter l'activation ou la désactivation d'un onglet, la modification du contenu d'une cellule, le déclenchement d'un lien hypertexte, la mise à jour d'un tableau croisé dynamique...etc...

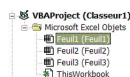
Ce tutoriel présente les événements disponibles au niveau de la feuille de calcul. Chaque onglet possède sa liste d'événements décrits dans le chapitre II.

Utilisez **les procédures événementielles du module objet ThisWorkbook** si vous devez écrire des macros communes à toutes les feuilles du classeur.

Consultez cet autre tutoriel pour gérer **les événements dans les onglets graphiques**.

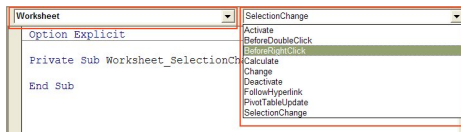
La liste des événements est accessible dans l'éditeur de macros (Alt+F11).

Double cliquez sur le module objet d'une feuille (Feuil1, Feuil2...).



Puis sélectionnez **Worksheet** dans la liste déroulante de gauche.

Vous pouvez maintenant visualiser et choisir un événement dans la liste déroulante de droite.



Utilisez la propriété `EnableEvents` pour désactiver provisoirement les événements. Si vous devez par exemple écrire dans une cellule à partir d'une procédure événementielle qui est déclenchée par la modification des cellules, vous allez provoquer une boucle récursive infinie. La propriété `Application.EnableEvents = False` permet de régler ce problème.

Cet exemple incrémente d'une unité la cellule A1, à chaque modification dans la feuille.

```
Vba

Private Sub Worksheet_Change(ByVal Target As Range)
    'Désactive l'évènement
    Application.EnableEvents = False
    'modification dans la cellule A1
    Range("A1") = Range("A1") + 1
    'Réactive l'évènement
    Application.EnableEvents = True
End Sub
```

Le paramètre CANCEL:

Certains événements disposent de cette option. La valeur par défaut de ce paramètre est égale à `False`. Si vous souhaitez désactiver l'action associée à une macro événementielle, spécifiez `Cancel = True` dans la procédure.

Le paramètre TARGET:

Permet de transmettre la référence des objets cibles (Cellules, Tableaux croisés dynamiques, liens hypertextes) dans la procédure.

II - Les événements

II-A - Worksheet_Activate

`Private Sub Worksheet_Activate()`

Cet événement est déclenché lorsque la feuille est activée (basculement entre les feuilles d'un même classeur).

La procédure n'est pas lancée si vous revenez d'une autre application ou d'un autre classeur Excel.

II-B - Worksheet_BeforeDoubleClick

`Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)`

Identifie le double clic dans une cellule.

Le paramètre `Target` correspond à la cellule qui reçoit le double clic.

Le paramètre `Cancel` désactive l'action associée à une macro événementielle.

Le double clic vous permet d'éditer la cellule (Le curseur clignote dans la cellule). Spécifiez `Cancel = True` pour empêcher l'édition.

Vba

```
Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)
    MsgBox "Vous avez double cliqué sur la cellule " & Target.Address
    Cancel = True
End Sub
```

II-C - Worksheet_BeforeRightClick

`Private Sub Worksheet_BeforeRightClick(ByVal Target As Range, Cancel As Boolean)`

Cet événement survient lors de l'utilisation du clic droit dans une cellule de la feuille.

Le paramètre `Target` correspond à la cellule qui reçoit le clic droit.

Le paramètre `Cancel` désactive l'action associée à une macro événementielle.

Cet procédé permet de désactiver le menu contextuel et d'afficher un menu personnalisé.

Vba

```
Private Sub Worksheet_BeforeRightClick(ByVal Target As Range, Cancel As Boolean)
    Dim i As Byte
    Dim oAnime As Balloon

    Cancel = True

    Assistant.Move 400, 500
    Assistant.On = True

    Set oAnime = Assistant.NewBalloon

    With oAnime
        .Animation = msoAnimationEmptyTrash
        .Button = msoButtonSetOK
        .Heading = "Ceci est un message personnalisé."
        .Text = "Sélectionnez une option dans la liste ci dessous:"

        .CheckBoxes(1).Text = "VB"
        .CheckBoxes(2).Text = "VBA"
        .CheckBoxes(3).Text = "ACCESS"

        .Show

        For i = 1 To 3
            If .CheckBoxes(i).Checked = True Then
                Select Case i
                    Case 1
                        Target = .CheckBoxes(1).Text
                    Case 2
                        Target = .CheckBoxes(2).Text
                    Case 3
                        Target = .CheckBoxes(3).Text
                End Select
            End If
        Next i

    End With

    Assistant.Visible = False
    Assistant.On = False
    Set oAnime = Nothing
End Sub
```

II-D - Worksheet_Calculate

Private Sub Worksheet_Calculate()

Cet événement est déclenché lors du recalcul (validation ou mise à jour des formules) dans la feuille, uniquement si l'option de calcul automatique n'est pas activée.

Menu Outils

Options

Onglet "Calcul"

Option "Sur ordre".

II-E - Worksheet_Change

Private Sub Worksheet_Change(ByVal Target As Range)

Cet événement est déclenché lorsque le contenu d'une cellule est modifiée dans la feuille.

La procédure ne prend pas en compte la modification des mises en forme dans la cellule. Consultez l'exemple proposé dans le chapitre Worksheet_SelectionChange pour identifier la modification d'une mise en forme.

Nota:

Cet événement ne se produit pas si la feuille vient d'être recalculée (Utilisation de la touche F9 en mode calcul sur ordre).

Le paramètre **Target** correspond à la cellule modifiée.

Vba

```
Private Sub Worksheet_Change(ByVal Target As Range)

    'permet de sortir de la procédure si plus d'une cellule est sélectionnée
    '(sinon la suite de la macro renvoie un message d'erreur)
    If Target.Count > 1 Then Exit Sub

    MsgBox "Vous venez de modifier la cellule " & Target.Address & _
        " (" & Target.Value & ")"

End Sub
```

II-F - Worksheet_Deactivate

Private Sub Worksheet_Deactivate()

Cet événement est déclenché lorsque la feuille est désactivée (basculement entre les feuilles d'un même classeur).

La procédure n'est pas lancée si vous activez une autre application ou un autre classeur Excel.

II-G - Worksheet_FollowHyperlink

`Private Sub Worksheet_FollowHyperlink(ByVal Target As Hyperlink)`

L'évènement survient lorsqu'un lien hypertexte est déclenché dans la feuille.

Le paramètre `Target` correspond à l'objet lien hypertexte que vous déclenchez.

Nota:

Il semblerait que l'évènement n'est pas déclenché lorsque le lien est issu de la formule `LIEN_HYPERTEXTE`.

`=LIEN_HYPERTEXTE("[nomClasseur.xls]Feuil2!D13";"La description")`

Cette macro permet d'afficher l'adresse du lien que vous venez de cliquer.

Vba

```
Private Sub Worksheet_FollowHyperlink(ByVal Target As Hyperlink)
    MsgBox Target.Address & vbCrLf & Target.SubAddress
End Sub
```

II-H - Worksheet_PivotTableUpdate

`Private Sub Worksheet_PivotTableUpdate(ByVal Target As PivotTable)`

Cet évènement survient lors de la mise à jour d'un tableau croisé dynamique contenu dans la feuille.

Le paramètre `Target` correspond à l'objet Tableau croisé dynamique mis à jour.

Vba

```
Private Sub Worksheet_PivotTableUpdate(ByVal Target As PivotTable)
    MsgBox "Le TCD '" & Target.Name & "' a été mis à jour."
End Sub
```


II-I - Worksheet_SelectionChange

Private Sub Worksheet_SelectionChange(ByVal Target As Range)

Cet événement survient lors de la sélection d'une cellule dans la feuille.

Le paramètre **Target** correspond à la cellule sélectionnée.

Cet exemple force la saisie dans la cellule A1 tant que la cellule est vide.

Vba

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
    If Range("A1") = "" Then Range("A1").Select
End Sub
```

La macro suivante vérifie que vous avez sélectionné une seule cellule puis indique si la sélection est contenue dans la plage B5:E20.

Vba

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
    Dim Plage As Range

    If Target.Cells.Count > 1 Then
        MsgBox "Sélectionnez une seule cellule."
        Exit Sub
    End If

    Set Plage = Range("B5:E20")

    If Application.Intersect(Target, Plage) Is Nothing Then
        MsgBox "Hors cible."
    Else
        MsgBox "Dans la cible."
    End If
End Sub
```

Cet autre exemple permet d'intercepter la modification de la couleur de fond dans les cellules.

Vba

Vba

```
Option Explicit

Dim x As Integer
Dim Cell As String

Private Sub Worksheet_Activate()
    x = ActiveCell.Interior.ColorIndex
    Cell = ActiveCell.Address
End Sub

Private Sub Worksheet_SelectionChange(ByVal Target As Range)
    On Error Resume Next

    If Cell = "" Then
        x = Target.Interior.ColorIndex
        Cell = Target.Address
        Exit Sub
    End If

    If Range(Cell).Interior.ColorIndex <> x Then _
        MsgBox "la couleur de la cellule " & Cell & " a changé"

    x = Target.Interior.ColorIndex
    Cell = Target.Address
End Sub
```

III - Téléchargement

