



Gestione degli Eventi nel foglio di lavoro

Ci sono molti modi per eseguire una macro, possiamo "lanciarla" avviando la finestra di dialogo Macro, tramite un tasto di scelta rapida, un comando di menu o un pulsante personalizzato, la caratteristica comune di tutti questi metodi è che per eseguire la routine l'utente deve fare qualcosa, scegliere un comando, premere una combinazione di tasti o fare clic su un pulsante. In VBA vi sono tuttavia diverse tecniche che permettono di eseguire delle routine automaticamente quando si verifica un determinato evento, ad esempio l'apertura di una cartella di lavoro o l'apertura di un determinato foglio di lavoro.

Vediamo adesso come si creano routine per eventi specifici associati agli oggetti di Excel, prima di tutto però può essere utile rivedere alcuni concetti relativi alle routine di evento. L'esecuzione di una routine guidata dagli eventi è ciclica, anziché seguire un percorso lineare dall'inizio alla fine, la routine rimane in attesa che si verifichino determinati eventi e quindi rispondere a questi. Un evento è qualcosa che avviene nel programma, per esempio l'apertura di una cartella di lavoro, l'attivazione di un foglio di lavoro, il salvataggio di una cartella di lavoro. Poiché gli eventi specifici che si verificano durante l'esecuzione del programma ne determinano il comportamento, si dice che questo è guidato da tali eventi. Quando si verifica un evento il programma esegue una o più routine correlate a tale evento.

Eventi del foglio di lavoro

Gli oggetti di Excel quali *Workbook*, *Worksheet* contengono un modulo di classe in cui sono memorizzate le routine di gestione degli eventi di tali oggetti. Il modulo di classe di un foglio di lavoro di Excel (*Worksheet*) si raggiunge cliccando sul nome di un foglio, come si vede in *figura 1* indicato dalla *freccia rossa* e selezionando l'oggetto *Worksheet* nell'apposita finestra a discesa indicata dalla *freccia blu*, compare nella finestra del codice la prima routine degli eventi, oltre alla quale, ne sono presenti molti altri selezionabili nella finestra a discesa indicata dalla *freccia verde*. Quando selezionate un evento nell'elenco routine (freccia verde), VBA inserisce una dichiarazione vuota per quella routine. Nella finestra Codice della figura 1 potete vedere la dichiarazione vuota della routine di gestione dell'evento **SelectionChange** del foglio di lavoro

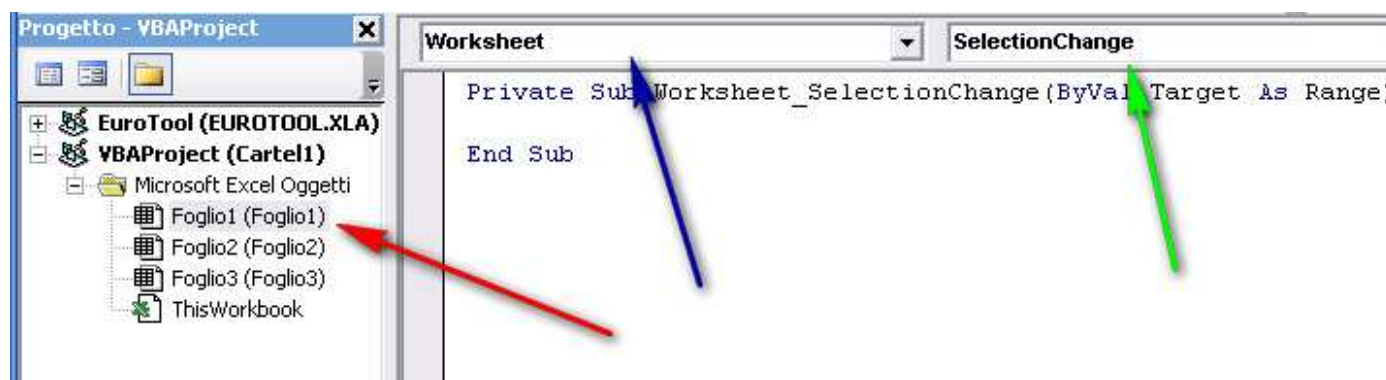


Fig. 1

I vari eventi associati all'oggetto *Worksheet* sono:

Private Sub Worksheet_Activate ()

Questo evento viene innescato quando il foglio viene attivato e se al suo interno è presente del codice VBA, questo viene eseguito. Alcuni esempi di utilizzo di questo evento



```
Private Sub Worksheet_Activate()  
    UserForm1.Show 'mostra la userform1  
End Sub  
  
Private Sub Worksheet_Activate()  
    Range("B1:B20").Sort Key1:=Range("B1"), Order:=xlAscending 'ordinare il range B1:B20  
End Sub  
  
Private Sub Worksheet_Activate()  
    MsgBox "Hai attivato il Foglio1" 'avviso che è stato attivato il foglio1  
End Sub
```

■ **Private Sub Worksheet_SelectionChange** (ByVal Target As Range)

Questo evento si verifica quando si seleziona una cella nel foglio ed il parametro Target corrisponde alla cella selezionata.

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)  
    'selezionare la cella A1 se è vuota  
    If Range("A1") = "" Then Range("A1").Select  
End Sub  
  
Private Sub Worksheet_SelectionChange(ByVal Target As Range)  
    'verificare se la cella selezionata si trova nel Range B5: E20  
    Dim cella1 As Range  
    If Target.Cells.Count > 1 Then  
        MsgBox "Seleziona una sola cella"  
        Exit Sub  
    End If  
    Set cella1 = Range("B5:E20")  
    If Application.Intersect(Target, cella1) Is Nothing Then  
        MsgBox "Fuori Range"  
    Else  
        MsgBox "Nel Range"  
    End If  
End Sub  
  
Option Explicit  
'intercettare il cambiamento del colore di sfondo nelle celle  
Dim x As Integer, Cell As String  
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 "Il Colore della cella " & Cell & " è cambiato"  
    x = Target.Interior.ColorIndex  
    Cell = Target.Address  
End Sub
```



■ **Private Sub Worksheet_Change** (ByVal Target As Range)

Questo evento viene generato quando il contenuto di una cella nel foglio di lavoro viene modificato, la procedura non tiene conto del cambiamento di formato nella cella, inoltre il parametro Target corrisponde alla cella modificata.

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

```
    'avvisa che si sta modificando una cella
```

```
    If Target.Count > 1 Then Exit Sub
```

```
    MsgBox "Vuoi Modificare la cella " & Target.Address & " col valore " & Target.Value
```

```
End Sub
```

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

```
    'converte in maiuscolo le celle modificate nel range A1:A10
```

```
    If Intersect(Target, Range("A1:A10")) Is Nothing Or Target.Cells.Count > 1 Then Exit Sub
```

```
    Application.EnableEvents = False
```

```
    Target.Value = UCase(Target.Value)
```

```
    Application.EnableEvents = True
```

```
End Sub
```

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

```
    'cambia il colore del font della cella se viene modificata
```

```
    Target.Font.ColorIndex = 5
```

```
End Sub
```

■ **Private Sub Worksheet_Deactivate** ()

Questo evento viene attivato quando il foglio è spento (commutazione tra i fogli della stessa cartella di lavoro.) La procedura non viene avviata se si attiva un'altra applicazione o un'altra cartella di lavoro di Excel.

```
Private Sub Worksheet_Deactivate()
```

```
    'proteggere un foglio all'uscita
```

```
    ActiveSheet.Protect DrawingObjects:=True, Contents:=True, Scenarios:=True
```

```
End Sub
```

```
Private Sub Worksheet_Deactivate()
```

```
    'nascondere un foglio
```

```
    Sheets("Foglio3").Select
```

```
    ActiveWindow.SelectedSheets.Visible = False
```

```
End Sub
```

```
Private Sub Worksheet_Deactivate()
```

```
    'avvisare che è stato abbandonato un foglio
```

```
    MsgBox "Hai disattivato " & Name & "." & vbCrLf & "e sei passato nel " & ActiveSheet.Name & "."
```

```
End Sub
```



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

Identifica il doppio click in una cella e il parametro Target corrisponde alla cella che riceve il doppio click, mentre il parametro Cancel disabilita l'azione della macro associata con un evento. Il doppio click consente di modificare la cella, ma se si specifica il valore di Cancel = True la modifica verrà impedita.

```
Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)
    `restituire l'indirizzo della cella che ha ricevuto il doppio click
    MsgBox "Hai fatto Doppio click sulla cella " & Target.Address
    Cancel = True
End Sub
```

```
Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)
    `attivare il foglio 1 facendo doppio click
    If Target.Address = "$A$1" Then Worksheets("Foglio1").Activate
End Sub
```

```
Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)
    `restituisce l'indirizzo di riga e Colonna dove si è fatto il doppio click
    Select Case Target.Column
        Case 1 'colonna A
            MsgBox "Hai fatto doppio click nella Colonna A e nella riga " & Target.Row
        Case 2 'colonna B
            MsgBox " Hai fatto doppio click nella Colonna A e nella riga " & Target.Row
        Case 3 'colonna C
            MsgBox " Hai fatto doppio click nella Colonna A e nella riga " & Target.Row
        Case Else
            MsgBox " Hai fatto doppio click nella Colonna A e nella riga D o altre"
    End Select
End Sub
```

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

Questo evento si verifica quando si utilizza il pulsante destro del mouse su una cella nel foglio, il parametro Target corrisponde alla cella che riceve il tasto destro del mouse, mentre il parametro Cancel disabilita l'azione della macro associata a questo evento.

```
Private Sub Worksheet_BeforeRightClick(ByVal Target As Range,Cancel As Boolean)
    `nasconde le colonne A-B e C cliccando col destro sulla cella D5
    If Target.Address(0, 0) = "D5" Then
        With Columns("A:C").EntireColumn
            .Hidden = Not .Hidden
            Target.Value = IIf(.Hidden, "Colonne", "hide") & " Nascoste"
        End With
    End If
    Cancel = True
End Sub
```

```
Private Sub Worksheet_BeforeRightClick(ByVal Target As Range, Cancel As Boolean)
    `impedisce di copiare il contenuto del range
    MsgBox "Contenuto protetto dalla copia"
    Cancel = True
    If Not Application.Intersect(Target, Range("A1:C20")) Is Nothing Then Exit Sub
End Sub
```



■ **Private Sub Worksheet_Calculate ()**

Questo evento viene attivato quando il foglio di lavoro viene ricalcolato solo se l'opzione di calcolo automatico non è attivata.

```
Private Sub Worksheet_Calculate()  
    `regolare le dimensioni delle colonne da A a F ogni volta che si ricalcola il foglio  
    Columns("A:F").AutoFit  
End Sub  
  
Private Sub Worksheet_Calculate()  
    If IsNumeric(Range("A1")) Then  
        If Range("A1").Value >= 100 Then  
            MsgBox "Il Range A1 ha raggiunto il limite 100", vbInformation  
        End If  
    End If  
End Sub  
Private Sub Worksheet_BeforeRightClick(ByVal Target As Range, Cancel As Boolean)  
    `impedire la comparsa del menu pop-up del destro del mouse  
    Cancel = True  
End Sub
```

■ **Private Sub Worksheet_FollowHyperlink** (ByVal Target As Hyperlink)

L'evento si verifica quando un collegamento viene attivato nel foglio di lavoro, il parametro Target è il collegamento ipertestuale

```
Private Sub Worksheet_FollowHyperlink(ByVal Target As Hyperlink)  
    `visualizzare l'indirizzo del link che hai appena fatto clic.  
    MsgBox Target.Address & vbCrLf & Target.SubAddress  
End Sub  
  
Private Sub Worksheet_FollowHyperlink(ByVal Target As Hyperlink)  
    `mantiene una lista, di tutti i collegamenti che sono stati visitati dal foglio di lavoro attivo.  
    With UserForm1  
        .ListBox1.AddItem Target.Address  
        .Show  
    End With  
End Sub
```

■ **Private Sub Worksheet_PivotTableUpdate** (ByVal Target As pivot)

Questo evento si verifica quando si aggiorna una tabella Pivot contenuta nella scheda, il parametro Target è la tabella pivot.

```
Private Sub Worksheet_PivotTableUpdate(ByVal Target As PivotTable)  
    MsgBox "La tabella '" & Target.Name & "' è stata aggiornata"  
End Sub  
  
Private Sub Worksheet_PivotTableUpdate(ByVal Target As PivotTable)  
    MsgBox "La connessione della tabella Pivot è stata aggiornata"  
End Sub
```