



Esegui macro VBA su un foglio di lavoro protetto

Se si esegue una macro che tenta di apportare delle modifiche in un foglio di lavoro protetto si verifica un errore run-time "Errore di run-time '1004' ". Una possibilità per evitare l'errore è quella di rimuovere la protezione del foglio di lavoro, eseguire il codice, e quindi proteggere nuovamente, come illustrato di seguito:

Foglio1.Unprotect Password: = "123"

Foglio1.Protect Password: = "123"

Tuttavia, questo metodo ha alcuni difetti:

- Se durante l'esecuzione del codice si verifica un errore o viene fermata l'esecuzione della macro, il foglio di lavoro rimarrà senza protezione
- La password di protezione del foglio di lavoro verrà visualizzata nel codice VBA e anche se è possibile proteggere il progetto VBA per impedire l'accesso o la visualizzazione del codice, in rete esistono tantissimi decompilatori che rendono vana la protezione della password
- Sarà necessario inserire il codice (Protect e Unprotect) più volte in ciascuna macro.

Per superare il primo inconveniente in cui il foglio rimane sproteetto sul codice di errore, è possibile utilizzare un *ErrorHandler*, come illustrato di seguito:

```
Sub protezione1 ()  
Foglio1.Unprotect Password: = "123"  
On Error GoTo ErrHandler  
Foglio1.Cells (1, 1) = UCase ("Ciao Mondo")  
Foglio1.Cells (2, 1) = 5/0  
Foglio1.Cells (3, 1) = Application.Max (24, 112, 66, 4)  
Foglio1.Protect Password: = "123"  
ErrHandler: Foglio1.Protect Password: = "123"  
End Sub
```

```
Sub protezione2 ()  
Foglio1.Unprotect Password: = "123"  
On Error Resume Next  
Foglio1.Cells (1, 1) = LCase ("CIAO MONDO")  
Foglio1.Cells (2, 1) = 5/0 .  
Foglio1.Cells (3, 1) = Application.Min (24, 112, 66, 4)  
Foglio1.Protect Password: = "123"  
On Error GoTo 0  
End Sub
```

Dichiarazione On Error

On Error Resume Next: Specifica che quando si verifica un errore di run-time, il controllo passa all'istruzione immediatamente successiva all'istruzione in cui si è verificato l'errore stesso e l'esecuzione continua da quel punto.

On Error GoTo 0: Disattiva l'intercettazione degli errori

On Error GoTo Line: Attiva la routine di gestione degli errori che inizia alla riga specificata. On Error GoTo intercetta tutti gli errori.



Un modo per eseguire le macro in un foglio di lavoro protetto potrebbe essere quella di utilizzare l'argomento *UserInterfaceOnly* nel metodo *Protect*, impostando l'argomento *UserInterfaceOnly* a *True*, nella maniera:

Foglio1.Protect Password: = "123", UserInterfaceOnly: = True.

UserInterfaceOnly è un argomento facoltativo nel metodo *Protect* e il suo valore predefinito è *False*. Impostarlo a *True* significa che la protezione del foglio di lavoro vale solo per l'interfaccia utente e non si applica alle macro e questo permetterà di eseguire tutte le macro nel foglio di lavoro. Se si omette questo argomento, la protezione si applica sia alle macro che all'interfaccia utente.

Si può osservare che se si applica il metodo *Protect* con l'argomento *UserInterfaceOnly* impostata a *True* per un foglio di lavoro e quindi si salva la cartella di lavoro, l'intero foglio di lavoro, non solo l'interfaccia, sarà completamente protetto quando si riapre la cartella di lavoro. Per riattivare la protezione di interfaccia utente dopo che la cartella di lavoro è aperta, è necessario ancora una volta applicare il metodo *Protect* con *UserInterfaceOnly* impostata su *true*.

Per riattivare la protezione di interfaccia utente dopo che la cartella di lavoro è aperta, è possibile utilizzare l'argomento *UserInterfaceOnly* nell'evento *Open* in cui esso viene attivato in tutti i fogli di lavoro o in quelli specificati, ogni volta che la cartella di lavoro è aperta. È inoltre possibile utilizzare l'argomento *UserInterfaceOnly* in un foglio di lavoro, all'inizio della macro, per abilitare la protezione di interfaccia utente ogni volta che la macro viene eseguita. Vedere esempi riportati di seguito:

```
Sub protezione3 ()  
Foglio1.Protect Password: = "123", UserInterfaceOnly: = True  
Foglio1.Cells (1, 1) = UCase ("Ciao Mondo")  
End Sub
```

Se tutti i fogli usano la stessa password, è possibile impostare l'argomento *UserInterfaceOnly* come *TRUE* come segue. Notare che il codice è inserito in *ThisWorkbook* della cartella di lavoro

```
Private Sub Workbook_Open ()  
Dim ws As Worksheet  
For Each ws In ThisWorkbook.Worksheets  
ws.Protect password: = "123", UserInterfaceOnly: = True  
Next ws  
End Sub
```

Per abilitare la protezione a livello di interfaccia utente solo se i fogli di lavoro hanno password differenti in fogli di lavoro specifici, impostare l'argomento *UserInterfaceOnly* come *True* nell'evento *Open*, come segue.

```
Private Sub Workbook_Open ()  
Foglio1.Protect Password: = "123", UserInterfaceOnly: = True  
Foglio2.Protect Password: = "456", UserInterfaceOnly: = True  
End Sub
```

Se si desidera eseguire una macro su un foglio di lavoro protetto, e mantenere il codice visibile e modificabile nell'editor di VB ma non si vuole rivelare o visualizzare la password si può utilizzare il seguente metodo.



Se per esempio si utilizzano tre fogli di lavoro (Foglio1, Foglio2 e Foglio3) in una cartella di lavoro che sono protetti da password, si può inserire un foglio di lavoro supplementare (Foglio4) e renderlo 'molto nascosto' (*VeryHidden*), il che significa che non sarà visibile all'utente e può essere visibile solo con un codice VBA, perché non compare nell'elenco Scopri dell'interfaccia. Un foglio può essere 'molto nascosto' utilizzando una singola linea di codice VBA come la seguente

```
Sub foglio_nascosto()  
Worksheets ("Foglio4"). Visible = xlVeryHidden  
End Sub
```

E' possibile inserire una password, per visualizzare il foglio 'molto nascosto', il che significa che la password dovrà essere ricordato separatamente e non sarà visibile o accessibile finché il foglio è veryhidden. Però è possibile memorizzare le password che proteggono i fogli di lavoro su cui si desidera eseguire le macro, e queste macro richiameranno le password memorizzate nel foglio "VeryHidden"

```
Sub foglio_nascosto()  
Dim pswd As String  
pswd = Cells (1, 1)  
mypass = pswd  
pswdMatch = InputBox ("Inserire la password per visualizzare il foglio")  
If pswdMatch = pswd Then  
Worksheets ("Foglio4"). Visible = True  
Else  
Exit Sub  
End If  
End Sub  
  
Sub Protezione5()  
Dim passfgl2 As String  
passfgl2 = Foglio4.Cells (2, 1)  
Foglio2.Protect Password: = passfgl2 , UserInterfaceOnly: = True  
Foglio2.Cells (1, 1) = UCase ("Ciao Mondo")  
End Sub
```

I 3 esempi esposti poco sopra faranno in modo di eseguire la macro su un foglio di lavoro protetto, la cui password è nascosta nel Foglio4 e lo stesso è stato protetto e reso "molto nascosto". E' possibile abilitare il *Filtro automatico* quando la protezione dell'interfaccia utente è attiva. Questa proprietà di attivare il filtro non viene salvata con il foglio di lavoro. Quando la protezione dell'interfaccia utente è attiva si può utilizzare la proprietà *AutoFilterMode* tramite VBA per utilizzare il filtro automatico si può utilizzare una macro in questo modo

```
Sub attiva_filtro1 ()  
With Foglio1  
. EnableAutoFilter = True  
.Protect Password:="123" , contents:=True, UserInterfaceOnly:=True  
. AutoFilterMode = False  
.Range("A1:B10" ).AutoFilter Field:=1, Criteria1:= "<35"  
End With  
End Sub  
  
Sub attiva_filtro2 ()  
With Foglio1  
.Protect Password:="123" , contents:=True, UserInterfaceOnly:=True  
.Range("A1:B10" ).AutoFilter  
.Range("A1:B10" ).AutoFilter Field:=1, Criteria1:= ">35"  
End With  
End Sub
```



Inoltre è possibile utilizzare `UserInterfaceOnly` per consentire il raggruppamento in un foglio di lavoro protetto quando l'interfaccia utente ha la protezione attiva

```
Sub attiva_gruppo()  
With Foglio1  
    . EnableOutlining = True  
    .Protect Password:="123" , contents:=True, UserInterfaceOnly:=True  
    .Rows ("12:14"). Group  
End With  
End Sub
```

■ Metodo Protezione Worksheet senza `UserInterfaceOnly`

Questo metodo Protegge un foglio di lavoro in modo che non può essere modificato. Se sono necessarie modifiche da apportare a un foglio protetto, è possibile utilizzare il metodo `Protect` di un foglio protetto se viene fornita la password. Inoltre, un altro metodo sarebbe quello di rimuovere la protezione del foglio di lavoro, apportare le modifiche necessarie, e quindi proteggere nuovamente il foglio di lavoro. Vediamo alcuni esempi:

```
Sub consenti_ordinamento1 ()  
Foglio1.Unprotect Password: = "123"  
Foglio1.Range. ("G1: H10") Locked = False  
Foglio1.Protect Password: = "123", contents: = True, AllowSorting: = True  
Foglio1.Range. ("G1: H10") Sort Key1: = Sheet1.Range ("G1"), Order1: = xlDescending  
End Sub  
  
Sub consenti_formattazione1 ()  
Foglio1.Protect Password: = "123", AllowFormattingCells: = True  
Foglio1.Range. ("G1: H10") Font.Bold = True  
Foglio1.Range. ("A1: B10") Font.Color = vbBlue  
End Sub  
  
Sub consenti_formattazione2 ()  
Foglio1.Protect Password: = "123", AllowFormattingColumns: = True,  
AllowFormattingRows: = True  
Foglio1.Columns (2). ColumnWidth = 25  
Foglio1.Rows ("4:5"). RowHeight = 25  
End Sub
```