

Aprire un file xls che è contenuto in un percorso il cui nome è nella cella A1

```
Sub Apri()  
Dim A As String  
A = "C:\Documenti\" & Range("A1").Value & ".xls"  
Workbooks.Open Filename:=A, ReadOnly:=False  
End Sub
```

Apri file xls contenuto in una cartella con richiesta tramite inputbox

```
Sub ApriFileCartella()  
Dim Cart As String  
cartella = InputBox("Scrivi il nome della cartella")  
If cartella = "" Then Exit Sub  
nome = InputBox("Scrivi il nome del file da aprire")  
If nome = "" Then Exit Sub  
Cart= "C:\\" & cartella & "\" & nome & ".xls"  
Workbooks.Open Filename:=Cart, ReadOnly:=False  
End Sub
```

Per visualizzare la calcolatrice sul foglio Excel

```
Sub Calcolatrice()  
Dim CALC  
CALC = Shell("C:\WINDOWS\System32\CALC.EXE", 1)  
End Sub
```

Cancella i campi , che nell'esempio si riferisce ai TEXTBOX , presenti in un Userform

```
Private Sub CommandButton1_Click()  
Dim obj As Control  
For Each obj In Me.Controls  
If TypeOf obj Is MsForms.TextBox Then ' se si trattasse di una combobox , basta chiamare  
l'oggetto  
obj.Text = ""  
End If  
Next  
End Sub
```

Colorare una cella in base al valore in questo caso 100 o superiore a 100 ed inferiore a 200

```
Sub Colora()  
Sheets("Foglio1").Select  
If Range("A1").Value = 100 Then  
Range("A1").Select  
With Selection.Interior  
.ColorIndex = 6 ' colore giallo  
.Pattern = xlSolid  
End With  
End If  
If Range("A1").Value > 100 And Range("A1").Value < 200 Then  
Range("A1").Select  
With Selection.Interior  
.ColorIndex = 3 ' colore rosso  
.Pattern = xlSolid  
End With  
End If
```

```

If Range("A1").Value = "" Then
Range("A1").Select
Selection.Interior.ColorIndex = xlNone ' nessuna colorazione
End If
End Sub

```

Colorare i contenuti in una cella SE si verifica una condizione

```

Private Sub Worksheet_Change(ByVal Target As Range)
If Target = 5 Then 'se il valore immesso nella cella è uguale (esempio: 5), allora
Target.Interior.ColorIndex = 3 'si colora la cella di rosso
End If
End Sub

```

Colorare i caratteri contenuti in una cella se corrispondono al valore di una variabile

```

Private Sub Worksheet_SelectionChange(ByVal Target As Range)
Dim CL As Object
For Each CL In Range("A1:G50") ' Range di celle -A1:G50- che sono oggetto di colorazione dei
caratteri
If CL.Value = "A" Then 'se il valore della cella è uguale ad A
CL.Font.ColorIndex = 3 'mi metti il colore del carattere uguale a rosso
ElseIf CL.Value = "B" Then
CL.Font.ColorIndex = 4
ElseIf CL.Value = "C" Then
CL.Font.ColorIndex = 8
ElseIf CL.Value = "D" Then
CL.Font.ColorIndex = 9
Next
End Sub

```

Combo box in un form, come richiamare date e selezionarle nel foglio

```

Private Sub ComboBox1_Change()
ComboBox1 = Format(ComboBox1, "DD/MM/YY") ' istruzione che ci fa comparire la data
anzichè un numero
End Sub
'<<<<<<<<<
Private Sub CommandButton1_Click() ' esegue la ricerca sul foglio sulla base della DATA
selezionata nella combobox
Dim Cl As Object
Dim zona As Range
Set zona = Sheets("Foglio1").Range(("A1"), Range("A1").End(xlDown)) ' l'intervallo delle date
presenti
For Each Cl In zona
If Cl = CDate(ComboBox1.Text) Then
RIGA = Cl.Row
Cells(RIGA, 2) = "Bravo !" ' qui si può inserire ciò che si vuole o far eseguire un'altra istruzione
End If
Next
Set Cl = Nothing
Set i = Nothing
Set zona = Nothing
End Sub

```

'>>>>>>>>>>>>>>>>>>

```
Private Sub UserForm_Activate()  
Columns("B:B").Clear  
Sheets("Foglio1").Select  
ComboBox1.RowSource = "A1:A30"  
Columns("B:B").Clear ' cancelliamo all'apertura del form i dati precedenti  
End Sub
```

Copia il contenuto ed i formati del Foglio1 nel Foglio2

```
Range("A1:E4").Copy Destination:=Worksheets("Foglio2").Range("A1")
```

Data , ricerca , questa semplice macro, in base alla data inserita nella cella A1 , se corrisponde alla data attuale, colora delle zone del foglio in base ai colori scelti in base ed al tempo scelto .

```
Sub Lampeggia()  
If Range("A1").Value = Date Then ' dove Date è la data attuale  
Dim PauseTime, Start, Finish  
For x = 1 To 5 'inizia il ciclo e lo ripete per 5 volte  
PauseTime = 0.5 ' Imposta la durata in secondi. ho messo 1/2 secondo  
Start = Timer ' Imposta l'ora di inizio.  
Do While Timer < Start + PauseTime  
DoEvents ' Passa il controllo ad altri processi.  
Range("A2:D7").Cells.Interior.ColorIndex = 3 'colora il range di celle di rosso  
Range("A8:D21").Cells.Interior.ColorIndex = 6 'colora il range di celle di giallo  
Loop  
Finish = Timer ' Imposta l'ora di fine della pausa.  
  
PauseTime = 0.5 ' Imposta la durata.  
Start = Timer ' Imposta l'ora di inizio.  
Do While Timer < Start + PauseTime  
DoEvents ' Passa il controllo ad altri processi.  
Range("A2:D7").Cells.Interior.ColorIndex = 6  
Range("A8:D21").Cells.Interior.ColorIndex = 3  
Loop  
Finish = Timer ' Imposta l'ora di fine della pausa.  
Next x  
Range("A2:D7").Cells.Interior.ColorIndex = xlNone 'cancella la colorazione  
Range("A8:D21").Cells.Interior.ColorIndex = xlNone  
End  
End If  
End Sub
```

Data , Cancella, questa macro elimina dal Foglio 1 tutte le RIGHE con date "scadute " rispetto alla data odierna.

```
Sub CancellaData()  
Dim SH As Worksheet
```

```

Dim Rng As Range
Dim rCell As Range
Dim delRng As Range
Dim iLastRow As Long
Dim CalcMode As Long
Const Col As String = "A"
Set SH = Sheets("Foglio1")
iLastRow = SH.Cells(Rows.Count, Col).End(xlUp).Row
Set Rng = SH.Range(Col & "2:" & Col & iLastRow)
On Error GoTo 10
With Application
CalcMode = .Calculation
.Calculation = xlCalculationManual
.ScreenUpdating = False
End With
For Each rCell In Rng.Cells
With rCell
If IsDate(.Value) And .Value < Date Then
If delRng Is Nothing Then
Set delRng = rCell
Else
Set delRng = Union(rCell, delRng)
End If
End If
End With
Next rCell
If Not delRng Is Nothing Then
delRng.EntireRow.Delete
End If
10:
With Application
.Calculation = CalcMode
.ScreenUpdating = True
End With
End Sub

```

Inserisce spazi tra i caratteri di una frase contenuta in una cella

```

Sub InserisciSpazi()
Dim parola As String, carattere As String, ctr As Long, finale As String
parola = Range("A1").Value 'dove anremo a inserire la frase da scomporre
For inter = 1 To Len(parola)
carattere = Mid(parola, inter, 1)
If inter = Len(parola) Then
finale = finale & carattere
Else
finale = finale & carattere & " "
End If
Next
Range("A1").Value = finale 'otteremo come risultato la parola scomposta
End Sub

```

Lampeggia la cella in cui il risultato di una formula supera un valore

Far Lampeggiare un risultato di una formula , quando il valore è superiore a :

1)Nel modulo Thisworkbook, incolla:

```
'=====
Private Sub Workbook_Open()
On Error Resume Next
ActiveWorkbook.Styles.Add Name:="FLASH"
On Error GoTo 0
End Sub
```

2)In un modulo standard, alla testa del modulo, e prima di qualsiasi codice, incolla:

```
Option Explicit
Public blFlash As Boolean
Dim NextTime As Date
Public Sub StartFlash()
NextTime = Now + TimeValue("00:00:01")
With ActiveWorkbook.Styles("Flash").Font
If .ColorIndex = 2 Then .ColorIndex = 3 Else .ColorIndex = 2
End With
Application.OnTime NextTime, "StartFlash"
End Sub
```

```
'-----
Public Sub StopFlash()
On Error Resume Next
Application.OnTime NextTime, "StartFlash", Schedule:=False
ActiveWorkbook.Styles("Flash").Font.ColorIndex = xlAutomatic
End Sub
```

3)Nel Modulo del foglio, incolla:

```
Private Sub Worksheet_Change(ByVal Target As Range)
Dim Rng As Range
Dim rCell As Range
Const myvalue As Variant = 10 ' Il valore di riferimento
Set Rng = Me.Range("B1:B10") ' Il Range di celle da analizzare
If Not Intersect(Rng, Target) Is Nothing Then
Call StopFlash
blFlash = False
For Each rCell In Rng.Cells
With rCell
If .Value > myvalue Then ' se il valore presente nel range ("B1:B10") è superiore a 10 allora.
blFlash = True
.Style = "FLASH"
Else
.Style = "Normal"
End If
End With
Next rCell
If blFlash Then
Call StartFlash
Else
Call StopFlash
End If
```

```
End If
End Sub
```

Lampeggia zona di dati

```
Sub Lampeggia()
If Range("E1").Value = Date Then ' dove Date è la data attuale
Dim PauseTime, Start, Finish
For x = 1 To 5 'inizia il ciclo e lo ripete per 5 volte
PauseTime = 0.5 ' Imposta la durata in secondi. ho messo 1/2 secondo
Start = Timer ' Imposta l'ora di inizio.
Do While Timer < Start + PauseTime
DoEvents ' Passa il controllo ad altri processi.
Range("A1:D7").Cells.Interior.ColorIndex = 3 'colora il range di celle di rosso
Range("A12:D21").Cells.Interior.ColorIndex = 6 'colora il range di celle di giallo
Loop
Finish = Timer ' Imposta l'ora di fine della pausa.
PauseTime = 0.5 ' Imposta la durata.
Start = Timer ' Imposta l'ora di inizio.
Do While Timer < Start + PauseTime
DoEvents ' Passa il controllo ad altri processi.
Range("A1:D7").Cells.Interior.ColorIndex = 6
Range("A12:D21").Cells.Interior.ColorIndex = 3
Loop
Finish = Timer ' Imposta l'ora di fine della pausa.
Next x
'finisce il ciclo, appare un messaggio
'che blocca i colori
MsgBox "ATTENZIONE!!!!"
'premuto ok sul messaggio, vengono eliminati i colori
Range("A1:D7").Cells.Interior.ColorIndex = xlNone
Range("A12:D21").Cells.Interior.ColorIndex = xlNone
'finisce
End
End If
End Sub
```

Macro , eseguire una macro con un tasto della tastiera, nell'esempio F5

```
Public Sub Aziona()
Application.OnKey "{F5}", "NomeMiaMacro"
End Sub
```

Macro a Tempo , realizzata da Ricky53, che permette impostando il tempo , di poter lavorare in un foglio di Excel e salvare ogni periodo di tempo i dati in altro files

```
Public Sub SalvaPeriodicamente()
Dim Nome_Attuale As String
Dim Percorso As String
Dim Nome_File As String
Dim Minuti As Integer
Dim Secondi As Integer
Dim Tempo As String
```

```

Nome_Attuale = ActiveWorkbook.Name
Percorso = "C:\"
NomeFile = Mid(Nome_Attuale, 1, Len(Nome_Attuale) - 4) + "_" +
Format(Date, "yyyymmdd") + "_" + Format(Time, "hhmmss") & ".XLS"
ActiveWorkbook.Save
ActiveWorkbook.SaveCopyAs Percorso & NomeFile
Minuti = 0
Secondi = 15
Tempo = "00:" + Format(Minuti, "0#") + ":" + Format(Secondi, "0#")
Application.OnTime Now + TimeValue(Tempo), "SalvaPeriodicamente"
End Sub

```

Elimina Riga vuota

```

Sub EliminaRigaVuota()
Range("A:A").SpecialCells(xlCellTypeBlanks).EntireRow.Delete
End Sub

```

Elimina Cella vuota

```

Sub EliminaCelleVuote()
Range("A:A").SpecialCells(xlCellTypeBlanks).Delete (xlShiftUp)
End Sub

```

Protezione e Sprotezione dei fogli di un file di Excel

Option Compare Text ' condiziona l'inserimento rispettando le minuscole maiuscole presenti nella password

```

Sub Sproteggi()
Dim ws As Worksheet
Const Valore As String = "Luca" ' da cambiare , inserisci la tua password
pwd = InputBox("INSERISCI PASSWORD", "Inserisci Password")
If pwd = "" Then ' se non 'c'è alcun valore
MsgBox " Il Campo non può essere vuoto ", Title:="Inserisci la Password"
Exit Sub
End If
If pwd <> Valore Then ' se la password digita è diversa da quella indicata
MsgBox "La Password è sbagliata", Title:="Inserisci la Password"
Exit Sub
End If
If pwd = Valore Then
For Each ws In Worksheets
ws.Unprotect Password = Valore
Next ws
MsgBox " Autenticazione riuscita , i fogli sono sproteetti ! ", Title:="Password Corretta "
End If
Set ws = Nothing
End Sub

```

' poi nel

```

Private Sub Workbook_Open()
Dim ws As Worksheet
For Each ws In Worksheets
ws.Protect

```

```
Next ws  
End Sub
```

Ricerca file presente in una cartella o percorso del computer

```
Sub Ricerca()  
Set Cerca = Application.FileSearch ' metodo di ricerca  
With Cerca  
.LookIn = "C:\Documenti" 'cerca nella cartella nell'esempio documenti oppure nel percorso  
desiderato  
.Filename = "Luca.xls" ' il nome del file da cercare : Luca.xls  
If .Execute() > 0 Then 'se eseguendo la ricerca trovi il file (execute sarà 1, quindi 'maggiore di  
zero)  
MsgBox "Il File è presente." 'messaggio di conferma della presenza del files  
Else 'altrimenti avvisi con questo messaggio:  
MsgBox "File non trovato."  
End If 'fine condizione  
End With  
End Sub
```

Ricerca file con tre variabili: cartella, nome, estensione

```
Sub RicercaFile()  
'sotto assegnazione a tre variabili del contenuto delle tre celle usate:  
cartella = Range("A1").Value  
nome = Range("B1").Value  
este = Range("C1").Value  
Set app = Application.FileSearch  
With app  
.NewSearch  
.LookIn = "C:\" & cartella & "" ' corrisponde al testo indicato nella cella A1, ovviamente  
nell'istruzione si cerca nel percorso C:,  
'ma può essere indicato il percorso voluto  
.SearchSubFolders = True  
'sotto usiamo i valori scritti in B1 e C1 per cercare nome file o tutti i file (*) con 'l'estensione  
voluta  
.Filename = "" & nome & "." & este & ""  
'sotto: utilizzo delle costanti di execute per ottenere la ricerca in ordine alfabetico  
If .Execute(SortBy:=msoSortByFileName, _  
SortOrder:=msoSortOrderAscending) > 0 Then  
'primo messaggio che avvisa quanti files sono stati trovati (con .FoundFiles.Count)  
MsgBox "Ci sono " & .FoundFiles.Count & " file(s) trovati."  
'inizio del ciclo For Next per restituire per ogni file trovato, il nome e il suo percorso;  
'assegnazione alla variabile I di detto nome, che cambierà ad ogni ciclo fino alla fine dei 'files  
trovati, e restituiti uno dopo l'altro con un messaggio  
For I = 1 To .FoundFiles.Count  
MsgBox .FoundFiles(I)  
'sotto: inserimento in questo punto, cioè dopo che sarà riportato il nome di un file con la 'msgbox  
sopra, che inseriamo la domanda se vorremo uscire  
chiedi = MsgBox("Vuoi uscire dalla ricerca ?", vbYesNo)  
'se la risposta sarà SI allora:  
If chiedi = vbYes Then  
Exit For
```



```
End If
Next I
Else
MsgBox "File(s) non trovato."
End If
End With
End Sub
```

Riga, blocca con il VBA

```
Sub Blocca()
Rows("2:2").Select ' riferimento alla riga 2 e bloccherà la prima riga
ActiveWindow.FreezePanels = True
End Sub
```

Riga, Sblocca con il VBA

```
Sub Blocca()
Rows("2:2").Select ' riferimento alla riga 2 e sbloccherà la prima riga
ActiveWindow.FreezePanels = False
End Sub
```

Rinominare i fogli in base al nome indicato nella Cella A1

```
Sub RinominaFogli()
Dim CL As Worksheet
For Each CL In Worksheets
CL.Name = CL.Range("A1")
Next
End Sub
```

Salva file , macro che consente di uscire dal file ed eseguire senza messaggi il salvataggio dello stesso, l'istruzione va inserita nel ThisWorkbook

```
Private Sub Workbook_BeforeClose(Cancel As Boolean)
Application.DisplayAlerts = False ' consente di non chiudere col messaggio di salvare o no
ActiveWorkbook.Save ' salva il file
Application.Quit 'esci dall'applicazione
End Sub
```

Textbox, Selezionare un Foglio di lavoro scrivendo il nome del foglio in una TextBox su una UserForm, e usando l'evento Click di un CommandButton.

```
Private Sub CommandButton1_Click()
X = TextBox1.Text
Worksheets("" & X & "").Select
End Sub
```

Textbox, messaggio di errore o meglio controllo dell'inserimento di solo numeri nel campo

```
Private Sub TextBox1_Exit(ByVal Cancel As MSForms.ReturnBoolean)
If Not IsNumeric(TextBox1.Text) Then ' se il campo non contiene dei numeri verrà
visualizzato il messaggio
MsgBox "Inserire solo numeri!", Title:="Errore"
End If
End Sub
```

Attivazione di un form, il testo di una TextBox viene evidenziato automaticamente e si cancella alla pressione di un tasto.

```
Private Sub UserForm_Activate()
With TextBox1
.SelStart = 0
.SelLength = Len(.Text)
.SetFocus
End With
End Sub
```

Userform, pulire i campi in tutte le TEXTBOX presenti, si può applicare anche alle COMBOBOX ed altri controlli del form

```
Private Sub CommandButton1_Click()
Dim obj As Control
For Each obj In Me.Controls
If TypeOf obj Is MsForms.TextBox Then ' per le combobox cambiare MsForms.ComboBox
obj.Text = ""
End If
Next
End Sub
```

Userform, permette di digitare in una Textbox un valore numerico con la Virgola, utilizzando il "." (punto) del tastierino numerico

```
Private Sub TextBox1_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
If KeyAscii = 46 Then '46 è il codice Ascii del punto e 44 della virgola
SendKeys ",", False
KeyAscii = 0
End If
End Sub
```

Userform, caricare una COMBOBOX mediante una variabile

```
Private Sub UserForm_Activate()
x = Sheets("Foglio1").[A1].End(xlDown).Row ' carico la combobox che prenderà tutti valori
presenti da A1 fino all'ultimo valore della colonna
'con l'uso della variabile x
ComboBox1.RowSource = "A1:A" & x & ""
End Sub
```

Word , aprire un file.doc

```
Sub ApriWord()
Dim WordApp As Object
Set WordApp = CreateObject("Word.Application")
WordApp.Documents.Open "C:\Luca.doc" ' o la directory dove è collocato il file di word
WordApp.Visible = True
Set WordApp = Nothing
End Sub
```