

Capitolo 43: Creazione di componenti aggiuntivi di Excel personalizzati

1. Comprendere e creare componenti aggiuntivi (Add-In) in Excel

a. Comprendere i componenti aggiuntivi (Add-Ins)

Un componente aggiuntivo (Add-In) è un file Excel (solitamente .xlam o .xla) che estende le funzionalità standard di Excel. Può contenere macro, funzioni personalizzate (UDF), moduli e moduli di classe. Gli Add-In sono particolarmente utili per riutilizzare codice o automatizzare attività comuni.

b. Lavorare con i componenti aggiuntivi

Per gestire gli Add-In:

- Vai su 'File' > 'Opzioni' > 'Componenti aggiuntivi'
- In basso, seleziona 'Gestisci: Componenti aggiuntivi di Excel' e clicca 'Vai'
- Da qui è possibile caricare, attivare o rimuovere Add-In.

In alternativa:

- Utilizza la scheda 'Sviluppo' > 'Componenti aggiuntivi di Excel'.

c. Quando creare un componente aggiuntivo

Considera la creazione di un Add-In se:

- Hai macro o funzioni da riutilizzare in più file.
- Vuoi distribuire codice senza esporre il contenuto originale.
- Vuoi mantenere l'interfaccia utente più pulita (niente moduli visibili).

d. Creazione di un componente aggiuntivo

1. Crea un file Excel standard e inserisci il codice VBA necessario.
2. Salva il file con estensione .xlam (File > Salva con nome > Excel Add-In).
3. Una volta salvato, attiva l'Add-In come descritto sopra.
4. Le funzioni e le macro saranno disponibili globalmente in Excel, ma l'Add-In non sarà visibile come normale cartella di lavoro.

Esempi pratici

Creare un Add-In con una funzione personalizzata SommaPari

```
Function SommaPari(rng As Range) As Double
    Dim cell As Range
    For Each cell In rng
        If IsNumeric(cell.Value) And cell.Value Mod 2 = 0 Then
            SommaPari = SommaPari + cell.Value
        End If
    Next cell
End Function
```

Creare un Add-In con una macro che ordina i dati

```
Sub OrdinaDati()
    Range("A1").CurrentRegion.Sort Key1:=Range("A1"), Order1:=xlAscending, Header:=xlYes
End Sub
```

Add-In con macro di salvataggio automatico

```
Sub Salvataggio()
    ThisWorkbook.Save
End Sub
```

Add-In che visualizza un promemoria ogni apertura

```
Private Sub Workbook_Open()
    MsgBox "Controlla la scadenza del progetto."
End Sub
```

Funzione personalizzata per convertire euro in dollari

```
Function EuroToUSD(importo As Double) As Double
    EuroToUSD = importo * 1.1 ' Simulazione cambio
End Function
```

Esercizi

Crea un Add-In con una funzione che restituisce il valore massimo di una colonna

```
Function MaxCol(rng As Range) As Double
    MaxCol = Application.WorksheetFunction.Max(rng)
End Function
```

Crea un Add-In che esegue una macro al salvataggio del file

```
Private Sub Workbook_BeforeSave(...)
    ' Codice
End Sub
```

Distribuisci un Add-In che protegge automaticamente un foglio

```
Sub Proteggi()
    ActiveSheet.Protect Password:="1234"
End Sub
```

Crea un Add-In con una UserForm personalizzata di input

Crea UserForm con caselle di testo, pulsanti e gestisci l'input utente

Scrivi un Add-In che aggiunge una nuova voce alla barra multifunzione (con XML personalizzato)

Richiede conoscenze su Ribbon XML e callback in VBA

2. Creare un componente aggiuntivo (Add-In): esempio pratico

a. Esempio di componente aggiuntivo (Add-In)

Un esempio completo di Add-In include moduli, UserForm, macro e interfaccia utente. Questo consente di fornire strumenti riutilizzabili in qualsiasi cartella di lavoro di Excel.

b. Informazioni sul Modulo1

Il modulo standard (es. Module1) contiene funzioni e macro generiche.

Esempio:

```
Sub MostraForm()  
    UserForm1.Show  
End Sub
```

c. Informazioni sulla UserForm

Una UserForm consente di creare interfacce utente personalizzate con pulsanti, caselle di testo, etichette, ecc. Gli eventi associati ai controlli sono scritti direttamente nel modulo della UserForm.

d. Test della cartella di lavoro

Prima di salvare come Add-In, è importante testare tutte le macro, validare le interazioni con la UserForm e controllare che non vi siano errori.

e. Aggiunta di informazioni descrittive

Nel VBA Editor: Strumenti > Proprietà VBAProject > scheda Protezione o Commenti per inserire autore, descrizione, ecc.

f. Creazione dell'interfaccia per la macro dell'Add-In

Puoi usare:

- Barra multifunzione personalizzata con Ribbon XML (richiede conoscenze avanzate)
- Un modulo con Sub da eseguire tramite macro
- Collegamento a pulsanti o comandi da Quick Access Toolbar

g. Protezione del progetto

VBA Editor > Strumenti > Proprietà VBAProject > Protezione > Spunta 'Blocca progetto alla visualizzazione' e imposta una password.

h. Creazione dell'Add-In

1. File > Salva con nome > Tipo file: 'Componente aggiuntivo di Excel (*.xlam)'
2. Assegna un nome significativo e salva.

i. Installazione dell'Add-In

1. File > Opzioni > Componenti aggiuntivi
2. Gestisci: Componenti aggiuntivi di Excel > Vai...
3. Sfoglia e seleziona il file .xlam

Esempi pratici

Mostrare una UserForm tramite macro

```
Sub Mostra()  
    UserForm1.Show  
End Sub
```

Salvare un file da macro dell'Add-In

```
Sub Salva()  
    ActiveWorkbook.SaveAs Filename:="C:\report.xlsx"  
End Sub
```

Usare una funzione personalizzata da Add-In

```
Function Doppio(valore As Double)  
    Doppio = valore * 2  
End Function
```

Mostrare una finestra di conferma prima di procedere

```
If MsgBox("Procedere?", vbYesNo) = vbYes Then  
    Call Azione  
End If
```

Aggiungere il nome dell'autore in Proprietà progetto
Proprietà VBAProject > Commenti: 'Creato da [Nome]'

Esercizi

Crea una UserForm che raccolga un nome e lo scriva in A1

Usa TextBox1.Text e Range("A1").Value

Crea un Add-In che calcoli il 22% di un valore

Function CalcoloIVA(valore As Double): CalcoloIVA = valore * 0.22

Proteggi il tuo progetto VBA con password

VBA Editor > Proprietà progetto > Protezione

Scrivi una macro nell'Add-In che mostri l'orario corrente

MsgBox Format(Now, "hh:mm:ss")

Collega una macro dell'Add-In a un pulsante sulla barra di accesso rapido

Opzioni > Barra di accesso rapido > Aggiungi macro