

Capitolo 40: Creazione di UserForms

1. Perché creare UserForm in VBA

a. Motivazioni per usare le UserForm

Le UserForm in VBA (maschere personalizzate) servono per creare interfacce utente grafiche che semplificano l'interazione con l'utente finale. Anziché utilizzare solo input in celle o MsgBox, le UserForm permettono di progettare veri e propri moduli con controlli come caselle di testo, pulsanti, elenchi a discesa, etichette, checkbox.

Le UserForm sono ideali quando:

- Si desidera raccogliere più dati contemporaneamente
- È necessario controllare la validità degli input
- Si vuole offrire un'interfaccia più intuitiva e professionale
- Si desidera navigare o modificare dati in modo guidato

Le UserForm si creano tramite l'Editor VBA:

- Alt + F11 per accedere
- Inserisci → UserForm
- Usa la toolbox per trascinare controlli

Per mostrare una UserForm:

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

Esempi pratici

Creare una UserForm per inserire nome e cognome

Aggiungi due TextBox e un pulsante OK. Al clic, i dati vengono inseriti in un foglio.

UserForm con un ComboBox per scegliere un dipartimento

La selezione popola una cella o avvia una Sub legata alla scelta.

Maschera con Checkbox per opzioni multiple

Raccogli le scelte dell'utente e compila celle specifiche.

UserForm che mostra dati da un foglio

Utilizza un controllo ListBox per mostrare un elenco da Excel.

Form con pulsante di uscita e validazione input

Usa MsgBox per avvisi se i campi obbligatori sono vuoti.

Esercizi

Crea una UserForm con due TextBox per nome e cognome e un pulsante che li scrive in A1 e B1

Usa il metodo .Value dei controlli e Range().Value per scrivere su foglio.

Aggiungi una ListBox che mostri i nomi delle colonne del foglio attivo

Utilizza un ciclo per popolare la ListBox dinamicamente.

Inserisci una ComboBox con i mesi dell'anno e scrivi la scelta in una cella

Carica i mesi in Initialize e usa .Value al clic del pulsante.

Aggiungi un controllo per chiudere la UserForm

Utilizza il comando Unload Me.

Valida un campo numerico: se non è numero mostra un messaggio d'errore

Utilizza la funzione IsNumeric per controllare l'input.

2. Alternative alle UserForm in VBA

a. Esplorare le alternative alle UserForm

Quando non è necessario creare un'interfaccia complessa, è possibile utilizzare strumenti più semplici come le funzioni `InputBox` e `MsgBox` per interagire rapidamente con l'utente. Queste funzioni sono più veloci da implementare e sufficienti per semplici input o messaggi.

b. Utilizzare la funzione `InputBox`

La funzione `InputBox` (`InputBox`) consente di raccogliere un valore dall'utente tramite una finestra di input semplice.

Sintassi:

```
variabile = InputBox(prompt[, titolo] [,default] [,xPos] [,yPos] [,helpfile, context])
```

Esempio:

```
Dim nome As String  
nome = InputBox("Inserisci il tuo nome:", "Richiesta nome")
```

c. Utilizzare la funzione `MsgBox`

La funzione `MsgBox` (`MsgBox`) mostra un messaggio all'utente e può restituire una risposta (OK, Cancel, Yes, No).

Sintassi:

```
risposta = MsgBox(prompt[, bottoni] [,titolo])
```

Esempio:

```
Dim conferma As Integer  
conferma = MsgBox("Vuoi continuare?", vbYesNo + vbQuestion, "Conferma")  
If conferma = vbYes Then  
MsgBox "Hai scelto di continuare."  
End If
```

Esempi pratici

Chiedere all'utente l'età tramite `InputBox`

```
Dim eta As Integer  
eta = InputBox("Inserisci la tua età")
```

Visualizzare un semplice messaggio con `MsgBox`

```
MsgBox "Operazione completata con successo!"
```

Usare `MsgBox` per scegliere tra Sì e No

```
If MsgBox("Salvare i dati?", vbYesNo + vbQuestion) = vbYes Then  
MsgBox "Dati salvati."  
End If
```

`InputBox` per inserire il nome di un cliente

```
Dim cliente As String  
cliente = InputBox("Nome cliente:", "Gestione clienti")
```

Messaggio di avviso prima di chiudere il file

```
If MsgBox("Sei sicuro di voler uscire?", vbYesNo + vbExclamation) = vbNo Then Exit Sub
```

Esercizi

Scrivi un codice che chiede all'utente il numero di prodotti acquistati

Usa `InputBox` e salva il valore in una variabile.

Crea un messaggio di benvenuto che usa il nome dell'utente

Usa `InputBox` per il nome e `MsgBox` per il saluto.

Visualizza un messaggio se l'utente non inserisce nulla in una `InputBox`

Controlla se il valore è una stringa vuota.

Chiedi conferma per cancellare un intervallo di celle

Usa `MsgBox` con `vbYesNo` e `If` per procedere o annullare.

Combina `InputBox` e `MsgBox` per calcolare e mostrare l'IVA di un prezzo

Chiedi il prezzo con `InputBox` e visualizza l'importo con `MsgBox`.

3. Creazione e Gestione delle UserForm in VBA

a. Panoramica sulla creazione di UserForm

Le UserForm sono interfacce grafiche in VBA che consentono agli utenti di interagire con il codice in modo strutturato. Permettono l'inserimento di dati, la selezione di opzioni e l'automazione delle attività attraverso controlli visivi come pulsanti, caselle di testo e altro.

b. Lavorare con le UserForm

Per creare una UserForm:

- Premi Alt + F11 per aprire l'Editor VBA
- Seleziona Inserisci → UserForm
- Usa la toolbox per aggiungere controlli grafici

Una volta creata, è possibile scrivere codice associato agli eventi dei controlli inseriti.

c. Aggiungere controlli

Dalla toolbox puoi trascinare:

- TextBox (Casella di testo)
- CommandButton (Pulsante)
- ComboBox (Elenco a discesa)
- CheckBox (Casella di spunta)
- Label (Etichetta)

Ogni controllo può essere configurato e personalizzato per l'uso desiderato.

d. Modificare le proprietà di un controllo

Le proprietà si modificano nella finestra delle proprietà (F4).

Esempi:

- Name: nome univoco del controllo
- Caption: testo visualizzato
- Enabled: abilita/disabilita il controllo
- Visible: visibilità del controllo
- Value/Text: valore o testo contenuto

e. Gestire gli eventi

Gli eventi permettono di eseguire codice in risposta a interazioni dell'utente.

Esempi:

- Click su un pulsante (CommandButton1_Click)
- Cambiamento valore in TextBox (TextBox1_Change)
- Selezione in ComboBox (ComboBox1_Change)

Scrivere codice all'interno di queste routine permette di personalizzare il comportamento della UserForm.

f. Visualizzare una UserForm

Per mostrare una UserForm:

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

Puoi anche chiudere la UserForm con Unload Me.

Esempi pratici

Creare una UserForm con due TextBox e un pulsante

Inserisci controlli e scrivi una Sub nel pulsante che copia i valori nelle celle A1 e B1.

Modificare il colore di sfondo di una TextBox

Usa la proprietà BackColor nella finestra delle proprietà.

Mostrare un messaggio al clic su un pulsante

Scrivi MsgBox "Operazione completata" nell'evento Click del pulsante.

Cambiare dinamicamente il contenuto di una Label

Label1.Caption = "Benvenuto " & TextBox1.Text nell'evento Change.

Riempire un ComboBox con i mesi dell'anno all'avvio

ComboBox1.AddItem "Gennaio" ... in UserForm_Initialize.

Esercizi

Crea una UserForm con un campo per nome e un pulsante che lo copia in A1

Usa un controllo TextBox e un CommandButton.

Aggiungi un ComboBox con i giorni della settimana e scrivi il valore selezionato in B1

Popola il ComboBox all'avvio e usa ComboBox1.Value.

Scrivi codice che disabilita il pulsante se il TextBox è vuoto

Controlla TextBox1.Text nella routine Change.

Mostra la UserForm con un pulsante nel foglio Excel

Crea una Sub che esegue NomeForm.Show.

Cambia la visibilità di una Label quando si clicca un pulsante

Label1.Visible = False all'interno del Click del pulsante.

4. Esempio di UserForm in VBA

a. Analisi di un esempio di UserForm

Un UserForm può essere progettato per raccogliere dati da un utente in modo semplice e guidato. In questo esempio, creeremo una UserForm che raccoglie il nome e il cognome dell'utente e li scrive nel foglio di Excel.

b. Creazione della UserForm

1. Premi Alt + F11 per aprire l'Editor VBA
2. Vai su Inserisci → UserForm
3. Trascina due controlli TextBox e un controllo CommandButton
4. Rinomina i controlli come segue:
 - TextBox1 → txtNome
 - TextBox2 → txtCognome
 - CommandButton1 → btnInserisci
5. Modifica il testo (Caption) del pulsante in 'Inserisci'

c. Test della UserForm

Per testare la UserForm:

- Premi F5 nell'Editor VBA mentre la UserForm è selezionata
- Oppure crea una Sub nel modulo per mostrarla:

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

Esegui AvviaForm da Excel o dalla finestra Immediate (Ctrl+G).

d. Creazione di una procedura evento

Una procedura evento è un blocco di codice che viene eseguito in risposta a un'azione dell'utente, come il clic su un pulsante.

Nel nostro esempio, la procedura per il pulsante sarà:

```
Private Sub btnInserisci_Click()  
    Range("A1").Value = txtNome.Value  
    Range("B1").Value = txtCognome.Value  
    Unload Me  
End Sub
```

Questa routine copia i dati immessi dall'utente in celle specifiche del foglio.

Esempi pratici

UserForm con campi per nome, cognome e email

Aggiungi tre TextBox e un pulsante che scrive i dati nelle celle A1, B1, C1.

Validazione campo vuoto prima dell'inserimento

If txtNome.Value = "" Then MsgBox "Inserire il nome." Exit Sub

Visualizzare un messaggio di conferma dopo l'inserimento

MsgBox "Dati inseriti correttamente!"

Inserire una ComboBox con scelta città

Aggiungi una ComboBox e popola con .AddItem in UserForm_Initialize

Disabilitare il pulsante se i campi sono vuoti

Controlla la proprietà .Enabled nel Change dei TextBox.

Esercizi

Crea una UserForm con tre TextBox: nome, cognome, età

Inserisci i dati in tre celle alla pressione di un pulsante.

Aggiungi un pulsante di reset che svuota i campi

Utilizza .Text = "" su ciascun TextBox.

Aggiungi una Label che mostra 'Benvenuto [nome]' al clic

Usa Label1.Caption = "Benvenuto " & txtNome.Text

Scrivi una procedura che mostra la UserForm alla selezione di una cella

Utilizza Worksheet_SelectionChange e UserForm1.Show.

Imposta il focus iniziale sul TextBox nome

Usa txtNome.SetFocus in UserForm_Initialize.

5. Secondo Esempio di UserForm in VBA

a. Analisi di un secondo esempio di UserForm

In questo secondo esempio, creeremo una UserForm più completa con più controlli e logiche di interazione. Lo scopo è raccogliere informazioni da un utente e permettere l'inserimento o la modifica dei dati in un foglio.

b. Creazione della UserForm

1. Premi Alt + F11 per aprire l'Editor VBA
2. Inserisci una nuova UserForm
3. Aggiungi:
 - 3 TextBox (Nome, Cognome, Email)
 - 1 ComboBox (Dipartimento)
 - 2 CommandButton (Salva, Annulla)
4. Rinomina i controlli per chiarezza e imposta le proprietà visive nella finestra delle proprietà (F4).

c. Creazione delle procedure evento

```
Private Sub btnSalva_Click()  
    If txtNome = "" Or txtCognome = "" Then  
        MsgBox "Compila tutti i campi obbligatori.", vbExclamation  
    End If  
    Range("A1") = txtNome  
    Range("B1") = txtCognome  
    Range("C1") = txtEmail  
    Range("D1") = cboDipartimento  
    MsgBox "Dati salvati con successo."  
    Unload Me  
End Sub
```

d. Visualizzazione della UserForm

Per mostrare la UserForm, crea una Sub nel modulo:

```
Sub MostraForm()  
    UserForm2.Show  
End Sub
```

e. Test della UserForm

Testa la UserForm:

- Premi F5 nell'editor
- Oppure lancia la Sub MostraForm dalla finestra Immediate (Ctrl + G).

f. Rendere la macro disponibile da un pulsante nel foglio

1. Vai su 'Sviluppo' → 'Inserisci' → 'Pulsante (controllo modulo)'
2. Disegna il pulsante nel foglio
3. Associa la macro MostraForm

g. Rendere la macro disponibile dalla barra di accesso rapido

1. Clic destro sulla barra multifunzione → 'Personalizza barra di accesso rapido'
2. Seleziona 'Macro' dal menu a discesa
3. Aggiungi la macro MostraForm all'elenco e clicca OK

Esempi pratici

UserForm con campi obbligatori e validazione

Mostra un messaggio se Nome o Cognome sono vuoti e blocca il salvataggio.

Riempire un ComboBox dinamicamente

Usa UserForm_Initialize per aggiungere Dipartimenti come HR, IT, Contabilità.

Confermare salvataggio con MsgBox

Mostra MsgBox "Dati salvati con successo." alla fine della routine Salva.

Creare un pulsante in foglio che apre la UserForm

Inserisci da 'Sviluppo' → 'Pulsante modulo' e collega la Sub.

Aggiungere macro alla barra di accesso rapido

Configura dalla finestra di personalizzazione della barra di Excel.

Esercizi

Crea una UserForm con campi per Nome, Cognome, Email, Età

Scrivi una Sub per copiare i valori nelle celle A1:D1.

Aggiungi un pulsante che pulisce i campi senza chiudere la UserForm

Imposta `TextBox.Text = ""` per ogni campo.

Mostra un messaggio di errore se l'email non contiene '@'

Usa la funzione `InStr` per validare il formato email.

Crea un pulsante nel foglio che apre la UserForm

Usa la funzione `Inserisci` → Pulsante e collega la Sub di apertura.

Aggiungi l'esecuzione della macro alla barra di accesso rapido

Usa la finestra 'Opzioni di Excel' per configurarlo.