



- [Home Page](#) 
- [Informazioni](#) 
- [Aiuto](#) 

Centrare un form sullo schermo

http://www.vbsimple.net/howto/ht_005.htm

- [Centrare un form in fase di progettazione.](#)
- [Centrare un form sullo schermo mediante la pressione di un pulsante.](#)
- [Creazione di una funzione per centrare un form sullo schermo.](#)

Centrare un form in fase di progettazione

Difficoltà: ► 1 / 5

Centrare un form sullo schermo in [fase di progettazione](#), ovvero ancora prima che venga avviato il programma, è un'operazione semplicissima.

Si tratta, infatti di impostare la proprietà *StartPosition* a **1 - CenterOwner** oppure a **2 - CenterScreen**. La differenza tra i due valori riguarda la chiamata del form.

Se un form viene caricato e visualizzato da un altro e la proprietà *StartPosition* del secondo è impostata a **1 - CenterOwner**, esso sarà centrato rispetto al suo form chiamante, e non rispetto allo schermo.

Per posizionare un form al centro dello schermo è necessario impostare la proprietà *StartPosition* del form a **2 - CenterScreen**.

Creiamo un nuovo form di nome **Form1**, impostiamo la proprietà *StartPosition* a 2 e proviamo ad eseguire il progetto. All'avvio del programma il form sarà al centro dello schermo.

Centrare un form sullo schermo mediante la pressione di un pulsante

Difficoltà: ► 1 / 5

Un caso del tutto differente è dato dal bisogno di centrare un form sullo schermo in un momento specifico, non necessariamente all'avvio del form. Un momento particolare può essere il click sopra un pulsante.

A tal scopo inseriamo, sopra un nuovo form di nome **Form2**, un pulsante  di nome **Centra**, così come appare nella figura 1.

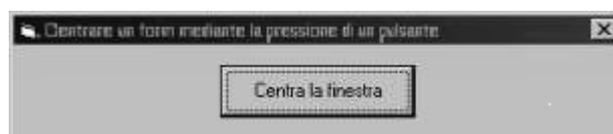


Figura 1

È palese che nel momento in cui l'utente clicca sul pulsante **Centra** la finestra debba spostarsi al centro dello schermo.

La formula generale per centrare qualcosa di una certa dimensione all'interno di uno spazio di dimensione maggiore è: $(\text{AREA} - \text{OGGETTO}) / 2$.

Applicata al nostro progetto: $(\text{Screen.Width} - \text{Me.Width}) / 2$ per quanto riguarda l'asse X e $(\text{Screen.Height} - \text{Me.Height}) / 2$ per l'asse Y.

Apriamo la finestra del codice e nella routine legata all'evento *Click* sul pulsante **Centra** scriviamo:

```
1. Dim X As Single
2. Dim Y As Single
3. X = (Screen.Width - Me.Width) / 2
4. Y = (Screen.Height - Me.Height) / 2
5. Me.Move X, Y
```

Le prime due righe definiscono due variabili di nome X e Y. Esse saranno utilizzate per stabilire le coordinate che dovrà assumere il form.

Alle righe 3 e 4 troviamo l'applicazione della formula vista poco prima.

L'ultima riga effettua il movimento del form alle coordinate X e Y.

Creazione di una funzione per centrare un form sullo schermo

Difficoltà:  2 / 5

Proveremo a generare una funzione generale per centrare qualsiasi form, partendo dal codice che abbiamo visto precedentemente.

Creiamo un nuovo modulo  ed inseriamo all'interno d'esso la nostra funzione **CentraForm**. Scriveremo per cui:

```
1. Public Sub CentraForm(ByVal NomeForm As Form)
2.     Dim X As Single
3.     Dim Y As Single
4.     X = (Screen.Width - NomeForm.Width) / 2
5.     Y = (Screen.Height - NomeForm.Height) / 2
6.     NomeForm.Move X, Y
7. End Sub
```

La funzione non introduce nulla di nuovo. Viene richiesto un parametro di nome **NomeForm** della classe *Form*. Notiamo invece l'uso di *ByVal* per la definizione del parametro.

In realtà Visual Basic, così come il linguaggio C, quando viene passato come parametro un oggetto, non viene effettuata una copia intera dell'oggetto, ad esempio un form. Invece ogni riferimento ad un oggetto caricato in memoria viene ridotto ad un semplice [puntatore](#) ad una locazione di memoria. La copia effettuata dall'utilizzo del *ByVal* non provoca una copia dell'oggetto puntato.

Nel nostro caso **NomeForm** contiene l'indirizzo del form caricato in memoria. La copia del puntatore, tuttavia, punterà alla stessa locazione di memoria.

In conclusione l'utilizzo di *ByVal* invece di *ByRef* non cambia nulla. In entrambi i casi ogni riferimento al parametro passato si riflette sull'oggetto originario.

Vedi a tal proposito le [informazioni aggiuntive sulle funzioni](#).

Proviamo la nostra funzione: inseriamo un nuovo form di nome **Form3** nel nostro progetto. All'interno del form inseriamo un pulsante di nome **CentraButton**, apriamo la finestra del codice e scriviamo all'interno dell'evento Click di CentraButton questa semplice istruzione:

CentraForm Me

La chiamata alla funzione *CentraForm* passerà come parametro un puntatore al form dentro il quale si trova la chiamata.

In ogni caso, si consiglia l'utilizzo della terza soluzione per due fondamentali motivi: la riutilizzabilità del codice e la semplicità d'uso.

[Fibia FBI](#)

28 Novembre 2000



[Torna all'indice degli HowTo](#)
