

GIOCHIAMO CON LE TABELLINE

TESTO DEL PROBLEMA

Creare un programma allegro e vivace per bambini, con il quale esercitarsi sulle tabelline.

Il gioco consiste nel trascinare con il mouse in ogni casella il numero corrispondente all'esatto valore della moltiplicazione, completando in questo modo la tabellina scelta.

Mano a mano che si indovinano le posizioni, il punteggio avanza verso il massimo punteggio, vale a dire "Eccellente".

E' possibile visualizzare un gli errori commessi, le risposte esatte ed il voto ottenuto.

Per sapere come svolgere il gioco, è possibile consultare le istruzioni cliccando l'icona apposita.

L'idea di realizzare questo progetto ci è stata presentata visitando il sito Internet www.vbscuola.it, grazie al quale abbiamo preso spunto dal progetto creato dal signor Enzo Iorio.

MANUALE DELL'UTENTE

LA FORM 1 ...

GIOCHIAMO CON LE TABELLINE



ITC B. Belotti - Bergamo
Anno Scolastico 2006/2007
Autori: Pesenti Erika e Sala Elena
Docenti: Gamba Annalisa e Milone Vincenzo
Classe 4ª

INIZIA A GIOCARE

LA FORM 2 ...



UN AIUTO PER CREARE QUESTO PROGRAMMA

CREAZIONE DELLA MATRICE CONTENENTE I RISULTATI DELLE 10 TABELLINE

- Per poter visualizzare i risultati della tabellina selezionata nelle rispettive label e per verificare se l'utente ha abbinato correttamente il risultato con la rispettiva tabellina, è necessario creare una matrice di 11 righe e di 10 colonne che contenga tutti i risultati delle tabelline dall'1 al 10.
- Questa matrice dovrà essere creata nella Form Load in quanto deve essere eseguita ogni volta che viene avviato il progetto.

```
Option Explicit
Dim TABELLINE(1 To 11, 1 To 10) As Byte

Private Sub Form_Load()
    WindowState = 2
    For CONT = 1 To 10
        CONTA = 0
        Select Case CONT
            Case 1: For X = 0 To 10 Step 1
                    CONTA = CONTA + 1
                    TABELLINE(CONTA, CONT) = X
                Next X
            ...
        End Select
    Next CONT
End Sub
```

- Quando viene cliccato un `CommandButton` con il quale viene scelta la tabellina che si vuole eseguire, compariranno nelle prime 11 label disposte verticalmente il valore della tabellina, mentre nelle label in basso i rispettivi risultati in ordine sparso. Per fare ciò è necessario comportarsi in questo modo:

```
'UNO
Private Sub Command2_Click()
Label1.Caption = "1"
Label2.Caption = "1"

Label168.Caption = TABELLINE(7, 1)
Label169.Caption = TABELLINE(2, 1)

...

VISUALIZZA
SPOSTAMENTO
End Sub
```

- La procedura `VISUALIZZA` consente di cancellare l'eventuale contenuto delle label dei risultati, le label contenenti i possibili risultati e di azzerare le variabili `ERRORI` e `ESATTE`.

CREAZIONE DI UN MODULO E TRASCINAMNETO LABEL

- Il seguente modulo è stato creato dal Signor Enzo Iorio, un esperto informatico, al quale ci siamo rivolte per un aiuto tramite il sito Internet www.vbscuola.it.

```
*****
'Determinare la posizione del mouse
'Per determinare la posizione in cui si trova il mouse
'è necessario usare la seguente funzione API:
'
'
'La funzione GetCursorPos restituisce in una variabile di
'tipo definito dall'utente denominata POINTAPI le coordinate
'x,y relative all'angolo superiore a sinistra dello schermo.
Declare Function GetCursorPos Lib "user32" (lpPoint As POINTAPI) As Long

'Utilizzando la seguente variabile del tipo POINTAPI:

Type POINTAPI
    X As Long
    Y As Long
End Type

'La funzione GetCursorPos restituisce in una variabile di
'tipo definito dall'utente denominata POINTAPI le coordinate
'x,y relative all'angolo superiore a sinistra dello schermo.

'Se necessita invece conoscere la posizione del mouse,
'relativamente alla finestra in uso (form)
'è necessario convertire tali valori usando la seguente dichiarazione API:

Declare Function ScreenToClient Lib "user32" (ByVal hwnd As Long, lpPoint As POINTAPI) As Long
Declare Function GetAsyncKeyState Lib "user32" (ByVal vKey As Long) As Integer

'Con le seguenti costanti:

Const VK_LMOUSEBUTTON = 1
Const VK_RMOUSEBUTTON = 2
Const VK_MMOUSEBUTTON = 4
```

- ☼ La variabile **POINTAPI** viene utilizzata nella procedura `PosizioneWindowMouse`, la quale restituisce le coordinate delle variabili X e Y, espresse in Pixels, in 2 Text Box resi invisibili e consente di trascinare le label in basso sulle label dei risultati.
- ☼ Questa procedura deve essere riscritta tante volte quante le tabelline che si vogliono utilizzare (in questo programma 10), ma con nomi diversi, in quanto si riferiscono ad una label specifica.

```
'La funzione ScreenToClient converte le coordinate x,y, ricavate
'con la funzione GetCursorPos, nelle coordinate relative all'angolo
'superiore sinistro di una determinata finestra specificata.

'CALCOLO COORDINATE X E Y
Sub PosizioneWindowMouse()
    Dim Posizione As POINTAPI

    'le due funzioni, GetCursorPos e ScreenToClient, restituiscono la
    'posizione del mouse con valori espressi in pixel
    GetCursorPos Posizione 'restituisce la posizione x,y relativamente allo schermo
    ScreenToClient Me.hwnd, Posizione 'converte la posizione x,y relativamente alla finestra specificata

    'per ricavare la posizione x e y espressa in Pixels
    PuntoX = Posizione.X 'coordinata del punto x relativa alla finestra (form) corrente
    PuntoY = Posizione.Y 'coordinata del punto y relativa alla finestra (form) corrente

    ' Posizione convertite in Twip usare le seguenti istruzioni:

    PuntoX = PuntoX * Screen.TwipsPerPixelX
    PuntoY = PuntoY * Screen.TwipsPerPixelY

    Text1.Text = "X=" & PuntoX
    Text2.Text = "Y=" & PuntoY

    If Trascina Then
        ' trascina l'oggetto agganciando sempre il puntatore mouse
        ' sempre nel centro dell'oggetto
        Label168.Move (PuntoX - Label168.Width / 2), (PuntoY - Label168.Height / 2)
    End If
End Sub
```

- ☼ Questa procedura deve essere richiamata per tutte le label che devono essere trascinate, utilizzando la proprietà **MouseMove**.

```
'SPOSTAMENTO LABEL
Private Sub Label168_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    ' Il mouse passa sullo stesso oggetto LABEL da trascinare
    ' se si salta questo evento l'oggetto da trascinare va a scatti
    PosizioneWindowMouse
End Sub
```

- Quando viene cliccata un label per essere trascinata viene eseguito questo evento, che utilizza la proprietà **MouseDown**. Attenzione: il seguente evento deve essere riscritto per tutte le label da trascinare.

```
'TRASCINAMENTO EFFETTIVO LABEL
Private Sub Label168_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    ' quando il pulsante del mouse e' premuto allora l'etichetta
    ' deve essere trascinata
    Trascina = True
End Sub
```

- Una volta che la label viene trascinata nel posto in cui si desidera, si rilascia il tasto sinistro del mouse e verranno eseguite le operazioni relative all'evento **MouseUp**.

```
'NON TRASCINAMENTO LABEL
Private Sub Label168_Mouseup(Button As Integer, Shift As Integer, X As Single, Y As Single)
    ' quando il pulsante del mouse non e' premuto allora l'etichetta
    ' non deve essere trascinata
    Trascina = False
Select Case Label1.Caption
    Case "1":
        If PuntoX > 3225 And PuntoX < 3945 And PuntoY > 4695 And PuntoY < 5175 Then
            Label152.Caption = Label168.Caption
            Label168.Visible = False
            ESATTE = ESATTE + 1
            Label166.Caption = ESATTE
        Else
            Label168.Move (1680), (9480)
            ERRORI = ERRORI + 1
            Label161.Caption = ERRORI
        End If
    ...
End Select
End Sub
```

- Per poter eseguire il trascinamento in modo corretto è necessario prestare molta attenzione alle coordinate di ciascuna label, sia quelle contenenti i risultati in ordine sparso, sia di quelle nelle quali mettere i risultati.

RENDERE VISIBILE O INVISIBILE UN CONTROLLO

🌀 Per rendere un controllo (Label, TextBox, List, ...) occorre utilizzare la proprietà **Visibile**.

RENDERE UN CONTROLLO INVISIBILE

```
Label164.Visible = False
```

RENDERE UN CONTROLLO VISIBILE DOPO CHE E' STATO RESO INVISIBILE

```
Label164.Visible = True
```