

### نحوه استفاده از رویدادهای ماوس و صفحه کلید

### هدفهای رفتاری

- پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:
- ۱- رویدادهای ماوس و صفحه کلید را بیان کند.
  - ۲- بتواند از رویدادهای ماوس و صفحه کلید استفاده کند.
  - ۳- کنترل‌های شکل (Shape) و خط (Line) را شرح داده، خصوصیات آن‌ها را توضیح دهد.
  - ۴- بتواند از کنترل‌های شکل (Shape) و خط (Line) استفاده کند.

## کلیات

یکی از مسائلی که در برنامه‌نویسی برای سیستم‌های عامل با رابط گرافیکی مانند ویندوز مورد توجه قرار می‌گیرد، عکس‌العمل‌های مناسب در رابطه با اتفاقاتی مانند فشردن دکمه‌های ماوس، حرکت اشاره‌گر آن یا فشردن کلیدها در صفحه کلید است. در زبان ویژوال بیسیک رویدادهای مناسب در این زمینه برای برآورده کردن نیازهای برنامه‌نویسان در نظر گرفته شده‌اند.

### ۹-۱ رویدادهای ماوس

رویدادهای ماوس در ویژوال بیسیک به چهار گروه تقسیم می‌شوند که عبارتند از: Click، DbClick، MouseDown، MouseUp، MouseMove و DragDrop. در این واحد کار رویدادهایی را که در این رابطه در ویژوال بیسیک وجود دارند، فرا می‌گیرید.

#### ۹-۱-۱ رویداد MouseDown

شکل کلی این رویه رویداد به صورت زیر است و زمانی اجرا می‌شود که یکی از دکمه‌های ماوس به پایین فشرده شود.

`Private Sub ButtonAs Integer, Shift As Integer, X As Single, Y As Single`

همان‌طور که مشاهده می‌کنید این رویداد برخلاف رویدادهایی که تاکنون دیده‌اید دکمه‌ای را که کاربر فشرده است، Button دارای چهار آرگومان است. آرگومان اول یعنی مشخص می‌کند. این مقدار می‌تواند یکی از مقادیر موجود در جدول ۹-۱ باشد به این شکل می‌توانید هر دکمه‌ای از ماوس را که فشرده می‌شود، شناسایی کنید.

جدول ۹-۱

ثابت رشته‌ای	ثابت عددی	توضیح
vbLeftButton	۱	در صورتی که دکمه سمت چپ ماوس فشرده شود.
vbRightButton	۲	در صورتی که دکمه سمت راست ماوس فشرده شود.
vbMiddleButton	۴	در صورتی که دکمه وسط ماوس فشرده شود.
Right Left	۳	هر دو

آرگومان Shift معین می‌کند که در هنگام فشردن دکمه‌های ماوس کدام یک از کلیدهای Alt, Ctrl, Shift یا ترکیبی از آنها فشرده شده است. مقادیر مربوط به آرگومان Shift در جدول ۹-۲ ارایه شده‌اند.

جدول ۹-۲

توضیح	ثابت عددی	ثابت رشته‌ای
در صورتی که کلید Shift فشرده شود.	۱	vbShiftMask
در صورتی که کلید Ctrl فشرده شود.	۲	vbCtrlMask
در صورتی که کلید Alt فشرده شود.	۴	vbAltMask
در صورتی که کلیدهای Shift+Ctrl فشرده شوند.	۳	vbShiftMask+vbCtrlMask
در صورتی که کلیدهای Shift+Alt فشرده شوند.	۵	vbShiftMask+vbAltMask
در صورتی که کلیدهای Ctrl+Alt فشرده شوند.	۶	vbCtrlMask+vbAltMask
در صورتی که کلیدهای Shift+Ctrl+Alt فشرده شوند.	۷	vbShiftMask+vbCtrlMask+vbAltMask

آرگومان‌های X و Y از نوع عدد اعشاری بوده و موقعیت اشاره‌گر ماوس را در زمان فشرده شدن کلیدهای آن معین می‌کند. مقدار X فاصله از سمت چپ و Y فاصله از بالای فرم یا کنترل مربوطه می‌باشند. نام شیء نیز می‌تواند نام یک کنترل یا عبارت Form برای فرم‌ها باشد.

### ۹-۱-۲ رویداد MouseUp

شکل کلی این رویه رویداد به صورت زیر است و زمانی اجرا می‌شود که یکی از دکمه‌های ماوس که به پایین فشرده شده است، رها شود.

`Private Sub نام شیء_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)`

آرگومان‌های این رویداد دقیقاً مشابه رویداد MouseDown است.

### ۹-۱-۳ رویداد MouseMove

شکل کلی این رویه رویداد به صورت زیر است و زمانی اجرا می‌شود که اشاره‌گر ماوس روی فرم یا کنترل مربوطه حرکت کند.

`Private Sub نام شیء_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)`

آرگومان‌های این رویداد نیز دقیقاً مشابه دو رویداد قبلی می‌باشند.



شکل ۹-۱



**مثال ۱:** پروژه‌ای طراحی کنید که

تصویری مطابق شکل ۹-۱ روی یک فرم نمایش دهد و اگر کاربر روی تصویر کلیک کند تصویر با حالت خوشحال نشان داده شود و در صورتی که کاربر کلیک راست کند تصویر با حالت ناراحت نشان داده شود.

۱ - یک پروژه از نوع Standard EXE ایجاد کرده و یک فرم با نام frmface و یک کنترل تصویر با نام imgface مطابق شکل ۹-۱ ایجاد کنید.

۲ - در کنترل تصویر با استفاده از کادرمحاوره Load Picture تصویر FACE10 را از مسیر

D:\Program Files\Microsoft Visual studio\Common\Graphics\Icons\Misc

انتخاب کنید. (با فرض این که برنامه ویژوال استودیو در درایو D: نصب شده باشد).

۳ - رویداد Mouse Down کنترل imgface را به صورت زیر تنظیم کنید.

```
Private Sub imgface_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
If Button = vbLeftButton Then
```

```
imgface.Picture = LoadPicture _
```

```
("C:\Program Files\Microsoft Visual Studio\Common\Graphics\Icons\Misc\FACE02.ICO")
```

```
If Button = vbRightButton Then
```

```
imgface.Picture = LoadPicture _
```

```
("C:\Program Files\Microsoft Visual Studio\Common\Graphics\Icons\Misc\FACE04.ICO")
```

```
End Sub
```

در این رویداد با استفاده از دستور If مقدار آرگومان Button بررسی می‌شود اگر این مقدار برابر با

Vb Left Button باشد به این معنی است که دکمه سمت چپ ماوس فشرده شده است

بنابراین تابع Load Picture تصویر صورت را با حالت خنده نمایش می‌دهد. به همین

ترتیب اگر دکمه سمت راست ماوس فشرده شود نتیجه بررسی شرط موجود در دستور If دوم درست (True) می‌شود. بنابراین تصویر صورت با حالت ناراحت نمایش داده می‌شود.

- ۴ - پروژه و فرم را با نام Show Face ذخیره نمایید سپس پروژه را اجرا کنید و روی تصویر یک بار کلیک و بار دیگر کلیک راست کنید و نتیجه را بررسی نمایید.
- ۵ - اجرای برنامه را متوقف کرده و به پنجره ویژوال بیسیک بازگردید.

### تمرین:



پروژه ShowFace را به شکلی تنظیم کنید تا پس از رها کردن دکمه سمت چپ یا راست، تصویر اول دوباره نمایش داده شود.

## ۴-۱-۹ رویداد DragDrop

شکل کلی این رویه رویداد به صورت زیر است و زمانی اجرا می‌شود که یک کنترل در روی فرم با استفاده از اشاره‌گر ماوس از یک محل به محل دیگر منتقل ((Drag & Drop) شود.

### Private Sub نام شیء \_ (Source As Control, X As Single, Y As Single) DragDrop

آرگومان Source شیء که عملیات انتقال (Drag & Drop) روی آن انجام شده است، تعیین می‌کند. اگر شیء یک کنترل باشد نام کنترل و اگر فرم باشد کلمه Form مورد استفاده قرار می‌گیرد. در رابطه با این رویداد باید توجه داشته باشید که رویداد DragDrop یک کنترل یا فرم زمانی اجرا می‌شود که کنترل دیگری پس از Drag شدن روی آن Drop شود. آرگومان‌های X و Y موقعیت نقطه‌ای را که عمل رها کردن (Drop) در آن‌جا صورت گرفته است، مشخص می‌کنند. اما لازم است بدانید که برای انجام عمل انتقال یک کنترل در روی فرم باید خصوصیت DragMode را برای کنترل موردنظر روی مقدار 1-Automatic تنظیم کنید و به علاوه دستور Source.Move x,y را در رویداد DragDrop فرم قرار دهید.



**مثال ۲:** پروژه Calculator را که در واحد کار پنجم طراحی کرده‌اید به شکلی تغییر

دهید تا بتوان با عمل Drag & Drop محتویات کادرمتن txtnum1 را در کادرمتن txtnum2 کپی کرد. برای این کار عملیات زیر را به ترتیب انجام دهید:

۱- پروژه Calculator را باز کنید.

۲- خصوصیت Drag Mode کنترل کادرمتن  $\text{txtnum}_1$  را روی مقدار ۱-Automatic تنظیم

کنید تا امکان انجام عمل Drag & Drop روی کنترل کادرمتن امکان پذیر باشد.

۳- رویداد Drag Drop کنترل کادرمتن  $\text{txtnum}_2$  را به این صورت تنظیم کنید.

```
Private Sub txtnum2_DragDrop(Source As Control, X As Single, Y As Single)
```

```
txtnum2.Text = txtnum1.Text
```

```
End Sub
```

وقتی کنترل  $\text{txtnum}_1$  در زمان اجرای برنامه با عمل Drag & Drop روی کنترل  $\text{txtnum}_2$

قرار بگیرد این رویداد اجرا می شود و محتویات کادرمتن  $\text{txtnum}_1$  را روی  $\text{txtnum}_2$  ذخیره می کند.

۴- تغییرات را ذخیره کنید و برنامه را اجرا نمایید.

۵- یک عدد در کادرمتن اول تایپ کنید سپس با عمل Drag & Drop کادرمتن اول را

روی کادرمتن دوم بیندازید و نتیجه را بررسی کنید.

۶- اجرای برنامه را خاتمه داده و به پنجره ویژوال بیسیک بازگردید.

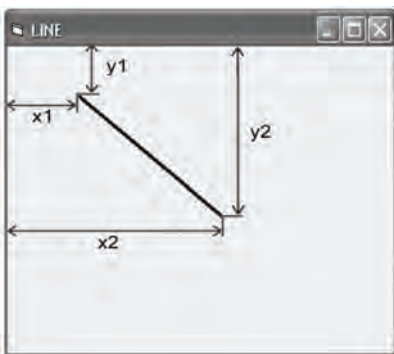
تمرین:



پروژه Calculator را به شکلی تنظیم کنید تا بتوان با عمل Drag & Drop محتویات

کادرمتن دوم را نیز در کادرمتن دوم کپی نمود.

## ۲-۹ کنترل خط (Line)



شکل ۲-۹

با استفاده از این کنترل می توان انواع خطوط

افقی و عمودی و مورب را رسم نمود. در این

کنترل خصوصیت های  $x_1$  و  $y_1$  مختصات نقطه

شروع برای ترسیم خط و خصوصیت های  $x_2$  و  $y_2$

مختصات نقطه انتهایی خط را مشخص می کنند

(شکل ۲-۹). به علاوه با استفاده از خصوصیت

BorderColor می توان رنگ خط و با خصوصیت


BorderWidth ضخامت خط را تعیین نمود.





شکل ۳-۹

مقادیری که خصوصیت `Border` -  
`Style` می تواند کسب کند مشابه مقادیر  
همین خصوصیت در کنترل `Line` است،  
خصوصیت `Shape` نیز مقادیر مختلفی را  
کسب می کند که در شکل ۳-۹ نمایش  
داده شده اند.

 **نکته** واحد اندازه گیری پیش فرض در ترسیم اشکال هندسی در روی فرم، `Twip` است. اگر  
بخواهید این واحد اندازه گیری را تغییر دهید، می توانید از خصوصیت `ScaleMode` فرم استفاده کنید.  
این خصوصیت واحدهای اندازه گیری مختلفی مانند `Pixel`، `Twip`، `Character`، `Inch`، `Millimeter` و  
`Centimeter` را در اختیار شما قرار می دهد.

## ۹-۴ رویدادهای صفحه کلید

در ویژوال بیسیک سه رویداد برای کلیدهای صفحه کلید در نظر گرفته شده است این  
رویدادها شامل رویدادهای `KeyDown`، `KeyUp` و `KeyPress` هستند.

### ۹-۴-۱ رویداد `KeyDown`

شکل کلی این رویه رویداد به صورت زیر است و زمانی اجرا می شود که یکی از  
کلیدهای صفحه کلید فشرده شوند.

**Private Sub** `KeyDown(KeyCode As Integer, Shift As Integer)` **نام شیء**

این رویداد دو آرگومان دارد. آرگومان `KeyCode` کداسکی کلید فشرده شده را معین  
می کند. در صفحه کلید به هر کلید یک عدد صحیح مثبت بین صفر تا ۲۵۵ نسبت داده  
شده است که با عنوان کداسکی (ASCII) نام گذاری شده اند. آرگومان `Shift` مقداری را با  
توجه به فشرده شدن کلیدهای `Ctrl`، `Alt` و `Shift` یا ترکیبی از آن ها را که به طور هم زمان با




کلید اصلی فشرده شده‌اند در اختیار شما قرار می‌دهد. مقادیر مربوط به این آرگومان‌ها به صورت ضمیمه در انتهای کتاب ارایه شده‌اند. برای کنترل‌ها از نام کنترل و برای فرم‌ها کلمه Form به‌عنوان نام شیء استفاده می‌شود.

## ۹-۴-۲ رویداد KeyUp

شکل کلی این رویداد نیز به صورت زیر است و زمانی اجرا می‌شود که کلید فشرده شده در صفحه کلید رها می‌شود.

**Private Sub KeyUp(KeyCode As Integer, Shift As Integer) نام شیء**

آرگومان‌های این رویداد کاملاً مشابه رویداد KeyDown هستند و نام شیء می‌تواند نام یک کنترل یا عبارت Form برای فرم‌ها باشد.

 **نکته** برای تشخیص کاراکترهایی که با استفاده از کلید Shift تایپ می‌شوند (مانند @) و غیره) از آرگومان KeyCode و Shift به‌طور هم‌زمان استفاده کنید.

## ۹-۴-۳ رویداد KeyPress

شکل کلی این رویه رویداد به صورت زیر است و زمانی اجرا می‌شود که یکی از کلیدهای صفحه کلید فشرده شده و رها شود:

**Private Sub KeyPress(KeyAscii As Integer) نام شیء**

این رویه یک آرگومان از نوع عدد صحیح دارد که کداسکی کلید فشرده شده را مشخص می‌کند. مقادیر مربوط به این آرگومان به صورت ضمیمه در انتهای کتاب ارایه شده‌اند. نام شیء می‌تواند نام یک کنترل یا عبارت Form برای فرم‌ها باشد.

تفاوتی که بین این رویداد با رویدادهای دیگر صفحه کلید وجود دارد این است که رویداد KeyPress برای حروف بزرگ و کوچک حرفی کدهای متفاوتی ارایه می‌کند، در صورتی که رویدادهای KeyUp و KeyDown کدهای یکسانی را بازگشت می‌دهند. به عنوان مثال رویداد KeyPress برای کاراکتر A کد ۶۵ و برای کاراکتر a کد ۹۷ تولید می‌کند، اما رویداد KeyDown برای هر دو کاراکتر کد ۶۵ را تولید می‌کند. به علاوه رویداد KeyPress برای سایر کاراکترها مانند علائم <، >، ؟ و غیره کدهایی را مطابق جدول ضمیمه در انتهای کتاب تولید می‌کند.



- **نکته** رویداد KeyDown نسبت به KeyPress اولویت دارد و زودتر اجرا می‌شود.
- در رویداد KeyPress کد مربوط به کلیدهای رقمی در بخش عددی صفحه کلید مطابق با کد کلیدهای رقمی در بخش حرفی صفحه کلید است.
- رویداد KeyPress ترکیب کلید Ctrl با کاراکترهای حرفی و علائم [ و ] را پشتیبانی می‌کند.
- رویداد KeyPress از کلیدهای تابعی، حرکت مکان‌نما و نظیر آن‌ها پشتیبانی نمی‌کند.
- رویداد KeyPress از کلیدهای Enter، Backspace و ESC پشتیبانی می‌کند.

## ۴-۹- خصوصیت KeyPreview

این خصوصیت یکی از خصوصیات مهم در فرم‌هاست و نحوه اجرای رویه‌های رویداد صفحه کلید را برای فرم و کنترل‌های موجود در روی آن معین می‌کند. این خصوصیت از نوع منطقی می‌باشد و مقدار پیش فرض آن مقدار False است. در این حالت اگر کلیدی در صفحه کلید فشرده شود، رویدادهای صفحه کلید مربوط به کنترلی که فوکوس را در اختیار دارد، اجرا می‌شوند. اما اگر مقدار این خصوصیت روی True تنظیم شده باشد ابتدا رویدادهای صفحه کلید فرم اجرا می‌شوند و سپس رویدادهای صفحه کلید مربوط به کنترلی که فوکوس را در اختیار دارد، اجرا می‌شوند. در ضمن اگر هیچ کنترلی وجود نداشته باشد رویدادهای صفحه کلید فرم اجرا می‌شوند.



- **نکته** وقتی فوکوس روی کنترل دکمه فرمان قرار می‌گیرد، کلیدهای Enter و حرکت مکان‌نما نمی‌توانند رویدادهای صفحه کلید فرم و کنترل دکمه فرمان را اجرا نمایند.



**مثال ۳:** یک بازی طراحی کنید که بازیکن، یک توپ متحرک را با استفاده از کلیدهای جهت‌دار در داخل یک قاب کنترل کند و از برخورد آن با دیوارهای این قاب جلوگیری نماید، در صورتی که توپ با دیوارها برخورد کند بازی خاتمه یافته و براساس مدت زمانی که بازیکن توانسته است توپ را کنترل کند، امتیاز وی محاسبه می‌شود. به علاوه کاربر بتواند با فشردن کلید ترکیبی Ctrl+Q هر زمان که مایل باشد از بازی خارج شود و با استفاده از کلید ترکیبی Alt+P حرکت توپ را به طور موقت متوقف کرده و با استفاده از کلید ترکیبی Shift+S توپ را مجدداً به حرکت در آورد.



شکل ۴-۹

۱- برنامه ویژوال بیسیک را اجرا کنید و یک پروژه از نوع Standard EXE به همراه یک فرم با عرض و ارتفاع، ۴۷۸۵ و ۳۷۵۰، مطابق شکل ۴-۹ ایجاد کنید.

همان‌طور که مشاهده می‌کنید یک توپ با کنترل شکل (Shape) و یک کنترل زمان‌سنج برای ایجاد حرکت در توپ استفاده شده است، خصوصیات آن‌ها را مطابق جدول ۵-۹ تنظیم کنید.

جدول ۵-۹

کنترل / خصوصیت	Timer	Shape
Name	tmrtimer	shpball
BackColor	_____	سیاه
BackStyle	_____	1-Opaque
Enabled	False	_____
Interval	۲۰	_____

۲- در بخش تعاریف، رویداد KeyDown و Load فرم را به صورت زیر تنظیم کنید:

Dim intdirection As Byte

Private Sub Form\_KeyDown(KeyCode As Integer, Shift As Integer)

Select Case KeyCode

Case 13: tmrtimer.Enabled = True

Case 37: intdirection = 1

Case 38: intdirection = 2

Case 39: intdirection = 3

Case 40: intdirection = 4

End Select

Select Case Shift

Case vbCtrlMask:

If KeyCode=81 Then Unload Me

Case vbAltMask:

If KeyCode=80 Then tmrTimer.Enabled=False

Case vbShiftMask:

If KeyCode=83 Then tmrTimer.Enabled=True

End Select

End Sub

Private Sub Form\_Load()

intdirection = 1

End Sub

همان‌طور که در رویداد KeyDown فرم مشاهده می‌کنید با استفاده از یک دستور Select Case کلید فشرده شده بررسی می‌شود. اگر مقدار آرگومان KeyCode برابر ۱۳ باشد به معنی فشرده شدن کلید Enter خواهد بود، در این صورت خصوصیت Enabled کنترل زمان‌سنج True می‌شود تا کنترل زمان‌سنج شروع به کار کند. به این ترتیب فشردن کلیدهای جهت‌دار، کدهای ۳۷ تا ۴۰ را تولید می‌کنند که متناسب با کلید فشرده شده متغیر عمومی intdirection مقداردهی می‌شود. از این متغیر برای ایجاد حرکت متناسب با کلید فشرده شده در رویه رویداد کنترل زمان‌سنج استفاده می‌شود. این متغیر در رویه رویداد Load فرم مقدار ۱ را به دست می‌آورد. اگر مقدار این متغیر ۱ باشد به معنی حرکت به چپ، مقدار ۲ به معنی حرکت به بالا، مقدار ۳ به معنی حرکت به راست و مقدار ۴ به معنی حرکت به پایین خواهد بود.

سپس برای مدیریت کلیدهای ترکیبی Ctrl+Q، Alt+P، و Shift+S از دستور Select استفاده می‌شود و با بررسی مقدار آرگومان Sift ابتدا فشرده شدن کلیدهای Ctrl، Alt و Shift بررسی می‌شود. برای این کار در هر Case از ثابت‌های مربوط به هر یک از این کلیدها استفاده شده است و برای تشخیص کلیدهای Q، P، و S همراه کلیدهای Ctrl، Alt و Shift در Case هر یک از آن‌ها یک دستور If قرار داده شده است تا با استفاده از آرگومان KeyCode کلید حرفی نیز شناسایی شود و در صورت درست بودن نتیجه بررسی دستور موردنظر اجرا شود.

۳ - در این مرحله رویداد Timer کنترل زمان‌سنج را به صورت زیر تنظیم کنید:

```
Private Sub tmrtimer_Timer()
```

```
    Select Case intdirection
```

```
        Case 1: shpball.Left = shpball.Left - 50
```

```
        Case 2: shpball.Top = shpball.Top - 50
```

```
        Case 3: shpball.Left = shpball.Left + 50
```

```
        Case 4: shpball.Top = shpball.Top + 50
```

```
    End Select
```

```
    If shpball.Left <= 18 Or shpball.Left >= 4319 Or _
```

```
        shpball.Top <= 22 Or shpball.Top >= 2724 Then
```

```
        tmrtimer.Enabled = False
```

```
        MsgBox "Game Over !!!!", vbCritical, "THE END"
```

```
    End If
```

```
End Sub
```

می‌بینید که با استفاده از یک دستور Select Case مقدار متغیر intdirection که در رویداد KeyDown با توجه به کلید فشرده شده مقداردهی شده است، کنترل می‌شود و با توجه به این مقدار حرکت مناسب برای توپ در نظر گرفته می‌شود. برای مثال اگر مقدار متغیر intdirection برابر ۱ باشد به این معنی است که کلید جهت‌دار چپ فشرده شده است، در نتیجه از خصوصیت Left کنترل شکل ۵۰ واحد کم می‌شود که این سبب حرکت توپ به سمت چپ خواهد شد و این کار تا زمانی که کاربر کلید دیگری از کلیدهای جهت‌دار را فشار ندهد، هر ۲۰ میلی‌ثانیه یک بار تکرار می‌شود. این امر سبب خواهد شد تا توپ با یک حرکت ملایم به سمت چپ حرکت کند.

به همین صورت وقتی کلید جهت‌دار راست فشرده شود Case سوم اجرا می‌شود و با اضافه کردن ۵۰ واحد به خصوصیت Left توپ، آن را به سمت راست فرم حرکت می‌دهد و اگر کلید جهت‌دار بالا یا پایین را فشار دهد به ترتیب Case دوم یا چهارم اجرا شده و با کاهش یا افزایش خصوصیت Top توپ، آن را به بالا یا پایین حرکت خواهد داد. در ضمن با استفاده از یک دستور If، در هر بار مکان قرارگیری توپ بررسی می‌شود تا در صورت برخورد توپ با دیواره‌های فرم، کنترل زمان‌سنج از کار بیفتد و پیام Game Over

با استفاده از یک کادر پیام نشان داده شود.

۴ - پروژه و فرم را با نام ball ذخیره کرده، سپس آن را اجرا کنید.

۵ - کلید Enter را فشرده و بلافاصله با کلیدهای جهت‌دار، توپ را کنترل کنید، سپس

کلیدها را رها کنید تا توپ با دیوارهای فرم برخورد کرده و پیام GameOver نمایش داده شود.

۶ - به پنجره ویژوال بیسیک بازگردید و برای آن که زمان سپری شده نیز نمایش داده

شود یک کنترل زمان سنج دیگر روی فرم قرار دهید، سپس خصوصیت Interval آن را

روی ۱۰۰۰ میلی‌ثانیه و خصوصیت Name و Enabled آن را به ترتیب روی tmrClock و

False تنظیم کنید.

۷ - اکنون به رویداد KeyDown بروید و در اولین Case دستور Select Case

tmrClock.Enabled=True را تایپ کنید تا با شروع به کار زمان سنج اول، زمان سنج دوم

نیز شروع به کار کند.

۸ - دستور Dim lngTime As Long را در بخش تعاریف ماژول فرم و دستور

lngTime=lngTime+۱ را در رویداد ( ) tmrClock\_Timer تایپ کنید.

۹ - به رویداد ( ) tmrTimer\_Timer بروید و دستور MsgBox را به صورت بعد تغییر دهید

تا در پایان بازی زمان به دست آمده نمایش داده شود.

**MsgBox «Game Over ! your Time Is: " \_**

**+Str(lngTime)+" Second.",vbCritical,"The End"**

۱۰ - تغییرات ایجاد شده را ذخیره کنید و پروژه را مجدداً اجرا و آزمایش نمایید و

در پایان به اجرای پروژه خاتمه داده، به پنجره ویژوال بیسیک بازگردید.

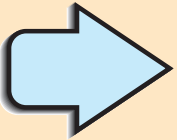
تمرین:



بازی ball را طوری تنظیم کنید که امکانات زیر به آن اضافه شود:

الف- امکان افزایش و کاهش سرعت حرکت توپ با کلیدهای + و - وجود داشته باشد.

ب- امکان توقف بازی با کلید Pause و شروع مجدد آن با کلید Esc وجود داشته باشد.



## Learn in English

### Responding to Mouse Events

You can use the `MouseDown`, `MouseUp`, and `MouseMove` events to enable your applications to respond to both the location and the state of the mouse. These mouse events are recognized by most controls.

Event	Description
<code>MouseDown</code>	Occurs when the user presses any mouse button.
<code>MouseUp</code>	Occurs when the user releases any mouse button.
<code>MouseMove</code>	Occurs each time the mouse pointer is moved to a new point on the screen.

Dash  
Dot  
Middle  
Pointer  
Recognize  
Responding  
Shape  
Solid  
State  
Transparent

واژه‌نامه  
خط تیره  
نقطه  
میانی، وسط  
اشاره‌گر  
تشخیص دادن  
پاسخ دادن  
شکل  
توپر  
حالت، وضعیت  
شفاف

## خلاصه مطالب

- رویدادهای ماوس عبارتند از: MouseDown، MouseUp، MouseMove و DragDrop.
- رویداد MouseDown زمانی اجرا می‌شود که یکی از کلیدهای ماوس فشرده شود.
- رویداد MouseUp زمانی اجرا می‌شود که یکی از کلیدهای ماوس که فشرده شده است، رها شود.
- رویداد MouseMove زمانی اجرا می‌شود که اشاره‌گر ماوس روی فرم یا یک کنترل به حرکت درآید.
- رویداد DragDrop زمانی اجرا می‌شود که موقعیت یک کنترل روی فرم به وسیله ماوس تغییر کند.
- از کنترل Shape برای ایجاد اشکال هندسی و از کنترل Line برای ایجاد خط استفاده می‌شود.
- رویدادهای صفحه کلید عبارتند از: KeyDown، KeyUp و KeyPress.
- رویداد KeyDown و KeyPress زمانی اجرا می‌شوند که یک کلید در صفحه کلید فشرده شود.
- رویداد KeyUp زمانی اجرا می‌شود که یک کلید فشرده شده در صفحه کلید رها شود.
- خصوصیت KeyPreview نحوه اجرای رویدادهای صفحه کلید را بین فرم و کنترل آن تعیین می‌کند.



## آزمون نظری

۱- کدام رویداد در صفحه کلید بین کاراکترهای حرفی کوچک و بزرگ تفاوت قائل می‌شود؟

الف - KeyDown      ب - KeyPress      ج - KeyUp      د - KeyPreview

۲ - کدام خصوصیت در کنترل Shape, نوع شکل را معین می‌کند؟

الف - BorderStyle      ب - Style      ج - Shape      د - Appearance

۳ - برای استفاده از رویداد DragDrop تنظیم کدام خصوصیت در کنترل الزامی است؟

الف - DragIcon      ب - DragMode      ج - DragDrop      د - DropMode

۴ - کدام خصوصیت در کنترل Line نوع خط رسم شده را تعیین می‌کند؟

الف - Border Style      ب - Border Width      ج - Border Color      د - Draw Mode

۵ - کدام گزینه در رابطه با رسم یک نقطه در روی فرم مناسب است؟

الف - Point      ب - PSet      ج - PointSet      د - Step

۶ - کدام آرگومان در رویداد DragDrop, کنترل جابه‌جا شده را مشخص می‌کند؟

الف - Control      ب - Source      ج - Shift      د - Ctrl

۷ - واحد اندازه‌گیری پیش‌فرض در کنترل Shape چیست؟

الف - سانتی‌متر      ب - اینچ      ج - Twip      د - Pixel

۸ - در صورتی که خصوصیت KeyPreview روی مقدار True تنظیم شود ..... .

الف - رویدادهای صفحه کلید فرم زودتر از کنترل‌ها اجرا می‌شوند.

ب - رویدادهای صفحه کلید کنترل زودتر از فرم اجرا می‌شوند.

ج - فقط رویدادهای صفحه کلید کنترل‌ها اجرا می‌شوند.

د - فقط رویدادهای صفحه کلید فرم اجرا می‌شوند.

۹ - کدام آرگومان در رویداد KeyUp کداسکی کلید فشرده شده را مشخص می‌کند؟

الف - KeyCode      ب - Key      ج - Shift      د - Keys

۱۰ - کدام آرگومان در رویداد KeyDown می‌تواند فشرده شدن کلیدهای تغییر حالت

مانند Alt را مشخص کند؟

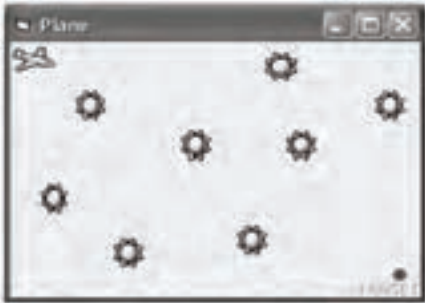
الف - Shift      ب - Alt      ج - Control      د - KeyCode

۱۱ - Which of the following mouse events can be used to respond mouse button clicks?

a- MouseUp      b- MouseDown      c- MouseMove      d- DragDrop

- ۱۲- تفاوت بین رویدادهای صفحه کلید KeyDown و Keypress را توضیح دهید.
- ۱۳- رویدادهای ماوس را نام برده و هر یک را همراه با آرگومان‌های مربوطه به اختصار شرح دهید.
- ۱۴- نقش و کاربرد آرگومان‌های KeyCode و Shift را در رویدادهای KeyDown و KeyUp بیان کنید.
- ۱۵- وظیفه آرگومان KeyAscii را در رویداد KeyPress توضیح دهید.
- ۱۶- رویداد DragDrop را همراه با آرگومان‌های آن توضیح دهید.
- ۱۷- وظیفه آرگومان‌های Button و Shift را در رویدادهای ماوس مشخص کنید.

### آزمون عملی



- ۱- یک بازی طراحی کنید که مطابق شکل زیر، بازیکن یک هواپیمای متحرک را از میان موانع و بدون برخورد با آن‌ها عبور داده به مقصد برساند و بر اساس زمان تلف شده امتیاز وی محاسبه شود، به علاوه در صورت برخورد وی با دو مانع ادامه بازی امکان‌پذیر نباشد.

- ۲- پروژه‌ای طراحی کنید که مطابق شکل بعد حرکت یک زیردریایی در زیر آب را شبیه‌سازی نماید و علاوه بر امکان حرکت زیردریایی با صفحه کلید، بتوان از پریسکوپ زیردریایی در نزدیکی سطح آب نیز استفاده کرد و در ضمن خورشید و ابرهای موجود در آسمان نیز به صورت متحرک باشند و شرایط طبیعی مثل روز و شب را پدید آورند.

