

Adobe Flash



فصل پانزدهم

توانایی به کارگیری Action Script

عملی	نظری
۱۲	۶

هدف های رفتاری

در پایان این فصل از هنرجو انتظار می رود:

- کاربرد اکشن اسکریپت را تشریح کند.
- اصول کد نویسی در اکشن اسکریپت را فراگیرد و عملاً آن را در برنامه نویسی مورد استفاده قرار دهد..
- حرکت Timeline با استفاده از اسکریپت ها را تشریح کند.
- نحوه استفاده از متدهای عمومی را فرا گرفته و عملاً آنرا در برنامه استفاده کند.
- نحوه استفاده از اسکریپت های خارجی را تشریح کند.
- انواع داده ها و متغیرها را نام ببرد.
- اصول استفاده از کلاسهای Flash را فرا گرفته و بتواند از آنها در پروژه ها استفاده کند.
- ساختار جملات شرطی را توضیح دهد.
- ساختار حلقه های تکرار را فرا گرفته و بتواند عملاً آن را در یک برنامه پیاده سازی کند.
- اصول تعریف نمودن تابع توسط کاربر را تشریح نموده و بتواند یک تابع را پیاده سازی نماید.
- اصول افزودن اسکریپت به دکمه ها و Movie Clip ها را بتواند عملاً در برنامه انجام دهد
- بتواند یک شی Movie Clip را به کلاس تبدیل کند

مقدمه

در فصل‌های قبل با نحوه ایجاد انیمیشن‌های مختلف آشنا شده‌اید تا به حال امکان کنترل انیمیشن‌ها به صورت تعاملی وجود نداشت در این فصل با استفاده از Action Script می‌توانید ضمن کنترل انیمیشن‌ها، رابط کاربر تعاملی نیز ایجاد نمائید تا کار شما به فیلم نمایشی قابل کنترل و تعاملی تبدیل شود همچنین می‌توانید ویژگی‌های فیلم را در حین نمایش تغییر دهید در Action Script با استفاده از کدها می‌توانید کنترل‌های مورد نظر را روی فیلم اعمال نمائید.

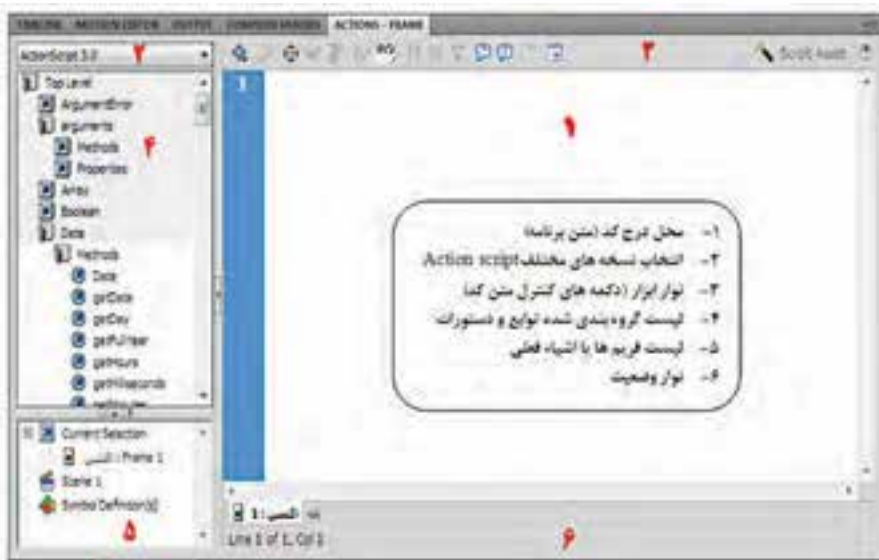
۱-۱۵ آشنایی با مفهوم اسکریپت و کاربرد اسکریپت نویسی

مجموعه کدهایی که در داخل یک فایل متنی قرار گرفته و قابلیت اجرا داشته باشند را اسکریپت می‌گویند. اسکریپت‌ها می‌توانند عملیات مختلفی را انجام دهند. آنها می‌توانند یک فیلم را متوقف کرده یا اجرا کنند، برنامه‌نویسان، اسکریپت‌ها را زبان مشترک خود با نرم‌افزار می‌دانند، به طوری که توسط آنها می‌توانند یک پروژه را تحت کنترل خود در آورند.

Action Script زبان برنامه‌نویسی است که اجازه ایجاد تعامل^۱، کنترل، پخش و نمایش داده‌ها به برنامه کاربردی ایجاد شده را می‌دهند. آخرین نسخه Action Script در فلش CS4، Action Script ۳ می‌باشد.

۲-۱۵ پانل اسکریپت‌نویسی در فلش

این پانل برای ایجاد و ویرایش کدهای اسکریپت برای کنترل موضوعات و فریم‌ها مورد استفاده قرار می‌گیرد و برای نمایش آن می‌توان از منوی Window گزینه Actions را انتخاب کرده یا از کلید F9 استفاده کرد.



شکل ۱-۱۵

نکته: به مجموعه بخش‌های ۲، ۴ و ۵ جعبه ابزار (Toolbox) می‌گویند.

با یک مثال ساده به اهمیت استفاده از Action Script می‌پردازیم. فرض کنید شما یک فیلم را ایجاد نموده‌اید اگر آن را تست کنید (Ctrl+Enter را بزنید) فیلم شما از فریم ۱ تا آخرین فریم ایجاد شده نمایش داده می‌شود و پس از اتمام نمایش مجدداً نمایش تکرار می‌گردد و تا وقتی که پنجره نمایش فیلم را نبندید این کار ادامه پیدا می‌کند برای توقف فیلم در انتهای نمایش باید از کدهای Action Script استفاده نمائید.

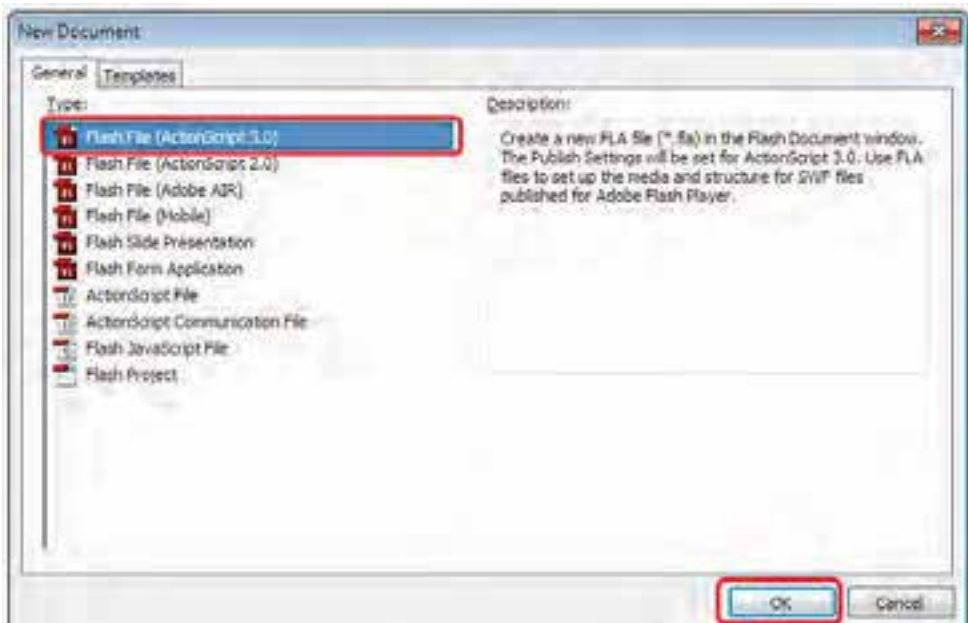
۱۵-۳ کنترل Time Line با استفاده از Action Script

برای کنترل حرکت Timeline با استفاده از اسکریپت‌ها می‌توان از متدهای `stop()`، `gotoAndPlay()`، `gotoAndStop()`، `nextFrame()`، `prevFrame()`، `play()` استفاده نمود.

تمرین ۱: توقف فیلم در فریمی خاص

۱- یک سند جدید به نام `stop fla` از نوع `Flash File (ActionScript 3.0)` ایجاد نمائید.

۱-۱- از منوی `File` گزینه `New...` را کلیک نموده سپس در کادر `New Document` گزینه `Flash File (ActionScript 3.0)` را انتخاب و در نهایت دکمه `OK` را کلیک نمائید



شکل ۱۵-۲

۱-۲- از منوی `File` گزینه `Save` را کلیک نموده سپس در پنجره `Save As` در بخش `File name` نام فایل `(stop)` را وارد نموده و در نهایت دکمه `Save` را کلیک نمائید



شکل ۱۵-۳

۲- نام لایه ۱ در TIMELINE که Layer1 می‌باشد به «محتوی» تغییر دهید (با دابل کلیک نمودن بر روی نام لایه)

۳- با کلیک بر روی New Layer در TIMELINE لایه جدیدی ایجاد نموده و نام آن را به «اکشن» تغییر دهید.

توجه: برای نوشتن کدهای Action Script یک لایه اختصاصی برای آن ایجاد نمایید



شکل ۱۵-۴

۴- لایه «محتوی» را انتخاب نموده و با استفاده از ابزار Rectangle Tool چهارضلعی در گوشه چپ بالای صحنه (Stage) ترسیم نمایید. (ابعاد چهارضلعی ۵۰ در ۵۰ باشد $x=$, $y=$, $w=5$, $h=5$) و به رنگ دلخواه



شکل ۱۵-۵

۵- در لایه «محتوی» فریم ۳۰ را انتخاب نموده سپس کلید F6 را (برای تبدیل فریم انتخاب شده به یک فریم کلیدی) فشار دهید



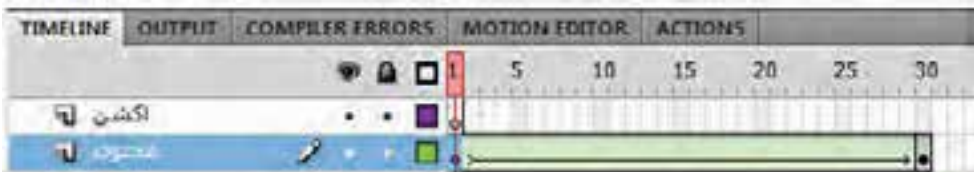
شکل ۱۵-۶

۶- چهار ضلعی را با استفاده از ابزار Selection Tool انتخاب نموده و به گوشه پائین سمت راست صحنه (Stage) انتقال دهید (با درگ نمودن) $x=498$, $y=347$



شکل ۱۵-۷

۷- بر روی یکی از فریم‌های ۱ تا ۳۰ در لایه «محتوی» کلیک راست نموده سپس گزینه Create shape Tween را انتخاب نمائید تا Shape Tween ایجاد شود



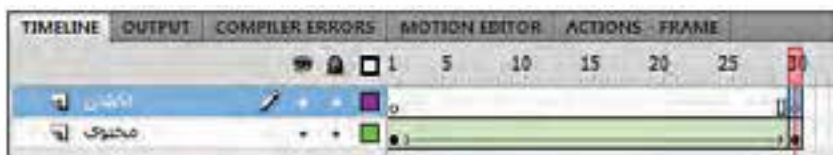
شکل ۱۵-۸

۸- با استفاده از $Ctrl+Enter$ فیلم خود را تست کنید خواهید دید که چهار ضلعی از گوشه بالا به سمت گوشه پائین حرکت می‌کند پس از پایان حرکت یعنی تا رسیدن به فریم ۳۰ مجدداً به فریم ۱ برگشته و دوباره حرکت ادامه می‌یابد حال ما می‌خواهیم با استفاده از Action Script کاری کنیم که به محض رسیدن چهار ضلعی به گوشه پائین صحنه نمایش فیلم متوقف شود



شکل ۹-۱۵

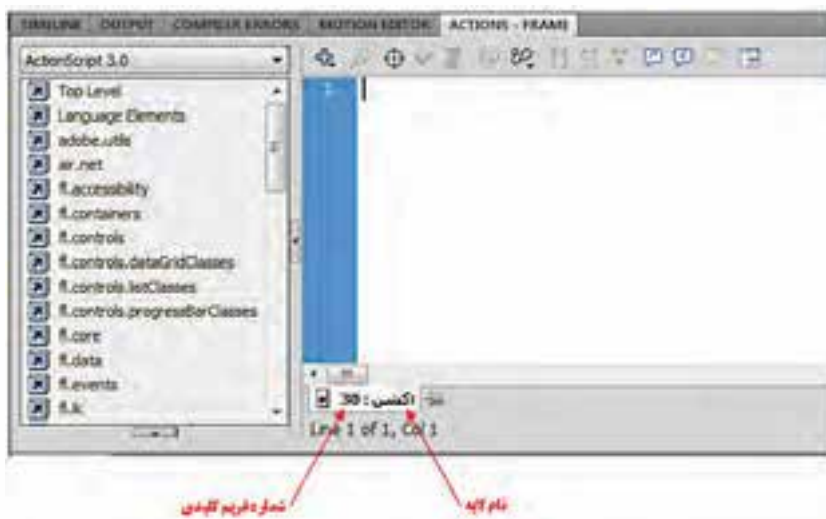
۹- در لایه «اکشن» فریم ۳۰ را انتخاب نموده و برای ایجاد فریم کلیدی، کلید F7 را فشار دهید



شکل ۱۰-۱۵

۱۰- در حالی که فریم ۳۰ انتخاب شده است پنجره Action Script را فعال نمائید (از منوی Window گزینه Actions و یا اینکه از کلید F9 استفاده نمائید)

توجه: فقط در فریم‌های کلیدی می‌توان در پنجره Action Script کد نوشت.



شکل ۱۱-۱۵

۱۱- در محل مکان نما در پنجره Action Script کد زیر را برای توقف فیلم بنویسید

stop();



شکل ۱۲-۱۵

متد stop(); برای توقف فیلم در فریم کلیدی جاری مورد استفاده قرار می‌گیرد. و آرگومان ورودی ندارد (به عبارت داخل پرانتز آرگومان می‌گویند)

۱۲- برای ذخیره کارهای انجام شده از Ctrl+S استفاده نمایید.

۱۳- با استفاده از Ctrl+Enter فیلم خود را تست کنید خواهید دید که چهار ضلعی از گوشه بالا به سمت گوشه پائین حرکت می‌کند و به محض رسیدن به فریم ۳۰ فیلم متوقف می‌شود. و دیگر شما حرکتی روی صحنه نخواهید داشت.

نکته ۱: در موقع کدنویسی Caps Lock خاموش باشد. و به قالب حروف به لحاظ کوچک یا بزرگ بودن دقت نمایید.

نکته ۲: در پایان هر دستور از ; (Semi Colon) استفاده نمایید.

نکته ۳: بعد از نوشتن کد Action Script در فریم کلیدی، حرف α به فریم مورد نظر در TIMELINE اضافه می‌گردد.



شکل ۱۳-۱۵

تمرین ۲: حال کلیپ بالا را طوری می‌خواهیم تغییر دهیم که ابتدا فیلم تا آخرین فریم (فریم ۳۰) پخش شده و سپس در فریم ۱۵ متوقف شود

۱- فایل قبلی را با نام gotoAndStop.fla ذخیره نمایید.

۱-۱- از منوی File گزینه Save As ... را کلیک نموده سپس در پنجره Save As در بخش File name

نام فایل (gotoAndStop) را وارد نموده و در نهایت دکمه Save را کلیک نمایید

۲- فریم ۳۰ در لایه «اکشن» را انتخاب نموده و با استفاده از کلید F9 پنجره Action Script را فعال نمایید.

۳- کد نوشته را پاک نموده و کد زیر را بنویسید

gotoAndStop(15);



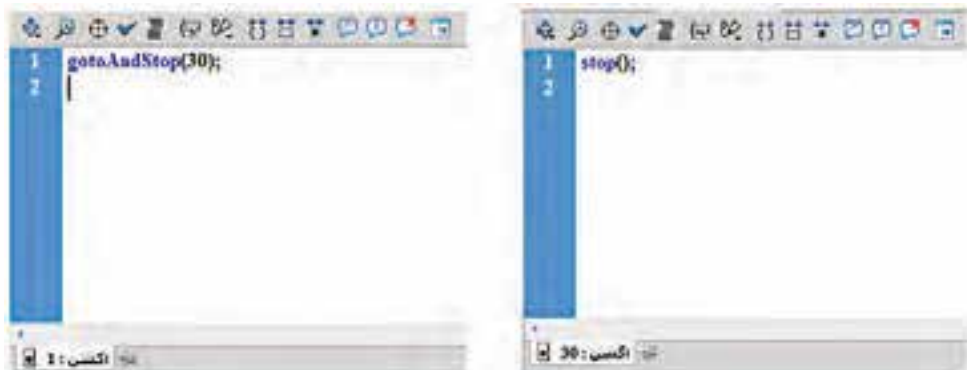
شکل ۱۴-۱۵

کد متد gotoAndStop () برای انتقال هد پخش (Playhead) در فریم مشخص شده و همچنین توقف در آن فریم می‌باشد داخل پرانتز می‌توان شماره فریم را نوشت.

۱۲- برای ذخیره کارهای انجام شده از Ctrl+S استفاده نمایید.

۱۳- با استفاده از Ctrl+Enter فیلم خود را تست کنید خواهید دید که چهار ضلعی از گوشه بالا به سمت گوشه پائین حرکت می‌کند و به محض رسیدن به فریم ۳۰ هدپخش به فریم ۱۵ منتقل شده و فیلم متوقف می‌شود.

نکته: اگر gotoAndStop(30) را در فریم کلیدی ۱ نوشته و کدهای نوشته شده در فریم ۳۰ را پاک کنید نتیجه با فایل stop, یکسان خواهد بود. یعنی با تست فیلم ابتدا تا فریم ۳۰ پخش شده و متوقف می‌شود



شکل ۱۵-۱۵

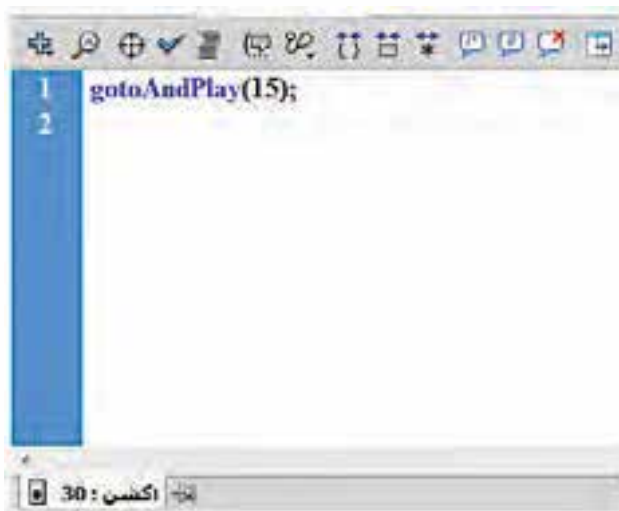
تمرین ۳: حال کلیپ بالا را طوری می‌خواهیم تغییر دهیم که ابتدا فیلم تا آخرین فریم (فریم ۳۰) پخش شده و سپس در دفعات بعدی از فریم ۱۵ تا انتها پخش شود.

۱- فایل تمرین قبلی را با نام gotoAndPlay.fla ذخیره نمائید.

۱-۱- از منوی File گزینه Save As ... را کلیک نموده سپس در پنجره Save As در بخش File name نام فایل (gotoAndPlay) را وارد نموده و در نهایت دکمه Save را کلیک نمائید.

۲- فریم ۳۰ در لایه «اکشن» را انتخاب نموده و با استفاده از کلید F9 پنجره Action Script را فعال نمائید.

۳- بجای gotoAndStop(15); از gotoAndPlay(15); استفاده نمائید



شکل ۱۵-۱۶

کلید متد gotoAndPlay () برای انتقال هد پخش در فریم مشخص شده به طوریکه پخش فیلم از آن فریم ادامه می‌یابد. داخل پرانتز می‌توان شماره فریم را نوشت.

۱۲- برای ذخیره کارهای انجام شده از Ctrl+S استفاده نمائید.

۱۳- با استفاده از Ctrl+Enter فیلم خود را تست کنید خواهید دید که چهار ضلعی از گوشه بالا به سمت گوشه پائین حرکت می‌کند و به محض رسیدن به فریم ۳۰ هدپخش به فریم ۱۵ منتقل شده و فیلم از فریم ۱۵ تا ۳۰ پخش شده و مجدداً به فریم ۱۵ انتقال یافته و پخش تا فریم ۳۰ ادامه می‌یابد.

نکته: اگر بخواهید پخش فیلم از فریم ۱۵ شروع شود باید gotoAndPlay(15) در فریم ۱ لایه «اکشن» بنویسید.

افزودن یادداشت‌های توضیحی

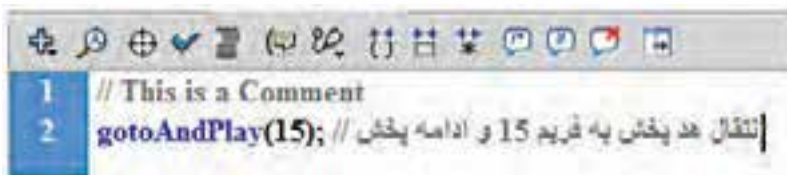
Comments یا توضیحات به بخش‌هایی از یک برنامه گفته می‌شود که در اجرای برنامه هیچ تاثیری نداشته ولی برنامه نویسان از آنها برای افزایش درک برنامه و بررسی‌های احتمالی بعدی استفاده می‌کنند. اگر شما در فیلم‌های فلش خود از یک توضیح کمکی استفاده نمایید، درک وقایع آن راحت‌تر خواهد شد. وجود این توضیحات و یادداشت‌ها که هیچ اثری در اندازه فایل فیلم منتشر شده ندارد، بررسی فیلم را سریعتر می‌نماید. اصولاً دو نوع Comment یا توضیحات را می‌توان در Flash ایجاد کرد:

۱. توضیحات برای کدنویسی در Action Script


۲. توضیحات برای فریم کلیدی در خط زمان


۵-۱۵ یادداشت‌های توضیحی در کدنویسی Action Script


برای توضیح یک خطی یا انتهای هر خط دستور، از علامت // در ابتدای توضیحات مورد نظر استفاده می‌کنیم که به رنگ خاکستری نمایش داده می‌شود مانند `// this is a comment`




شکل ۱۷-۱۵

اگر بخواهید دستوری را غیر فعال نمائید می‌توانید در ابتدای دستور از // استفاده نمائید. ضمناً از آیکون  Apply line Comment در نوار ابزار نیز می‌توان برای این منظور استفاده کرد. ضمناً اینک توجه داشته باشید از این روش برای غیر فعال کردن بخشی از کد در Action Script نیز می‌توان استفاده کرد

نکته: اگر شما چند خط از کدهای تایپ شده را انتخاب کنید سپس بر روی آیکون  Apply line Comment در نوار ابزار کلیک کنید تمام خطوط انتخاب شده به عبارت توضیحی تبدیل می‌شوند یعنی در ابتدای تمام خطوط انتخاب شده علامت // قرار می‌گیرد

برای توضیحات چند خطی در Flash نیز از /* */ استفاده می‌شود بطوریکه علامت /* برای شروع توضیحات و علامت */ برای پایان توضیحات مورد استفاده قرار می‌گیرد. ضمناً اینک از آیکون  Apply block Comment در نوار ابزار نیز می‌توان برای این منظور استفاده کرد.

نکته: اگر چند خط از کدهای تایپ شده را انتخاب کنید سپس بر روی آیکون  Apply block Comment در نوار ابزار کلیک کنید تمام خطوط انتخاب شده به عبارت توضیحی تبدیل می‌شوند یعنی در ابتدای اولین خط انتخاب شده علامت /* و در انتهای آخرین خط انتخاب شده علامت */ قرار می‌گیرد



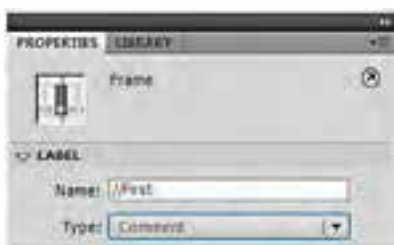
شکل ۱۸-۱۵

۶-۱۵ یادداشت‌های توضیحی برای فریم کلیدی در خط زمان

- علاوه بر اضافه کردن توضیحات به کدهای موجود در پنجره اسکریپت، در Flash می‌توان به یک فریم کلیدی نیز در قالب یک Comment توضیحات اضافه کرد. برای این منظور کافی است مراحل زیر را انجام دهید:
۱. فریم کلیدی مورد نظر خود را انتخاب نمایید.
 ۲. در صورتی که پانل خصوصیات مخفی باشد با انتخاب گزینه Properties از منوی Window آن را نمایش دهید.
 ۳. در بخش Label، در کادر Name ابتدا از علامت // استفاده نموده سپس توضیح مورد نظر را وارد کنید در این صورت Type به نوع Comment تبدیل می‌شود.

نکته: اگر شما ابتدا توضیحات مورد نظر را در کادر Name وارد نمائید سپس Type را به نوع Comment تبدیل کنید علامت // به ابتدای آن اضافه می‌شود.

۴. کلید Enter را برای اعمال توضیحات فشار دهید.



شکل ۱۹-۱۵

- بعد اضافه کردن توضیحات علامت // با رنگ سبز به فریم کلیدی انتخاب شده، اضافه شده و متن توضیح در ادامه نمایش داده می‌شود.



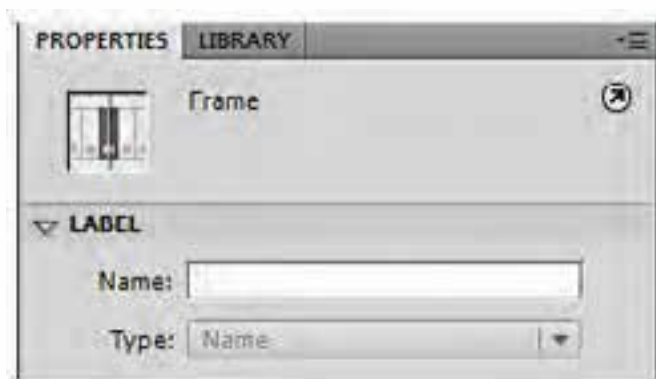
شکل ۲۰-۱۵

۷-۱۵ افزودن برچسب (Label) به خط زمان (Time Line):

برچسب‌ها نیز همانند توضیحات، خط زمان را قابل فهم تر می‌نمایند و به فریم‌های کلیدی اضافه می‌شوند ولی کاربرد دیگری نیز در هنگام استفاده از برنامه‌نویسی Action Script دارند. در زمان کدنویسی بجای شماره فریم مورد نظر خود، می‌توانید برچسب فریم را در دستورات استفاده کنید. این نوع استفاده باعث می‌شود که در صورت اعمال تغییرات در فریم‌ها، شماره فریم قبلی برای دستور Action منظور نشده و فلش همچنان براساس برچسب تعریف شده، فرمان‌های Action Script را بپذیرد. در این حالت برچسب اختصاص داده شده به یک فریم کلیدی، همانند توضیحات به آن اضافه می‌شود.

برای افزودن برچسب به یک فریم (Frame Label)، مراحل زیر را انجام دهید:

۱. فریم کلیدی مورد نظر خود را انتخاب نمایید.
۲. در صورتی که پانل خصوصیات مخفی باشد با انتخاب گزینه Properties از منوی Window آن را نمایش دهید.
۳. در بخش Label، در کادر Name در صورتی که Type از نوع Name باشد نام برچسب مورد نظر را وارد کنید (به طور پیش فرض مقدار Type از نوع Name یا برچسب می‌باشد).
۴. کلید Enter را برای اعمال برچسب فشار دهید.



شکل ۲۱-۱۵

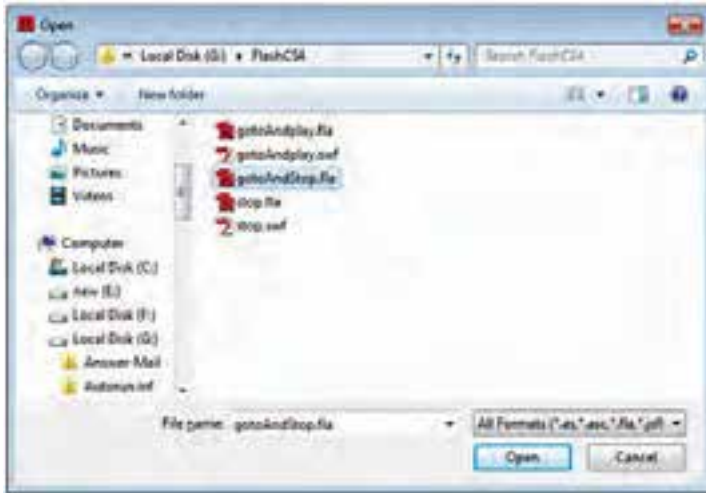
بعد اضافه کردن برچسب، علامت پرچم قرمز به فریم کلیدی انتخاب شده، اضافه شده و برچسب در ادامه نمایش داده می‌شود.



شکل ۲۲-۱۵

تمرین ۴: استفاده از Label یا برچسب

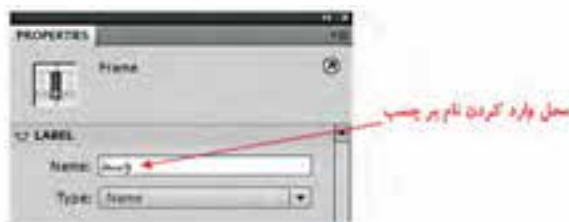
- ۱- فایل gotoAndStop.fla را باز کنید (از منوی File گزینه Open را کلیک نموده سپس در پنجره Open فایل gotoAndStop.fla را انتخاب نموده و در نهایت بر روی دکمه Open را کلیک نمایید



شکل ۲۳-۱۵

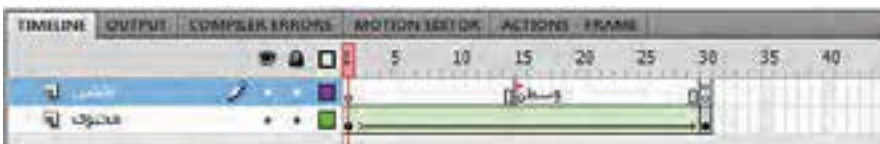
۱۵

- ۲- در لایه «اکشن» فریم ۱۵ را انتخاب نموده و برای ایجاد فریم کلیدی، کلید F7 را فشار دهید
- ۳- از منوی Window گزینه Properties انتخاب نمائید تا پنجره Properties ظاهر شود در بخش Name نام بر چسب را بنویسید



شکل ۲۴-۱۵

- ۳- در لایه «اکشن» فریم ۳۰ را انتخاب نموده و نام بر چسب لایه ۳۰ را «انتها» قرار دهید.



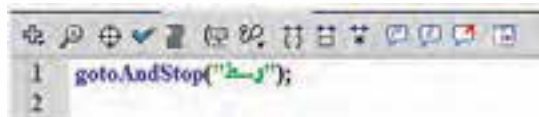
شکل ۲۵-۱۵

همانطور که در Time Line شکل ۲۵-۱۵ دیده می‌شود نام برچسب و علامت پرچم قرمز برای فریم ۱۵ کاملاً نمایان می‌باشد ولی برای فریم ۳۰ نام برچسب و علامت پرچم قرمز مخفی می‌باشد حال اگر در فریم‌های جلوتر مثلاً ۴۰ یک فریم کلیدی ایجاد کنید نام برچسب و علامت پرچم قرمز فریم ۳۰ نیز نمایان می‌شود



شکل ۲۶-۱۵

۴- فریم ۳۰ در لایه «اکشن» را انتخاب نموده و کلید F9 را برای نمایش پنجره Action Script فشار دهید
 ۵- حال بجای عدد ۱۵ در gotoAndStop نام برچسب فریم ۱۵ را بنویسید (نام برچسب را داخل دابل کوتیشن "" بنویسید)



شکل ۲۷-۱۵

۶- فیلم خود را تست کنید خواهید نتیجه نسبت به قبل تفاوتی نخواهد کرد چون ما به جای شماره فریم به برچسب فریم ارجاع داده ایم

کلیه در متدهای gotoAndPlay و gotoAndStop علاوه بر شماره فریم از نام برچسب هم می‌توان استفاده نمود دقت داشته باشید که نام برچسب باید داخل دابل کوتیشن "" نوشته شود.

متد nextFrame() برای انتقال به فریم بعدی بوده و آرگومان ورودی ندارد،

متد prevFrame() برای انتقال به فریم بعدی بوده و آرگومان ورودی ندارد،

متد play() برای پخش فیلم مورد استفاده قرار می‌گیرد.

۸-۱۵ مفاهیم پایه در کدنویسی Flash

۱-۸-۱۵ شیء (Object)

با توجه به اینکه Action Script یک زبان برنامه‌نویسی شیء گرا است شناخت شیء از اهمیت زیادی برخوردار است. در Action Script هر عنصر درون فلش (چه تعاملی باشد و چه نباشد- چه عناصر بصری باشند و چه نباشند) یک شیء محسوب می‌شود. اشیاء در فلش می‌توانند کلیپ‌های فیلم، دکمه‌ها، اشکال داخل Movie، کادرهای متنی یا توابع باشند که در این میان دکمه‌ها و کادرهای متن به عنوان عناصر تعاملی و بصری، اشکال به عنوان عناصر غیرتعاملی و بصری، و توابع نیز اشیاء غیر تعاملی و غیربصری محسوب می‌گردند.

۲-۸-۱۵ کلاس (Class)

گروهی از اشیاء است که ویژگی‌های مشترکی داشته و رفتارهای یکسانی از خود نشان می‌دهند کلاس گفته می‌شود. شیء، نمونه خاصی از کلاس است. به طوری که هر شیء در Action Script دارای کلاس مخصوص به خود می‌باشد. کلاس‌ها دارای سه مشخصه اصلی شامل: ویژگی، متد و رویداد بوده به طوری که این مشخصات، اعضای یک کلاس محسوب می‌شوند. برای این که بیشتر با این مشخصات و نحوه استفاده از آنها در Flash آشنا شوید در ادامه به بررسی آنها می‌پردازیم:

۳-۸-۱۵ خواص (Properties)

از این مشخصه برای تعیین ویژگی‌های شیء در فلش استفاده می‌شود. برای دسترسی به ویژگی‌ها در Action Script، نیاز است که کلیپ فیلم با نام نمونه (Instance Name) فراخوانی شود. برای این منظور کافی است از شکل کلی زیر استفاده کنید:

مقدار = ویژگی . نام شیء

```
MyButton1.width=500;
```

کد فوق باعث می‌شود پهنای شیء MyButton1 به ۵۰۰ تغییر یابد. (MyButton1 نام نمونه شیء روی صحنه یا Stage می‌باشد)

۴-۸-۱۵ متدها Methods

عملیاتی هستند که به صورت تابع نوشته می‌شوند و برای کنترل اشیاء به کار می‌روند مانند: متد متوقف نمودن نمایش فیلم - انتقال نمایش به فریم خاص - متد شروع نمایش از فریمی خاص

۵-۸-۱۵ رویدادها Events

وقایعی هستند که شیء می‌تواند نسبت به آنها واکنش نشان دهد. از رویدادهای مهم می‌توان به کلیک، دابل کلیک، فشار دادن یا رها نمودن دکمه‌ای از صفحه کلید اشاره کرد.

۹-۱۵ شناخت اصول نام گذاری و افزودن Instance name

وقتی شما شیء را روی یک فریم قرار می‌دهید و آن را به نماد Movie clip تبدیل می‌کنید در کادر Properties می‌توانید برای آن یک نام در نظر بگیرید که اصطلاحاً Instance Name گفته می‌شود و از آن برای دسترسی به مشخصات اشیاء در Action Script استفاده می‌شود. (نام نمونه در واقع نام شیء در کدنویسی می‌باشد)

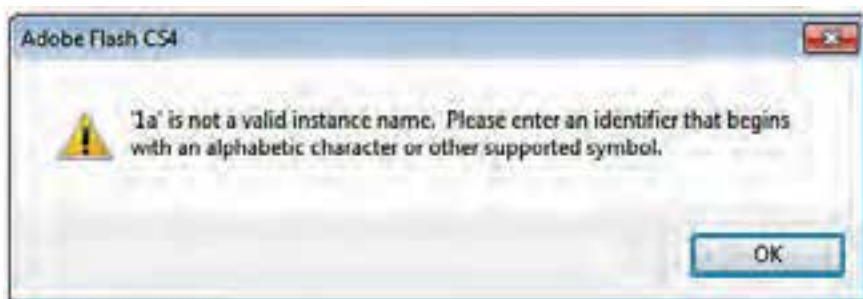


شکل ۲۸-۱۵

۱۰-۱۵ اصول نامگذاری Instance name

برای نام گذاری نمونه‌ها در Flash لازم است به نکات زیر توجه شود:

- ۱- نام باید با حروف یا زیرخط (Underscore) و یا علامت دلار (\$) شروع شود در صورت عدم رعایت این موارد، پیغام خطای زیر ظاهر می‌گردد

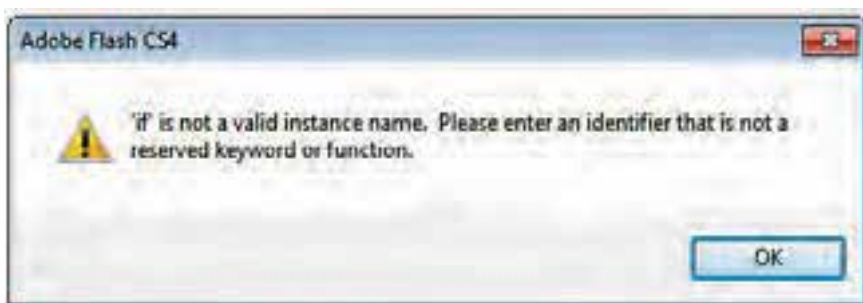


شکل ۲۹-۱۵

- ۲- از کاراکتر فاصله خالی Spacebar یا whitespace نمی‌توان استفاده نمود.

- ۳- نام نمونه باید منحصر به فرد (unique) باشد.

- ۴- از کلمات رزرو شده (reserved keyword) مانند if,for,else,and,or,... نمی‌توان استفاده کرد.



شکل ۳۰-۱۵

۱۱-۱۵ شناخت اصول وارد کردن کد در پانل Actions

همانطور که قبلاً اشاره شد فقط در فریم‌های کلیدی می‌توان کدهای Action Script را وارد نمود لازم

است با نکاتی که در زمان نوشتن کد 3 Action Script باید مورد توجه قرار داد آشنا شوید:

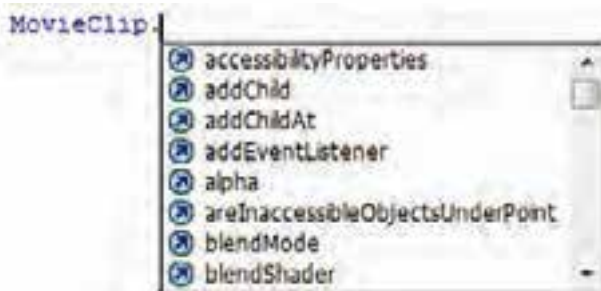
- ۱- حساس بودن به حروف کوچک و بزرگ (Case Sensitive)

کلمات رزرو شده با رعایت حروف کوچک و بزرگ به رنگ آبی دیده می‌شود مانند trace ولی اگر یکی از

حروف را بزرگ بنویسید به دلیل عدم رعایت این اصل به رنگ سیاه دیده می‌شود.

- ۲- استفاده از Semi colon (;) در پایان هر خط استفاده از این علامت به مفهوم مشخص کردن انتهای یک دستور می باشد.
- ۳- نام کلاسها همیشه با حروف بزرگ شروع می شود مانند `MovieClip` و `StageDisplayState`

نکته: بعد از نوشتن نام کلاس و گذاشتن علامت نقطه لیست اتومات مربوطه (Auto list) ظاهر می گردد.



شکل ۳۱-۱۵

- ۴- ثابتها به طور کامل با حروف بزرگ نوشته می شود و فاصله بین آنها به زیرخط (Underscore) پر می شود `FULL_SCREEN`



شکل ۳۲-۱۵

نکته: ثابتها مشخصه هایی هستند که در طول برنامه تغییر نمی کنند. به عنوان مثال اسامی و اعداد جزء دسته ثابتها محسوب می شوند.

- ۵- در کدنویسی با تایپ علامت { و زدن کلید Enter تو رفتگی به طور اتوماتیک رعایت می شود (Automatic Indentation)
- ۶- نام رویدادها با حروف بزرگ نمایش داده می شوند مانند

رویدادهای ماوس

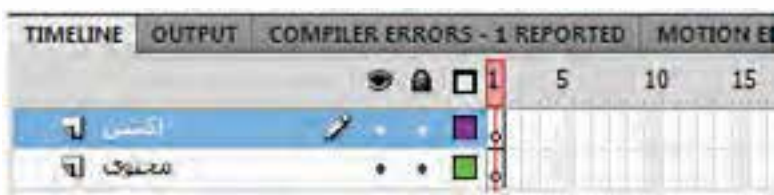
MOUSE_DOWN , MOUSE_UP , MOUSE_MOVE , MOUSE_OVER , CLICK ,
DOUBLE_CLICK ,

رویدادهای صفحه کلید

KEY_DOWN , KEY_UP

تمرین ۵: کار با ویژگی‌های اشیاء

۱. یک سند جدید از نوع Flash File (ActionScript 3.0) ایجاد نموده و با نام ps.fla ذخیره نمایید.
۲. نام لایه Layer1 را به «محتوی» تغییر داده و لایه جدیدی ایجاد کنید و نام آن را به «اکشن» تغییر دهید

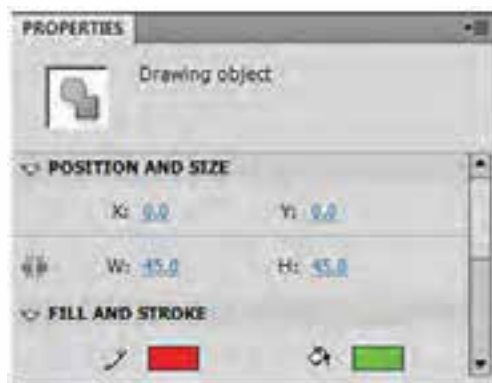


شکل ۳۳-۱۵

۳. در لایه «محتوی» در فریم ۱ با استفاده از ابزار Rectangle Tool یک مربع با مشخصات شکل ۱۵-۳۳-۲ رسم کنید (توجه مستطیل Drawing object باشد) وقتی که ابزار Rectangle Tool را انتخاب می‌کنید مطابق شکل ۱۵-۳۳-۱ در زیر جعبه ابزار، آیکن Object Drawing را انتخاب نمایید سپس اقدام به ترسیم نمایید.

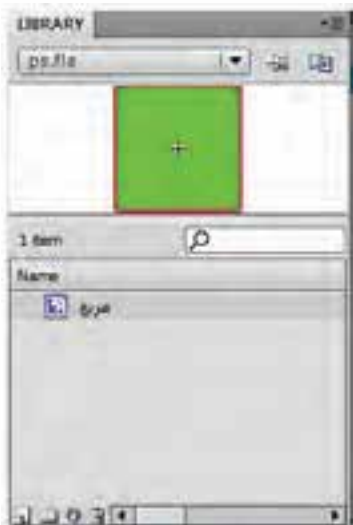


شکل ۳۴-۱-۱۵



شکل ۳۴-۲-۱۵

۴. در لایه «محتوی» مربع را انتخاب و آن را به یک نماد (Symbol) به نام «مربع» تبدیل کنید تا به Library اضافه شود (از منوی Modify گزینه Convert to Symbol ... انتخاب کنید یا از کلید F8 استفاده نمایید)



شکل ۱۵-۳۵

۵. عدد از انتخاب مربع از کادر Properties نام نمونه (Instance Name) آن را در پانل ویژگی‌ها ps وارد نمائید.
۶. فریم ۱ لایه «اکشن» را انتخاب نمائید و با استفاده از کلید F9 پانل Action Script را فعال کنید
۷. کد زیر را وارد کنید.

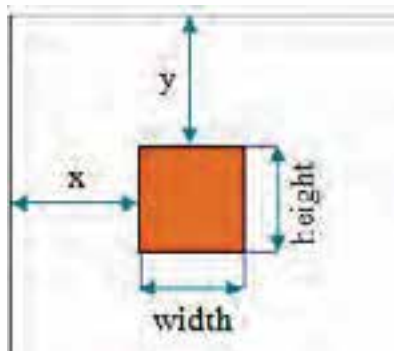
```
trace(ps.x);
trace(ps.y);
```

همانطور که در عبارت فوق مشاهده می‌کنید خاصیت x و y موقعیت مکانی شیء را مشخص می‌کنند.

Ps نام نمونه پنچ ضلعی می‌باشد

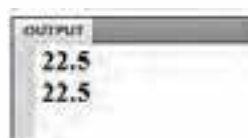
ویژگی x فاصله افقی شیء از سمت چپ صحنه می‌باشد

ویژگی y فاصله عمودی شیء از سمت بالای صحنه می‌باشد



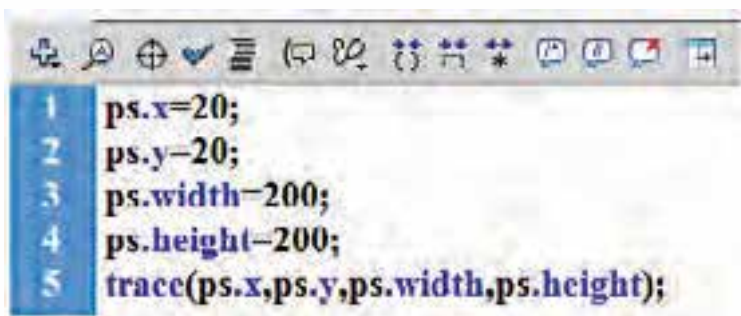
شکل ۱۵-۳۶

ویژگی width پهناى شی می‌باشد
 ویژگی height پهناى شی می‌باشد
 تابع () trace: خروجی آرگومان‌های آن بر روی پانل Output در زمان نمایش فیلم swf چاپ می‌شود.
 مثلاً اگر در داخل پرانتز تابع trace عدد وارد کنید عدد در Output نمایش داده می‌شود
 و یا جمله محاسباتی عددی وارد کنید نتیجه جمله محاسباتی در Output نمایش داده می‌شود
 اگر بخواهید یک عبارت رشته‌ای را در Output نمایش دهید باید آن عبارت را داخل "" بنویسید
 اگر بخواهیم در () trace بیش از یک پارامتر مشخص کنید از علامت , (ویرگول) استفاده نمائید.
 ۸ با استفاده از Ctrl+Enter فیلم خود را تست کنید. در پنجره OUTPUT اعداد 22.5 و 22.5 نمایش داده می‌شود.



شکل ۳۷-۱۵

حال کدها را به صورت زیر تغییر می‌دهیم



شکل ۳۸-۱۵

تشریح کدهای نوشته شده

۱. ps.x=20; باعث می‌شود ویژگی x به عدد ۲۰ تغییر یافته و شی ps به صورت افقی به ۲۰ انتقال می‌یابد
۲. ps.y=20; باعث می‌شود ویژگی y به عدد ۲۰ تغییر یافته و شی ps به صورت عمودی به ۲۰ انتقال می‌یابد
۳. ps.width=200; باعث می‌شود ویژگی width به عدد ۲۰۰ تغییر یافته و پهناى شی ps به ۲۰۰ تغییر می‌یابد (بزرگتر می‌شود)
۴. ps.height=200; باعث می‌شود ویژگی height به عدد ۲۰۰ تغییر یافته و ارتفاع شی ps به ۲۰۰ تغییر می‌یابد (بزرگتر می‌شود)
۵. در پانل Output اعداد زیر نمایش داده خواهد شد.

200 200 20 20

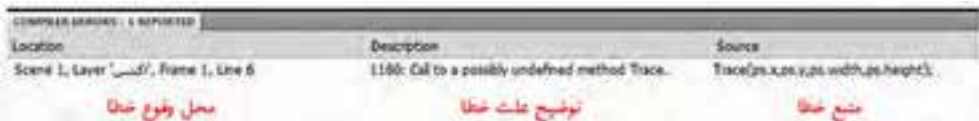
توانایی به کارگیری Action Script

با توجه مثال فوق می‌توان نتیجه گرفت که برای انتقال یک شی می‌توان از ویژگی‌های `x` و `y` استفاده نمود و برای تغییر سایز، از ویژگی‌های `width` و `height` استفاده نمود.

نکته: اگر یک خط از کد دارای خطا باشد شما هیچ عبارتی را در Output به عنوان خروجی نخواهید دید بلکه خطای مورد نظر نمایش داده می‌شود

```
ps.x=20;
ps.y=20;
ps.width=200;
ps.height=200;
trace(ps.x,ps.y,ps.width,ps.height);
Trace(ps.x,ps.y,ps.width,ps.height);
```

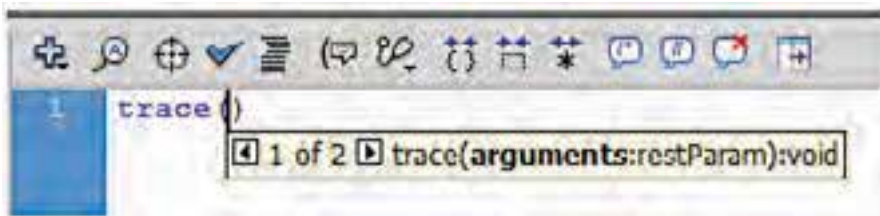
در آخرین خط کد فوق، تابع `trace` با حرف بزرگ نوشته شده است پس از اجرای کدهای فوق، خطای زیر در پانل Compiler Errors نمایش داده می‌شود



شکل ۳۹-۱۵

نکته: بجای تایپ تابع `trace` می‌توان از سه روش زیر برای درج آن در پنجره کدنویسی اقدام نمود

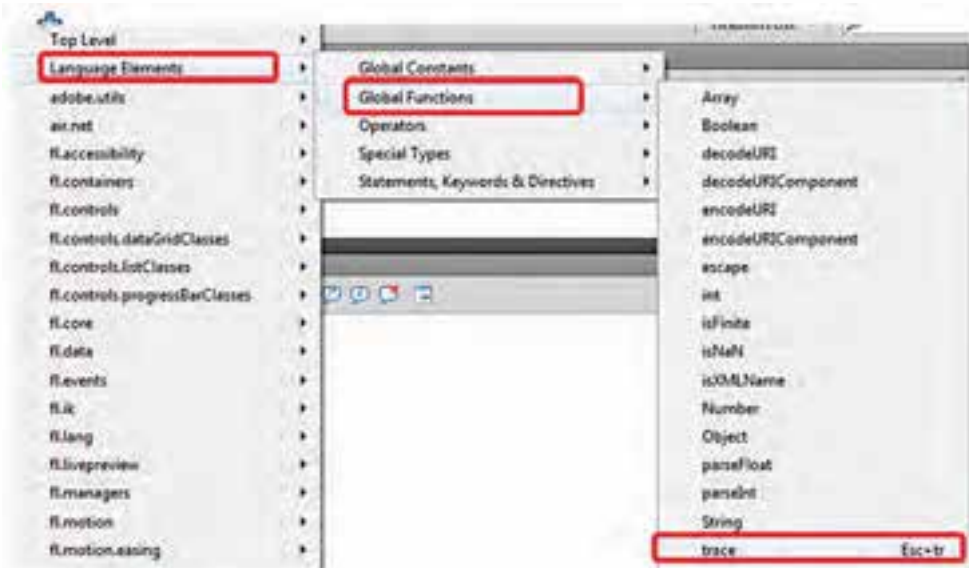
روش اول: استفاده از کلید ترکیبی `Esc + tr` (کلید `Esc` را نگه داشته و حروف `t` و `r` را تایپ کنید) تابع `trace` به صورت زیر درج می‌شود



شکل ۴۰-۱۵

روش دوم: استفاده از آیکن منوی اکشن **Action Pop-up Menu** در نوار ابزار برنامه ابتدا بر روی آیکن منوی اکشن کلیک نموده سپس از مسیر زیر به تابع `trace` دسترسی پیدا می‌کنید.

Language Elements → Global Function → trace

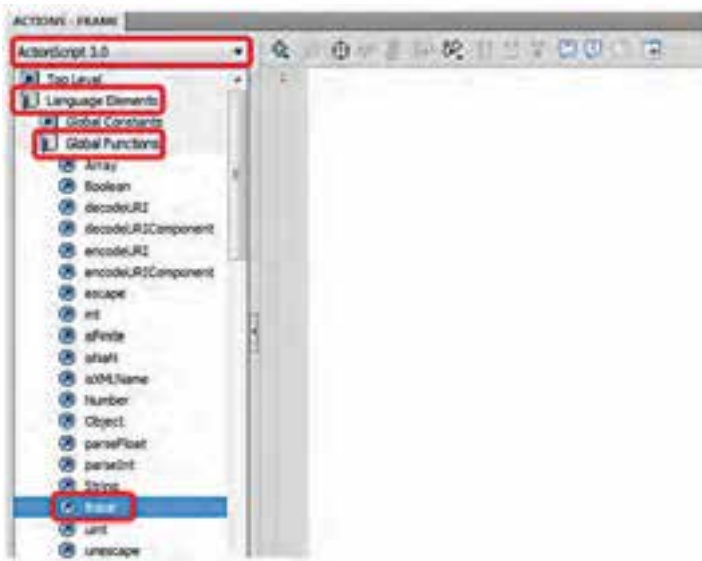


شکل ۴۱-۱۵

روش سوم: استفاده از بخش جعبه ابزار اکشن
ابتدا گزینه Action Script 3.0 را انتخاب نموده سپس در جعبه ابزار اکشن از مسیر زیر به تابع trace دسترسی پیدا می کنید.



Language Elements → Global Function → trace



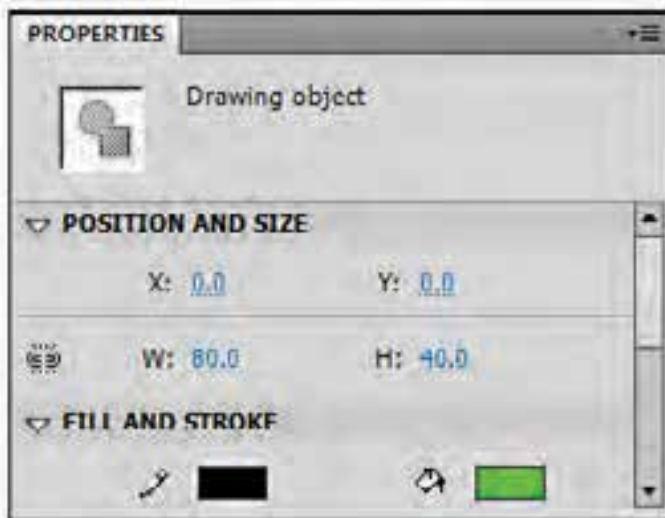
شکل ۴۲-۱۵

- تمرین ۶: کنترل اشیا با استفاده از ویژگی‌های اشیا
۱. یک سند جدید از نوع Flash File (ActionScript 3.0) ایجاد نموده و با نام ps1.fla ذخیره نمایید.
 ۲. سه لایه به نام‌های «مستطیل» - «بیضی» و «اکشن» ایجاد نمایید



شکل ۴۳-۱۵

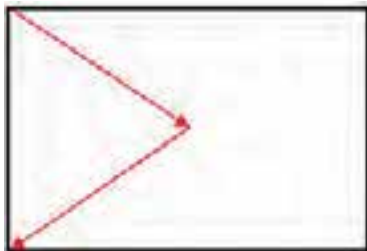
۳. در لایه «مستطیل» در فریم ۱ با استفاده از ابزار Rectangle Tool یک مستطیل با مشخصات ذیل رسم کنید (توجه مستطیل Drawing object باشد)



شکل ۴۴-۱۵

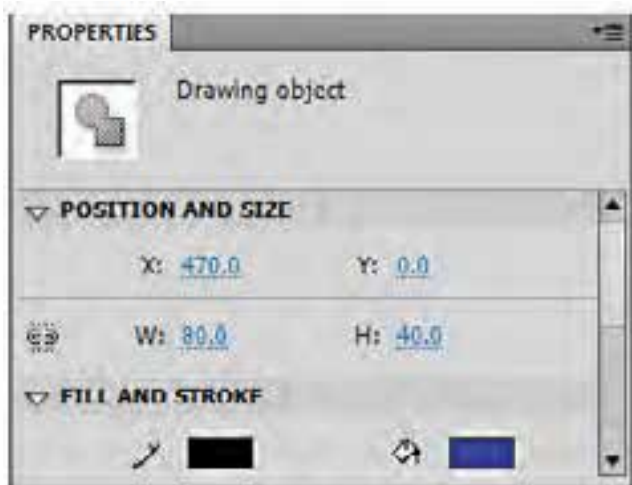
۴. در لایه «مستطیل»، مستطیل را انتخاب و آن را به یک نماد (سمبل) به نام «مستطیل» از نوع Movie Clip تبدیل کنید تا به Library اضافه شود (از منوی Modify گزینه Convert to Symbol... انتخاب کنید یا از کلید F8 استفاده نمایید)
۵. بعد از انتخاب مستطیل از کادر Properties نام نمونه (Instance Name) آن را در پانل ویژگی‌ها ms وارد نمایید.

۶. در لایه «مستطیل» فریم ۳۰ را انتخاب نموده و برای تبدیل آن به فریم کلیدی کلید F6 را فشار دهید. در این حالت مستطیل را به وسط صحنه انتقال دهید
۷. روی یکی از فریم‌های ۱ تا ۳۰ لایه «مستطیل» کلیک راست نموده و گزینه Create Classic Tween را انتخاب نمایید.
۸. در لایه «مستطیل» فریم ۶۰ را انتخاب نموده و برای تبدیل آن به فریم کلیدی کلید F6 را فشار دهید. در این حالت مستطیل را به گوشه پائین سمت چپ صحنه انتقال دهید
۹. روی یکی از فریم‌های ۳۰ تا ۶۰ لایه «مستطیل» کلیک راست نموده و گزینه Create Classic Tween را انتخاب نمایید.
۱۰. اگر فیلم را تست کنید مستطیل در مسیری مانند شکل فلش‌ها حرکت می‌کند



شکل ۴۵-۱۵

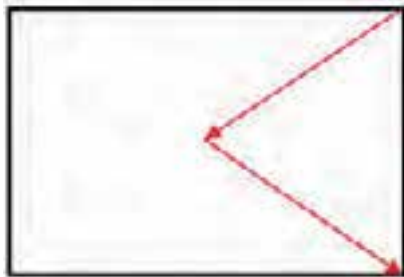
۱۱. در لایه «بیضی» در فریم ۱ با استفاده از ابزار Oval Tool یک بیضی با مشخصات ذیل رسم کنید (توجه بیضی Drawing object باشد)



شکل ۴۶-۱۵

۱۲. در لایه «بیضی»، بیضی را انتخاب و آن را به یک نماد (سمبل) به نام «بیضی» تبدیل کنید تا به Library اضافه شود (از منوی Modify گزینه Convert to Symbol ... انتخاب کنید یا از کلید F8 استفاده نمایید)

۱۳. بعد از انتخاب بیضی از کادر Properties نام نمونه (Instance Name) آن را در پانل ویژگی‌ها bz وارد نمائید.
۱۴. در لایه «بیضی» فریم ۳۰ را انتخاب نموده و برای تبدیل آن به فریم کلیدی کلید F6 را فشار دهید. در این حالت بیضی را به وسط صحنه انتقال دهید (دقیقا روی مستطیل)
۱۵. روی یکی از فریم‌های ۱ تا ۳۰ لایه «بیضی» کلیک راست نموده و گزینه Create Classic Tween را انتخاب نمائید.
۱۶. در لایه «بیضی» فریم ۶۰ را انتخاب نموده و برای تبدیل آن به فریم کلیدی کلید F6 را فشار دهید. در این حالت مستطیل را به گوشه پائین سمت راست صحنه انتقال دهید
۱۷. روی یکی از فریم‌های ۳۰ تا ۶۰ لایه «بیضی» کلیک راست نموده و گزینه Create Classic Tween را انتخاب نمائید.
۱۸. اگر فیلم را تست کنید بیضی در مسیری مانند شکل فلش‌ها حرکت می‌کند



شکل ۴۷-۱۵

در نهایت خط زمان Time Line به صورت شکل زیر در خواهد آمد



شکل ۴۸-۱۵

۱۹. در لایه «اکشن» فریم ۳۰ را انتخاب نموده و برای فعال کردن از کلید F9 استفاده نمائید. و کد زیر را وارد کنید

`ms.visible=false;`

اگر فیلم را تست کنید خواهید دید که به محض رسیدن مستطیل به وسط صحنه (فریم ۳۰) مخفی می‌شود و روی صحنه فقط بیضی در مسیر ایجاد در حال حرکت می‌باشد

نکته ۱: برای مخفی کردن یک شی از ویژگی `visible` به صورت زیر استفاده می‌شود

نام شی `visible=false`;

نکته ۲: برای نمایش شی مخفی شده از ویژگی `visible` به صورت زیر استفاده می‌شود

نام شی `visible=true`;

۲۰. حال برای نمایش مجدد مستطیل فریم ۱ را در لایه «اکشن» انتخاب نموده، با استفاده از کلید F9 پنجره Action Script را فعال نمائید و کد زیر را بنویسید

`ms.visible=true`;

حال فیلم را تست نمائید خواهید دید که ابتدا مستطیل به همراه بیضی به طرف وسط صحنه حرکت کرده و به محض رسیدن به وسط صحنه مستطیل مخفی شده و بیضی به حرکت خود ادامه داده و در مرحله بعدی چون هدپخش به فریم یک می‌رسد مستطیل مجدد نمایان شده و این سیکل ادامه می‌یابد.

۲۱. یک لایه جدید به نام «دایره» ایجاد نموده و در فریم ۱ با استفاده از ابزار Oval Tool یک دایره با شعاع ۲۰ (پهنا و ارتفاع ۴۰) در گوشه پائین سمت چپ صحنه رسم کنید.

۲۲. بیضی را به یک نماد (سمبل) به نام «دایره» تبدیل کنید تا به Library اضافه شود (از منوی Modify گزینه Convert to Symbol ... انتخاب کنید یا از کلید F8 استفاده نمائید)

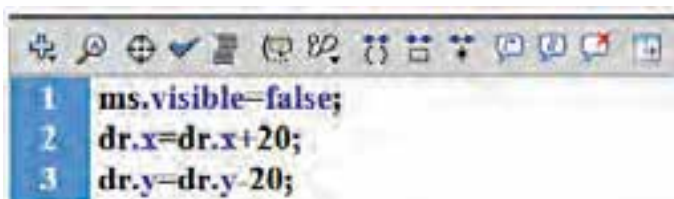
۲۳. بعد از انتخاب دایره از کادر Properties نام نمونه (Instance Name) آن را در پانل ویژگی‌ها `dr` وارد نمائید

۲۴. در لایه «اکشن» فریم ۳۰ را انتخاب نموده و برای فعال کردن از کلید F9 استفاده نمائید. و کد زیر را به کدهای قبلی اضافه نمائید.

`dr.x = dr.x+20;`

`dr.y = dr.y -20;`

کدهای فوق باعث می‌شود بازای هر بار رسیدن هدپخش به فریم ۳۰، دایره ۲۰ واحد به سمت راست و ۲۰ واحد به سمت بالا حرکت کند



شکل ۴۹-۱۵

۲۵. در فریم ۶۰ لایه «اکشن» کدهای زیر را اضافه نمائید.

`dr.width= dr.width+20;`

`dr.height= dr.height+20;`

کدهای فوق باعث می‌شود بازای هر بار رسیدن هدپخش به فریم ۶۰، سایز دایره ۲۰ واحد بزرگتر می‌شود.



شکل ۱۵-۵۰

با تست فیلم

به محض رسیدن هدپخش به فریم ۳۰ مستطیل مخفی شده- دایره نیز از پایین به سمت مرکز صحنه ۲۰ واحد جابجا می شود

به محض رسیدن هدپخش به فریم ۶۰ دایره ۲۰ واحد بزرگتر می شود از فریم ۱ نیز مستطیل نمایان می گردد. و این سیکل ادامه می یابد

برای قرار دادن یک شیء در وسط صحنه می توان از ویژگی های stage استفاده نمود. (stageWidth پهنا و stageHeight ارتفاع) فرض کنید ما می خواهیم در ابتدای نمایش فیلم شیء دایره در مرکز صحنه قرار بگیرد در فریم یک لایه «اکشن» کدهای زیر را اضافه نمائید.

```
;dr.x=stage.stageWidth/2-dr.width/2
;dr.y=stage.stageHeight/2-dr.height/2
```

۱۲-۱۵ شناخت متغیر و انواع آن

متغیر (variable) مکانی در حافظه است که برای نگهداری یک مقدار مورد استفاده قرار می گیرد. مقداری که در متغیر قرار داده می شود، قابل تغییر است (نام آن نیز هم بر همین ویژگی دلالت دارد). وقتی مقداری را در یک متغیر قرار می دهید، مقدار قبلی آن از بین خواهد رفت. متغیرها می توانند انواع داده ها را در خود نگهداری کنند مانند: اعداد، رشته ها، مقادیر بولی، عنصر یا برش فیلم و ... در Action Script، یک متغیر در واقع متشکل از سه بخش متفاوت است:

- نام متغیر
- نوع داده ای که می تواند در متغیر ذخیره شود.
- ارزش واقعی ذخیره شده در حافظه کامپیوتر

۱۳-۱۵ انواع داده ها در Action Script

۱. نوع داده ای String: از نوع رشته ای (مجموعه ای از کاراکترها) است و برای ذخیره کاراکترها تا ۳۲ بیت ظرفیت داشته ضمن اینکه مقدار اولیه آن null (خالی) می باشد.
۲. نوع داده ای int: از نوع عددی صحیح، شامل مقادیر $(2^{31} - 1)$ to $2,147,483,647$ (-2,147,483,648) می باشد و مقدار اولیه آن صفر است.

۳. نوع داده‌ای `uint`: از نوع عددی صحیح بدون علامت شامل مقادیر ۰ تا ۴۲۹۴۹۶۷۲۹۵ می‌باشد و مقدار اولیه آن صفر است.
۴. نوع داده‌ای `Boolean`: از نوع منطقی (بولی) دارای دو مقدار `true` و `false` می‌باشد و مقدار اولیه آن `false` است.
۵. نوع داده‌ای `Number`: از نوع عددی اعشاری است نوع داده‌ای `Number` از قالب اعداد ۶۴ بیتی با دقت مضاعف استفاده می‌کند.

۱۴-۱۵ شناخت اصول ایجاد و مقدار دهی متغیرها (محلی^۱ و فراگیر^۲)

قبل از استفاده از یک متغیر باید آن را اعلان کنید. اعلان یک متغیر یعنی نامگذاری آن و تعیین نوع مقداری که می‌تواند داشته باشد. متغیرها می‌توانند فقط از همان نوعی که اعلان شده‌اند مقدار بگیرند برای تعریف متغیر از کلمه رزرو شده `var` استفاده می‌شود.

فرم کلی تعریف متغیر بدون مقدار اولیه به صورت زیر می‌باشد.

نوع داده‌ای: نام متغیر `var` ;

فرم کلی تعریف متغیر با مقدار اولیه به صورت زیر می‌باشد. برای اختصاص دادن یک مقدار به یک متغیر از علامت = استفاده می‌شود

: نوع داده‌ای = مقدار نام متغیر `var` ;

; مقدار = نوع داده‌ای : نام متغیر `var` ;

مثال:

```
var myVar:String = "iran" ;
```

`myVar`: نام متغیر

`String`: نوع داده‌ای رشته‌ای

"iran": مقدار رشته‌ای (برای مشخص کردن مقدار رشته‌ای حتماً باید آن را داخل دابل کوتیشن مارک "

" قرار دهید)

بجای دستور بالا از دستورات زیر نیز می‌توان استفاده نمود

```
var myVar:String ;
```

```
myVar = "iran";
```

نکاتی که در تعریف متغیر باید رعایت شود

- نام متغیر باید با حروف یا زیرخط (Underscore) و یا علامت دلار (\$) شروع شود.
- نام متغیر می‌تواند شامل حروف، اعداد، زیرخط (Underscore) و یا علامت دلار (\$) باشد.
- از کاراکتر فاصله خالی Blank Space یا whitespace در نام متغیر نمی‌توان استفاده کرد.
- از کلمات رزرو شده (reserved keyword) را به عنوان نام متغیر نمی‌توان استفاده کرد. مانند `if, for, else, and, or, ...`
- بهتر است نام متغیر با حرف کوچک شروع شود.

1 - Local

2 - Global

۶ تعیین نوع داده‌ای اختیاری است ولی بهتر است آن را تعیین کنید .
 ۷. می‌توان با یک var چند متغیر تعریف نمود (علامت جداکننده بین متغیرها ، می‌باشد.
 var i:int, m:int=12, s:String;

int: نوع داده‌ای عددی است

۱۵-۱۵ انواع عملگرها (Operator)

عملگرها علائمی هستند که موجب انجام عملیات بر روی عملوندهای خود می‌شوند. در Flash عملگرهای مختلفی وجود دارد که انواع آنها عبارتند از:

الف) عملگرها ریاضی (+ , - , * , / , %): دارای خروجی عددی هستند
 ب) عملگرهای مقایسه‌ای یا رابطه‌ای (= , < , > , <= , >= , !=): دارای خروجی true (درست) و false (نادرست) هستند

ج) عملگرهای انتساب (= , += , -= , *= , /=): دارای خروجی عددی هستند

د) عملگرهای منطقی (&& , ||): دارای خروجی true (درست) و false (نادرست) هستند

ه) عملگر الحاق رشته (+): دارای خروجی رشته‌ای هستند

برای اینکه بیشتر با عملگرها و کاربرد آنها در برنامه‌نویسی آشنا شوید به مثال‌های زیر توجه کنید:
 عملگر %: برای محاسبه باقیمانده صحیح

$$13\%2 \rightarrow 1 \qquad 14\%5 \rightarrow 4$$

عملگر == مساوی با

$$3==2 \rightarrow \text{false} \qquad 5==5 \rightarrow \text{true}$$

عملگر != نامساوی با

$$3!=2 \rightarrow \text{true} \qquad 5!=5 \rightarrow \text{false}$$

عملگر = انتساب

عدد یک را در متغیر a قرار بده $a=1;$ →

عملگر += انتساب با اضافه کردن به مقدار قبلی

یک واحد به مقدار قبلی a اضافه نموده و در آن قرار می‌دهد $a+=1;$ →

عملگر -= انتساب با کم کردن به مقدار قبلی

یک واحد از مقدار قبلی a کم نموده و در آن قرار می‌دهد $a-=1;$ →

عملگر *= انتساب با ضرب کردن در مقدار قبلی

عدد ۲ را در مقدار قبلی a ضرب نموده و در آن قرار می‌دهد $a*=2;$ →

عملگر /= انتساب با تقسیم کردن بر مقدار قبلی

مقدار قبلی a را بر ۲ تقسیم نموده و در آن قرار می‌دهد $a/=2;$ →

عملگر && معادل And منطقی است (در صورتیکه هر دو عملوند صحیح باشد نتیجه صحیح خواهد

بود در غیر این صورت نتیجه غلط خواهد بود. مثلاً اگر شما بخواهید عبارت ریاضی $1 \leq m \leq 5$ را به زبان Action Script بنویسید باید به صورت زیر بنویسید

$m >= 1 \ \&\& \ m <= 5$

عملگر $\|\|$ معادل OR منطقی است (در صورتیکه حداقل یکی از دو عملوند صحیح باشد نتیجه صحیح خواهد بود)

مثال ۱: عملگرهای ریاضی

<pre> 1 var b:int=3; 2 trace(b-2); 3 trace(b+2); 4 trace(b*2); 5 trace(b/2); 6 trace(b%2); </pre>	<p>خروجی</p>	<table border="1"> <thead> <tr> <th>OUTPUT</th> </tr> </thead> <tbody> <tr><td>1</td></tr> <tr><td>5</td></tr> <tr><td>6</td></tr> <tr><td>1.5</td></tr> <tr><td>1</td></tr> </tbody> </table>	OUTPUT	1	5	6	1.5	1
OUTPUT								
1								
5								
6								
1.5								
1								

شکل ۱۵-۵۱

مثال ۲: عملگرهای انتساب

<pre> 1 var a:int=2; 2 a=3; 3 trace(a+=2); 4 trace(a-=1); 5 trace(a*=2); 6 trace(a/=2); 7 trace(a%=3); </pre>	<p>خروجی</p>	<table border="1"> <thead> <tr> <th>OUTPUT</th> </tr> </thead> <tbody> <tr><td>5</td></tr> <tr><td>4</td></tr> <tr><td>8</td></tr> <tr><td>4</td></tr> <tr><td>1</td></tr> </tbody> </table>	OUTPUT	5	4	8	4	1
OUTPUT								
5								
4								
8								
4								
1								

شکل ۱۵-۵۲

مثال ۳: عملگرهای مقایسه‌ای

<pre> 1 var m:int=3; 2 trace(m==3); 3 trace(m>2); 4 trace(m<2); 5 trace(m>=2); 6 trace(m<=2); 7 trace(m!=2); </pre>	<p>خروجی</p>	<table border="1"> <thead> <tr> <th>OUTPUT</th> </tr> </thead> <tbody> <tr><td>true</td></tr> <tr><td>true</td></tr> <tr><td>false</td></tr> <tr><td>true</td></tr> <tr><td>false</td></tr> <tr><td>true</td></tr> </tbody> </table>	OUTPUT	true	true	false	true	false	true
OUTPUT									
true									
true									
false									
true									
false									
true									

شکل ۱۵-۵۳

مثال ۴: عملگر الحاق رشته (+)



شکل ۵۴-۱۵

تمرین ۷: تعیین تعداد دفعات اجرای فیلم

۱. فایل stop.fla مربوط به تمرین ۱ را باز کنید
۲. فریم ۱ در لایه اکشن را انتخاب نموده و برای فعال کردن پنجره Action Script از کلید F9 استفاده نمائید.
۳. کد زیر را برای تعریف متغیر i از نوع عددی صحیح با مقدار اولیه صفر بنویسید.

```
var i:int=0;
```

۴. در فریم ۳۰ لایه Action Script کد نوشته شده را پاک کرده و کدهای زیر را بنویسید

```
i=i+1;  
trace (i);
```

۵. فیلم را تست کنید (Ctrl+Enter) هر بار که هدپخش به فریم ۳۰ می‌رسد عدد ۱ در پنجره Output نمایش داده می‌شود در صورتیکه ما مقدار i در فریم یک واحد اضافه نموده ایم. علت را تحقیق کنید.
۶. در فریم ۳۰ لایه Action Script کد زیر را اضافه نمائید.

```
gotoAndPlay(2);
```

- حال فیلم را تست کنید. بار که هدپخش به فریم ۳۰ می‌رسد عدد ۱ در پنجره Output نمایش داده می‌شود با دوم دوم عدد ۲ و بار سوم عدد ۳ و الی آخر. علت را تحقیق کنید.
۷. فایل جاری را با نام جدید num.fla ذخیره نمائید.



شکل ۵۵-۱۵

۱۶-۱۵ تعریف متغیر شیء

برای تعریف متغیر شیء از دستور var به شکل زیر استفاده می‌شود

Var شیء = نام کلاس : نام متغیر شیء

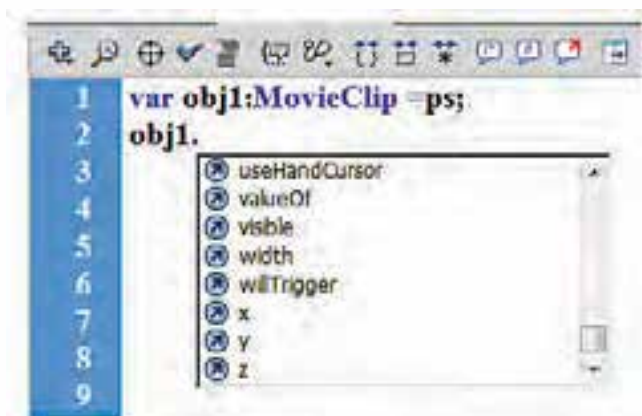
فرض کنید شیء‌ای از نوع Movie Clip با نام نمونه ps داریم می‌خواهیم در Action Script شیء جدیدی به نام obj1 از نوع ps تعریف کنیم از کدهای زیر استفاده می‌نمائیم.

```
var obj1:MovieClip=ps;
```

کلاس‌های Action Script

MovieClip - Date - Timer - Video - Sprite و می‌باشد.

نکته: بعد از تعریف متغیر شیء، اگر شما نام متغیر شیء را تایپ کرده و از کاراکتر نقطه استفاده نمائید. لیست اجزای کلاس مربوط به آن شیء نمایش داده می‌شود



شکل ۵۶-۱۵

۱۷-۱۵ شناخت اصول نوشتن و فراخوانی تابع

توابع، مجموعه‌ای از کدها برای وظایف خاص هستند که برای استفاده مجدد در برنامه به کار می‌روند. از توابع در Action Script برای سازماندهی و گروه‌بندی دستورات در یک برنامه استفاده می‌شود. به طوری که می‌توان با یکبار نوشتن مجموعه دستوراتی خاص، بارها و بارها در بخش‌های مختلف برنامه با فراخوانی تابع مورد نظر، از خروجی تابع استفاده کرد. بر این اساس در Action Script دو نوع تابع را می‌توان مورد استفاده قرار داد که عبارتند از:

- متدها
- بسته‌های تابعی

۱۸-۱۵ نحوه تعریف تابع

هر تابع از یک نام و تعدادی آرگومان یا متغیر ورودی تشکیل شده، به طوری که آرگومان‌ها با وارد کردن مقادیر ورودی به بدنه تابع باعث خواهند شد تابع مورد نظر پس از انجام عملیات بر روی این مقادیر، یک

خروجی را به عنوان نتیجه به برنامه اصلی برگرداند.

```
function نام تابع (لیست آرگومانها) :  
{  
    مجموعه دستورات (بدنه تابع)  
}
```

نکته ۱: در توابع، آرگومان‌ها بوسیله کاما از یکدیگر جدا می‌شوند.

نکته ۲: اگر تابع مقدار بازگشتی نداشته باشد یا مقدار بازگشتی آن void باشد به آن تابع «متد» گفته می‌شود.

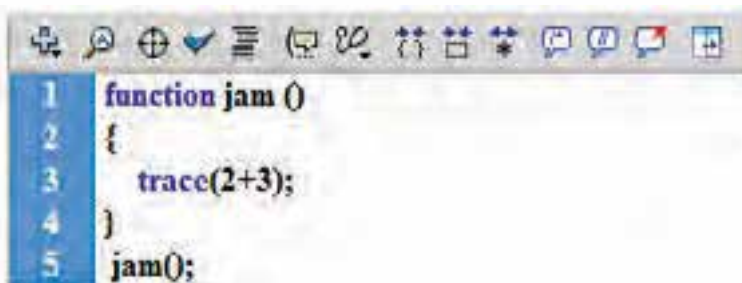
نکته ۳: اگر تابع مقدار بازگشتی داشته باشد به آن تابع «بسته تابعی» گفته می‌شود.

۱۹-۱۵ فراخوانی تابع

پس از این که بدنه اصلی تابع موردنظر و دستورات مربوط به آن توسط function معرفی شدند می‌توان با فراخوانی تابع به شکل زیر، خروجی حاصل از آن را در بخش‌های مختلف برنامه مورد استفاده قرار داد. اگر تابع به صورت متدی تعریف شده باشد فراخوانی تابع به صورت زیر خواهد بود (لیست آرگومانها) نام تابع

نکته: اگر تابع تعریف شده بدون آرگومان باشد برای فراخوانی باید در جلوی نام تابع از پرانتز () استفاده شود.

اگر تابع به صورت بسته تابعی تعریف شده باشد فراخوانی تابع به صورت زیر خواهد بود (لیست آرگومانها) نام تابع = متغیر
باید از خروجی بسته تابعی با انتساب یا در ورودی تابع دیگر استفاده نمود.
مثال ۱: تابعی که آرگومان ورودی ندارد - اعداد ۲ و ۳ را با هم جمع می‌کند



```
1 function jam ()  
2 {  
3     trace(2+3);  
4 }  
5 jam();
```

شکل ۵۷-۱۵

خروجی کدهای فوق عدد ۵ خواهد بود
مثال ۲: تابعی که یک آرگومان ورودی عددی دارد و آن عدد به توان ۳ می‌رساند

```

1 function tavan3 (m:Number )
2 {
3     trace(m*m*m);
4 }
5 tavan3(2);

```

شکل ۱۵-۵۸

خروجی عدد ۸ خواهد بود
مثال ۳: تابعی که یک آرگومان ورودی رشته‌ای دارد به عبارت رشته کلمه Mr را اضافه می‌کند

```

1 function mr(s:String )
2 {
3     trace("Mr " + s);
4 }
5 mr("Ahmadi");

```

شکل ۱۵-۵۹

خروجی Mr Ahmadi خواهد بود
مثال ۴: تابعی که دو آرگومان ورودی عددی دارد عدد اول را در عدد دوم ضرب می‌کند

```

1 function zarb(a:Number ,b:Number )
2 {
3     trace(a*b);
4 }
5 zarb(2,3);

```

شکل ۱۵-۶۰

خروجی عدد ۶ می‌باشد.
مثال ۵: تابعی که دارای خروجی می‌باشد و دو آرگومان ورودی عددی نیز دارد (بسته تابعی)

```

1 var c:Number =0;
2 function zarb(a:int,b:int):Number
3 {
4     c=a*b;
5     return c;
6 }
7 trace(zarb(3,2));

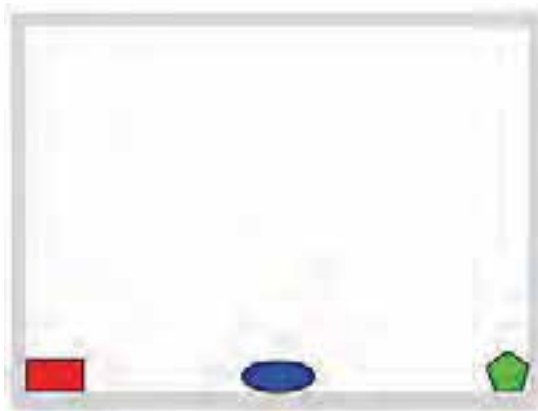
```

شکل ۱۵-۶۱

خروجی عدد ۶ می‌باشد.

تمرین ۸- استفاده از توابع تعریف شده توسط کاربر برای کنترل اشیا

۱. سند جدیدی به نام func.fla ایجاد نمائید.
۲. چهار لایه به نام‌های «مستطیل» - «بیضی» - «پنج ضلعی» و «اکشن» ایجاد نمائید.
۳. در لایه «مستطیل» یک مستطیل در گوشه پائین سمت چپ صحنه ترسیم نموده و آن را به یک Movie Clip با نام نمونه ms تبدیل نمائید.
۴. در لایه «بیضی» یک بیضی در وسط پائین صحنه ترسیم نموده و آن را به یک Movie Clip با نام نمونه bz تبدیل نمائید.
۵. در لایه «پنج ضلعی» یک پنج ضلعی در پائین سمت راست صحنه ترسیم نموده و آن را به یک Movie Clip با نام نمونه ps تبدیل نمائید. مطابق شکل زیر



شکل ۶۲-۱۵

۶. در لایه «اکشن» کدهای زیر را بنویسید

```

1 function moveMs():void
2 {
3     ms.y=50;
4     ms.rotation=45;
5 }
6 moveMs();
    
```

شکل ۶۳-۱۵

- توجه: ویژگی rotation برای چرخاندن اشیا به کار می‌رود. عدد مثبت چرخش در جهت عقربه‌های ساعت و عدد منفی چرخش در جهت عکس عقربه‌های ساعت.
۷. از Ctrl+Enter برای تست استفاده نمائید. نتیجه به صورت شکل زیر خواهد بود



شکل ۱۵-۶۴

۸. کدهای نوشته شده را به صورت زیر تغییر دهید

```

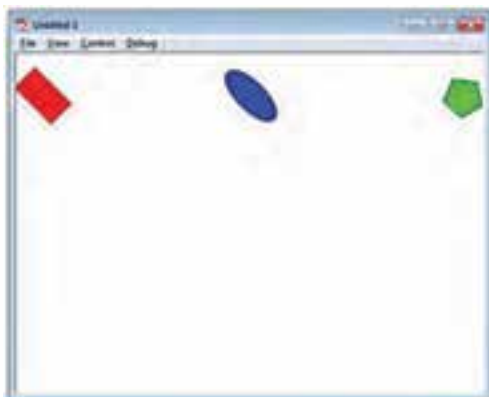
1 function moveMs(shp:MovieClip):void
2 {
3     shp.y=50;
4     shp.rotation=45;
5 }
6 moveMs(ms);
7 moveMs(bz);
8 moveMs(ps);
    
```



شکل ۱۵-۶۵

در تعریف تابع ما آرگومان ورودی را یک شیء از نوع Movie Clip در نظر گرفته‌ایم و از تابع moveMS سه بار استفاده کرده‌ایم

کدهای فوق باعث می‌شود هر سه شیء روی صحنه به نقطه ۵۰ عمودی انتقال یافته و ۴۵ درجه نیز بچرخند
 ۹. از Ctrl+Enter برای تست استفاده نمائید. نتیجه به صورت شکل زیر خواهد بود



شکل ۱۵-۶۶

حال اگر بدنه تابع که کدهای زیر می باشد

```
;shp.y=50
```

```
;shp.rotation=45
```

آنها را به صورت زیر تغییر دهیم (استفاده از عملگرهای انتساب = و +=)

```
;shp.y-=50
```

```
;shp.rotation+=45
```

۱۰. می خواهیم آرگومانهای ورودی علاوه بر شیء ویژگی های y و $rotation$ نیز باشد کدها را به زیر تغییر می دهیم یعنی محل و میزان چرخش را در زمان فراخوانی تابع تعیین نمائیم.

```

1 function moveMs(shp:MovieClip,movement:Number ,rotate:Number ):void
2 {
3     shp.y-=movement;
4     shp.rotation+=rotate;
5 }
6 moveMs(ms,150,45);
7 moveMs(bz,250,90);
8 moveMs(ps,50,180);
    
```

شکل ۶۷-۱۵

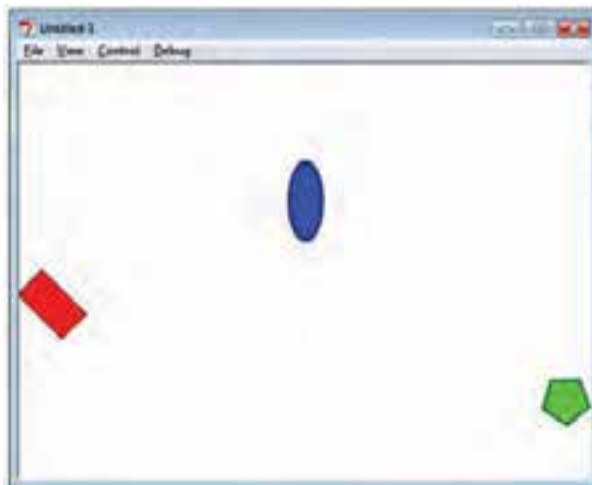
Shp: نام متغیر شیء

Movement: متغیر جایابی از نوع عددی

Rotate: متغیر چرخش از نوع عددی

۱۱. کارهای انجام شده را ذخیره نمائید (Ctrl+S)

پس از تست فیلم نتیجه به صورت زیر می باشد.



شکل ۶۸-۱۵

۲۰-۱۵ شناخت اصول قالب‌بندی اسکریپت‌ها

امکاناتی در Action Script برای قالب‌بندی کدها وجود دارد که به فهم راحت‌تر و خطایابی ساده‌تر آن کمک فراوان خواهد کرد. امکاناتی مانند: رنگ‌بندی کدها، قالب‌بندی اتومات، باز و بسته کردن کدها و ... می‌باشد. به‌عنوان مثال در رنگ‌بندی کدها (Syntax Color) به طور پیش فرض کلمات رزرو شده به رنگ آبی، رشته‌ها به رنگ سبز، متغیرها به رنگ سیاه و توضیحات به رنگ خاکستری می‌باشند

```

1 trace(12);
2 trace("iran");
3 trace(13);
4 trace("action");
5 trace(13+14); // output is 17
6 trace("13"+14); // output is 1314

```

شکل ۶۹-۱۵

اگرچه رنگ‌های کدنویسی را می‌توان تغییر داد، ولی بهتر است ترکیب رنگ بندی‌ها را تغییر ندهید



۲۱-۱۵ باز و بسته کردن کدها (Code Collapse)

چند خط کد نوشته شده را می‌توان به یک خط تبدیل کرد (Code Collapse) برای این منظور می‌توانید از روش‌های زیر استفاده کنید:
 الف) کدهایی که بین دو علامت آکولاد { } نوشته می‌شوند در حالتی که مکان نما بین { } باشد این کار انجام می‌گیرد.
 مثال: کد زیر را در Action Script وارد نمائید.

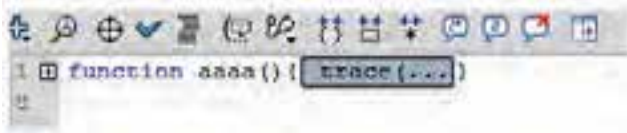
```

1 function aaaa() {
2   trace(12);
3   trace("iran");
4   trace(13);
5   trace("action");
6   trace(13+14);
7   trace("13"+14);
}

```

شکل ۷۰-۱۵

حال مکان نما را در خطوط ۲ تا ۶ قرار دهید و سپس بر روی آیکن بسته شدن بین آکولاد { } کلیک کنید نتیجه به صورت زیر خواهد بود.



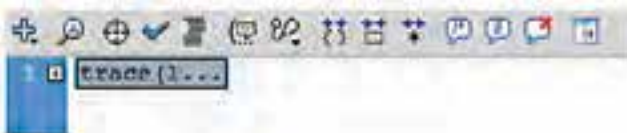
شکل ۱۵-۷۱

ب) کدهای انتخاب شده را هم می‌توان به یک خط تبدیل کرد (☐) برای باز کردن کدهای بسته شده (Collapse) از آیکون Expand All * استفاده نمائید. مثال: کدهای زیر را در Action Script وارد نمائید.

```
1 trace (12);
2 trace ("iran");
3 trace (13);
4 trace ("action");
5 trace (13+14);
6 trace ("13"+"14");
```

شکل ۱۵-۷۲

حال خطوط ۱ تا ۶ را انتخاب کنید و سپس بر روی آیکون بسته شدن کدهای انتخاب شده ☐ کلیک کنید نتیجه به صورت زیر خواهد بود.



شکل ۱۵-۷۳

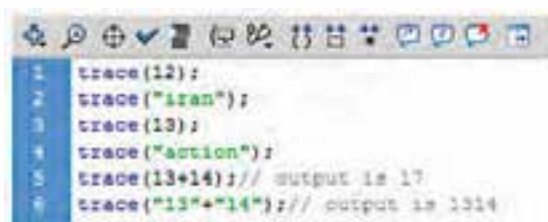
۱۵-۲۲ قالب‌بندی خودکار (Auto Format)

وقتی که شما کدهای برنامه را می‌نویسید و در انتهای خطوط علامت ; را قرار نمی‌دهید، با کلیک کردن بر روی آیکون قالب‌بندی خودکار (Auto Format) ☰ در Action Script در انتهای هر دستور علامت ; را بطور خودکار قرار می‌دهد. و تو رفتگی رعایت نشده را نیز اصلاح می‌کند و قالب نوشتن توابع و دستورات کنترلی را نیز اصلاح می‌کند به عنوان مثال کدهای زیر را در Action Script بدون علامت ; در انتهای خطوط وارد نمائید.

```
1 trace(12)
2 trace("iran")
3 trace(13)
4 trace("action")
5 trace(13+14) // output is 17
6 trace("13"+"14") // output is 1314
```

شکل ۱۵-۷۴

حالتی که در آن روی آیکون قالب‌بندی خودکار (Auto Format) کلیک کنید Action Script در انتهای هر دستور علامت؛ را بطور خودکار قرار خواهد داد.



```

1 trace(12);
2 trace("iran");
3 trace(13);
4 trace("action");
5 trace(13+14); // output is 17
6 trace("13"+"14"); // output is 1314

```

شکل ۱۵-۷۵

۱۵-۲۳ نقطه گذاری یا گرامر نقطه‌ای (dot syntax)

اشیاء در فلش می‌توانند درون یکدیگر نیز قرار بگیرند که برای مشخص کردن آنها از یکدیگر از علامت نقطه استفاده می‌شود (مانند پوشه‌ها و زیر پوشه‌ها در ویندوز که به وسیله \ از یکدیگر جدا می‌شوند) فرض کنید داخل کیفی (Bag)، جعبه مداد (pencilBox) و داخل جعبه، مداد پاک کن (Eraser) قرار دارد. بیرونی‌ترین کلیپ فیلم bag و داخل این کلیپ، کلیپ pencilBox قرار دارد و در داخل آن نیز کلیپ Eraser قرار دارد اگر هر کلیپ فیلم یک نام نمونه (Instance Name) به ترتیب bag، pencilBox و Eraser داشته باشد داخلی‌ترین کلیپ را می‌توان با استفاده از عبارت زیر، از خط زمان اصلی مورد استفاده قرار داد:

Bag.pencilBox.Eraser



شکل ۱۵-۷۶

اگر Eraser دارای تعدادی فریم در خط زمانش باشد با استفاده از گرامر نقطه‌ای می‌توانید به متد gotoAndPlay() کلاس MovieClip دسترسی داشته باشید:

Bag.pencilBox.Eraser.gotoAndPlay(5);

اگر اکشن شما در نمونه فیلم Eraser قرار داشته باشد برای ارجاع به آن می‌توانید از کلمه رزرو شده this استفاده نمایید و به جای کد بالا از کد زیر استفاده نمایید

this.gotoAndPlay(5);

this همان Time Line جاری است.

در کد زیر خروجی pencilBox خواهد بود چون pencilBox یک سطح بالاتر از Eraser قرار دارد. و ویژگی name نام شی را برمی‌گرداند

trace(bag.pencilBox.Eraser.parent.name);

مثال: کدهای زیر را برای یک فریم کلیدی فایل جدید بنویسید

trace(this);

trace(this.parent);

توانایی به کارگیری Action Script


خروجی کدهای بالا در پانل output به صورت زیر خواهد بود

[object MainTimeline]

[object Stage]

با این توضیحات می‌توان گفت نقطه در برنامه‌نویسی Action Script می‌تواند در موارد زیر مورد استفاده قرار گیرد:

- نقطه به عنوان جداکننده شیء (Object) از ویژگی (Property)
- نقطه به عنوان جداکننده شیء (Object) از متد (Method)
- نقطه به عنوان جداکننده شیء (Object) از رویداد (Event)
- نقطه به عنوان جداکننده شیء (Object) از شیء (Object)

در زمان کدنویسی برای دسترسی به اشیاء به همراه مسیرشان می‌توان از آیکون  Insert target Path استفاده نمود



شکل ۱۵-۲۷

وقتی ما متغیری را با استفاده دستور var تعریف می‌کنیم سپس از نام متغیر در خطهای بعدی استفاده نمائیم با تایپ نقطه منوی Code Hint (لیست متدها و ویژگی‌ها متناسب با نوع داده‌ای) ظاهر می‌گردد. شکل زیر مربوط به متغیر عددی است



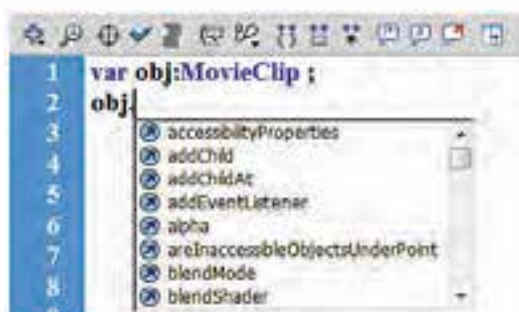
شکل ۱۵-۲۸

اگر نوع داده‌ای را رشته‌ای در نظر بگیرد Code Hint مربوط به متغیر رشته ظاهر می‌گردد.



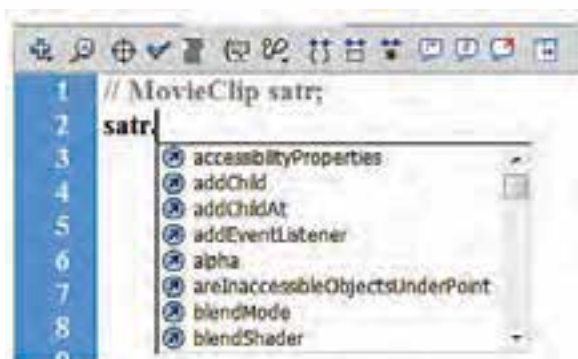
شکل ۱۵-۷۹

اگر متغیر شی از نوع MovieClip تعریف کنید Code Hint مربوط به کلاس Movie Clip بعد از تایپ نقطه ظاهر می‌گردد



شکل ۱۵-۸۰

نکته: اگر نوع متغیر (شی) را در توضیحات (Comment) نیز مشخص کنید بعد از تایپ نام شی و قرار دادن نقطه در Code Hint در جلوی آن نمایش داده می‌شود.



شکل ۱۵-۸۱

توجه: توضیحات زیر نیز باعث فعال شدن Code Hint می‌شود

```
// String s;
// Number n;
// int i;
// Button btn;
```

۱۵-۲۴ شناخت ساختار جملات شرطی

همانطور که می‌دانید دستورات شرطی دستوراتی هستند که روند اجرای برنامه را مشروط به اجرای شرطی می‌نمایند به طوری که در نتیجه درست یا غلط بودن شرط مورد نظر می‌تواند روند اجرایی برنامه تغییر نماید.

۱۵-۲۵ جملات (دستورات) شرطی (conditionals):

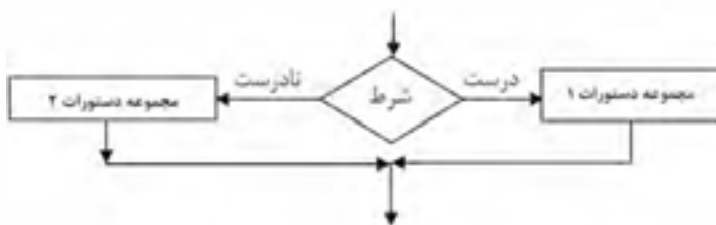
همانطور که گفتیم دستورات شرطی به توجه به این که روند اجرایی برنامه را مشروط به اجرایی شرطی می‌نمایند خواهند توانست به برنامه قدرت تصمیم‌گیری داده و شرایط مختلفی را در یک برنامه مورد بررسی قرار دهند.

در Action Script مانند اکثر زبان‌های برنامه‌نویسی Multimedia از دو دستور شرطی می‌توان استفاده کرد:
switch... case -۲ if -۱

۱۵-۲۶ دستور شرطی if ... else

ساده‌ترین دستور، برای تصمیم‌گیری در Action Script if می‌باشد و ساختار کلی آن به صورت زیر می‌باشد

```
if (عبارت شرطی)
{
    مجموعه دستورات ۱
}
else
{
    مجموعه دستورات ۲
}
```



اگر عبارت شرطی جلوی if درست باشد «مجموعه دستورات ۱» اجرا می‌شوند در غیر این صورت «مجموعه دستورات ۲» اجرا خواهند شد.

نکته: در شکل کلی دستور If بخش Else اختیاری است.

مثال ۱:



شکل ۱۵-۸۲

- تمرین ۹: توقف اجرای فیلم پس از سه بار اجرا
۱. فایل مربوط به تمرین ۷ را باز کنید (num fla)
 ۲. فایل را با نام numif fla ذخیره نمایید.
 ۳. کدهای فریم ۳۰ را به صورت زیر تغییر دهید.

```

1 i=i+1;
2 if (i<3)
3 {
4     gotoAndPlay(2);
5 }
6 else
7 {
8     stop();
9 }

```

شکل ۱۳-۱۵

تشریح کدهای نوشته شده

در فریم ۱ مقدار i با مقدار اولیه صفر تعریف شده است
 در فریم ۳۰ ابتدا به مقدار i یک واحد اضافه می شود (خط ۱)
 در خط ۲ مقدار i بررسی می شود چنانچه مقدار i کمتر ۳ بود دستور خط ۴ اجرا شده و باعث می شود
 هدپخش به فریم شماره ۲ انتقال یابد و چنانچه مقدار i از ۳ بزرگتر شد (یا مساوی با ۳ شد) دستور خط ۸ اجرا
 می گردد و باعث توقف فیلم می گردد

نکته: بجای $i=i+1$ می توان از $i++$ یا از $i+=1$ نیز استفاده نمود.

اگر نیاز دارید چندین شرط با هم بررسی شود در این حالت می توانید از فرم `if` تودرتو به صورت زیر استفاده کنید

`if` (عبارت شرطی)

{

مجموعه دستورات ۱

{

`else if` (عبارت شرطی ۲)

{

مجموعه دستورات ۲

}

`else if` (عبارت شرطی ۳)

}

.

.

نکته: از انجایی که دستورات If تودرتو ساختار برنامه و خوانایی آن را دچار اشکال می‌نماید، بهتر است بجای استفاده از این دستور، از دستور switch ...case که در ادامه در مورد آن صحبت خواهیم کرد در برنامه‌نویسی استفاده نمایید.

۱۵-۲۶ switch ...case

در صورتیکه بخواهیم شرط به ازای مقادیر مختلف مورد بررسی قرار گرفته و گروهی از مجموعه دستورات اجرا شوند از دستور شرطی switch ...case استفاده می‌شود که فرم کلی آن به صورت زیر می‌باشد

```
switch (نام متغیر)
{
case مقدار ۱ :
    مجموعه دستورات ۱
    break;
case مقدار ۲ :
    مجموعه دستورات ۲
    break;
case مقدار ۳ :
    مجموعه دستورات ۳
    break;
.
.
.
default:
    مجموعه دستورات
    break;
}
```

اگر متغیر جلوی switch با هر کدام از مقادیر جلوی case برابر باشد مجموعه دستورات مربوط به همان case اجرا شده و فرمان switch خاتمه می‌یابد. اگر متغیر جلوی Switch با هیچکدام از مقادیر جلوی case برابر نباشد دستورات default اجرا می‌شود.

نکته: اگر دو Case دارای مقدار یکسانی باشند دستورات اولین case اجرا می‌شود و فرمان switch خاتمه می‌یابد

کلمه رزرو شده break اغلب برای بستن یک فرمان حلقه در حال اجرا مورد استفاده قرار می‌گیرد. که در بخش بعدی به بررسی کامل آن خواهیم پرداخت.

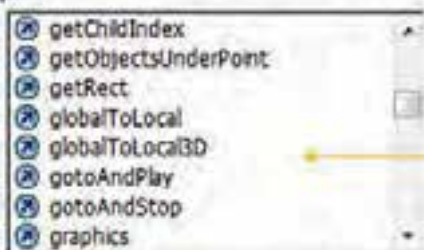
۲۷-۱۵ شناخت اصول استفاده از کلاسهای Flash

اگر بخواهیم یک نمونه از کلاس ایجاد کنیم از دستور `var` به شکل زیر استفاده می‌کنیم
`var` () نام کلاس = `new` نام کلاس: نام کلیپ

یکی از کلاس‌های فلش `MovieClip` می‌باشد

```
var clip1: MovieClip = new MovieClip()
```

```
clip1.
```



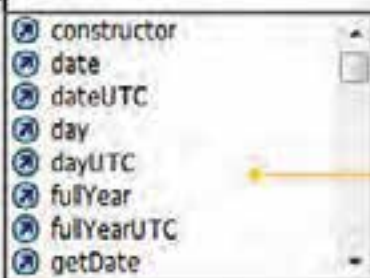
اجزای کلاس `MovieClip`

شکل ۱۴-۱۵

یکی از کلاس‌های فلش `Date` می‌باشد

```
var myBirthday: Date = new Date();
```

```
myBirthday.
```



اجزای کلاس `Date`

شکل ۱۵-۱۵

تمرین ۱۰: برنامه‌ای بنویسید که نام روز هفته را در پانل Output نمایش دهد.
 توجه:

- تعریف شیئی `someDate` از کلاس `date`
- تعریف متغیر `dayNum` از نوع عددی بدون علامت با مقدارهی عدد روز جاری (`getDate` عدد روز جاری هفته را برمی‌گرداند)

```

1  var someDate:Date = new Date();
2  var dayNum:uint = someDate.getDay();
3  switch(dayNum)
4  {
5      case 0:
6          trace("Sunday");
7          break;
8      case 1:
9          trace("Monday");
10         break;
11     case 2:
12         trace("Tuesday");
13         break;
14     case 3:
15         trace("Wednesday");
16         break;
17     case 4:
18         trace("Thursday");
19         break;
20     case 5:
21         trace("Friday");
22         break;
23     case 6:
24         trace("Saturday");
25         break;
26     default:
27         trace("Out of range");
28         break;
29 }

```

شکل ۱۵-۱۶

بعد از تشریح مبحث رویدادها، تمرین‌های کاربردی برای if و Switch مورد توجه قرار خواهد گرفت.

۱۵-۲۸ رویدادها Events

در Action Script ۳ برای اینکه بفهمیم در طول اجرای برنامه (پخش فیلم) چه رویدادی رخ داده یا ممکن است رخ دهد از مفهومی تحت عنوان Event Listener (گوش دهنده رویداد) استفاده می‌نمائیم برای ایجاد

یک Event Listener به متدی به نام addEventListener() نیاز داریم

فرم کلی addEventListener

; (متد پاسخدهی به رویداد , نام رویداد:: منبع رویداد) addEventListener . نام شیء

منبع رویداد: می‌توانند صفحه کلید (KeyboardEvent) - ماوس (MouseEvent) - تایمر سیستم

(TimerEvent) - سایر رویدادها (Event) می‌باشد

نام رویداد: متناسب با منبع رویداد می‌باشد تمام رویدادها با حروف بزرگ نمایش داده می‌شوند.

رویدادهای ماوس

MOUSE_DOWN , MOUSE_UP , MOUSE_MOVE , MOUSE_OVER , CLICK ,

DOUBLE_CLICK , MOUSE_WHEEL, ROLL_OUT, ROLL_OVER

رویدادهای صفحه کلید

KEY_DOWN , KEY_UP

رویدادهای تایمر سیستم:

TIMER , TIMER_COMPLETE

رویداد خط زمان:

ENTER_FRAME

متد پاسخدهی به رویداد: در واقع تابعی است که مشخص می‌کند بازای رخ دادن رویدادی، اشیاء

Action Script چه کاری باید انجام بدهند (چه دستوراتی باید اجرا شوند). که با استفاده از function به

صورت زیر تعریف می‌شوند.

void : (منبع رویداد: شیء رویداد) متد پاسخدهی به رویداد function

{

کدهایی که باید در ازای رخ دادن رویداد اجرا شوند

}

تمرین ۱۱- شروع اجرای فیلم با کلیک کردن بر روی دکمه شروع (کنترل خط زمان)

۱. فایل stop fla را که مربوط به تمرین ۱ می‌باشد باز کرده و با نام btnStart fla ذخیره نمائید.

۲. یک لایه جدید به نام «شروع» ایجاد نموده و در گوشه پائین سمت چپ صحنه متن با محتوی Start با

فونت ۲۰ نوشته و آن را به نماد Button با نام «دکمه شروع» تبدیل نمائید

۳. نام نمونه متن را btnSt قرار دهید



شکل ۱۵-۱۷

۴. فریم ۳۰ لایه «اکشن» را انتخاب کرده و کد نوشته شده را پاک کنید. و در فریم ۱ کدهای زیر را بنویسید

۱۵

```

1 stop();
2 btnSt.addEventListener(MouseEvent.CLICK ,playMovie);
3 function playMovie (evt:MouseEvent) :void
4 {
5     play();
6 }
    
```

شکل ۱۵-۱۸

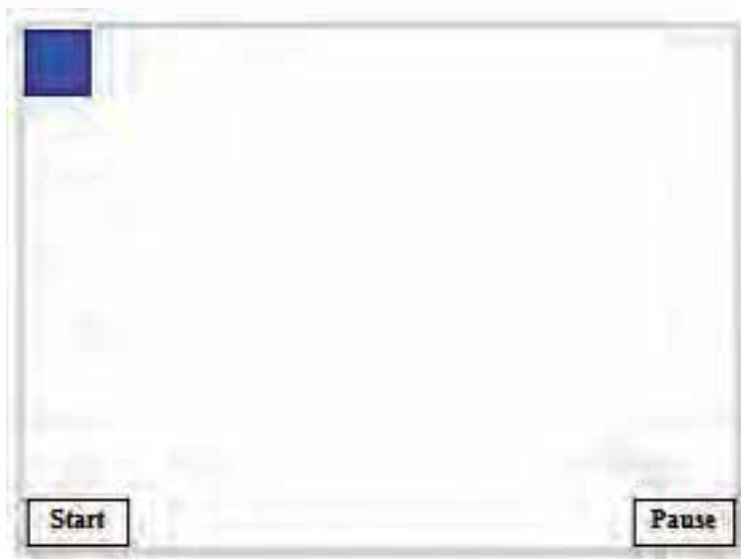
در کدهای بالا در خط دوم با استفاده از متد `addEventListener` مشخص کرده‌ایم در صورت کلیک کردن بر روی دکمه `Start` (که نام نمونه آن `btnSt` است) تابع `playMovie` فراخوانی شود (در بدنه تابع `playMovie` متد `play()` برای اجرای فیلم نوشته شده است)

۵. فیلم را تست کنید شما هیچ حرکتی بر روی صحنه ندارید. اگر بر روی `Start` کلیک کنید ملاحظه خواهید کرد چهار ضلعی شروع به حرکت کرده و در فریم ۱ متوقف خواهد شد. برای ادامه دار بودن حرکت مربع، کد زیر را در فریم ۳۰ لایه `Action Script` بنویسید

`gotoAndPlay (2);`

۶. حال می‌خواهیم دکمه‌ای به عنوان توقف قرار دهیم که نمایش فیلم ادامه دار بوده و با کلیک بر روی دکمه توقف پخش فیلم متوقف شود. یک لایه جدید به نام «توقف» ایجاد نموده و در گوشه پائین سمت راست صحنه متن با محتوی `Pause` با فونت ۲۰ نوشته و آن را به نماد `Button` با نام «دکمه توقف» تبدیل نمائید

۷. نام نمونه دکمه توقف را btnPs قرار دهید



شکل ۱۵-۱۹

۸. در فریم ۱ کدهای زیر را اضافه کنید

```

7 btnPs.addEventListener(MouseEvent.CLICK, pauseMovie);
8 function pauseMovie (evt:MouseEvent) :void
9 {
10     stop();
11 }

```

شکل ۱۵-۹۰

۹. فیلم را تست کنید با کلیک بر روی دکمه Start مربع شروع به حرکت کرده و هر لحظه بر روی Pause کلیک کنید حرکت مربع متوقف می‌شود.

۱۰. کارهای انجام شده را با استفاده از Ctrl+S ذخیره نمایید.

در تمرین فوق پخش و توقف فیلم در اختیار کاربر می‌باشد. ضمناً می‌توانید بجای رویداد CLICK از رویدادهای MOUSE_DOWN و MOUSE_UP نیز استفاده نمایید.

تمرین ۱۲- می‌خواهیم با کلیک بر روی صحنه یکی شی هم جابجا شود و هم بچرخد و هم محو شود

۱. سند جدیدی به نام stageMouse fla ایجاد نمایید

۲. دو لایه به نام‌های «اکشن» و «پنج ضلعی» ایجاد نمایید.

۳. یک پنج ضلعی ترسیم نموده و پس از تبدیل به نماد Movie Clip نام نمونه آن را pg قرار دهید

۴. فریم یک لایه «اکشن» را انتخاب نموده و با زدن کلید F9 وارد محیط Action Script شوید و کدهای

زیر را درج نمایید.

```

1 stage.addEventListener(MouseEvent.CLICK ,movePg);
2 function movePg (evt:MouseEvent ):void
3 {
4     pg.x+=30;
5     pg.rotation+=45;
6     pg.alpha-=0.1;
7 }
    
```

شکل ۹۱-۱۵

۵. فیلم را تست کنید با کلیک بر روی صحنه، پنج ضلعی ۳۰ واحد به سمت راست حرکت کرده و ۴۵ درجه می چرخد و رنگ آن نیز یک دهم شفاف می شود
 کد خط ۴ باعث می شود تا پنج ضلعی ۳۰ واحد به سمت راست حرکت کند
 کد خط ۵ باعث می شود تا پنج ضلعی ۴۵ درجه در جهت عقربه های ساعت بچرخد
 کد خط ۶ باعث می شود تا رنگ پنج ضلعی تا یک دهم شفاف شود.

نکته: اگر بجای stage در خط یک از pg استفاده کنید با هر بار کلیک بر روی پنج ضلعی، پنج ضلعی ۳۰ واحد به سمت راست حرکت کرده و ۴۵ درجه می چرخد و رنگ آن نیز یک دهم شفاف می شود

۱۵

تمرین ۱۳- کنترل اشیاء روی صحنه با استفاده از رویدادهای صفحه کلید

۱. فایل تمرین ۱۳ را با نام StageKeyboard.fla را ذخیره کنید.
۲. کدهای فریم ۱ لایه اکشن را پاک کرده و کدهای زیر را وارد نمایید.

```

1 stage.addEventListener(KeyboardEvent.KEY_DOWN ,movePg);
2 function movePg (evt:KeyboardEvent ):void
3 {
4     pg.x+=20;
5     pg.rotation+=20;
6 }
    
```

شکل ۹۲-۱۵

۳. فیلم را تست کنید با فشردن هر کدام از کلیدهای جهتی «پنج ضلعی» ۲۰ واحد به سمت راست کرده و ۲۰ درجه نیز می چرخد و کلید را در همان حالت نگه دارید این حرکت و چرخش ادامه می یابد.
۴. حال می خواهیم فقط با فشردن کلید جهتی راست، پنج ضلعی به سمت راست حرکت کند از دستور if به صورت زیر استفاده می کنیم

```

1 stage.addEventListener(KeyboardEvent.KEY_DOWN ,movePg);
2 function movePg (evt:KeyboardEvent ):void
3 {
4     if(evt.keyCode==Keyboard.RIGHT )
5     {
6         pg.x+=20;
7         pg.rotation+=20;
8     }
9 }

```

شکل ۹۳-۱۵

نکته ۱: وقتی کلیدی روی صفحه کلید فشرده می‌شود با استفاده از ویژگی `keyCode` می‌توان کد کلید را بدست آورد.

نکته ۲: بجای `Keyboard.RIGHT` می‌توان کد کلید جهتی راست که عدد ۳۹ هست را قرار داد.

اگر بخواهیم با فشردن کلد جهتی چپ نیز پنج ضلعی به سمت چپ حرکت کند بعد از `if` بعد از خط هشتم کدهای زیر را اضافه نمائید.

```

else if (evt.keyCode==Keyboard.LEFT)
}

```

```

pg.x-=20;
pg.rotation-=20;

```

```

{

```

چنانچه بخواهید پنج ضلعی به سمت بالا و پائین حرکت کند از ویژگی `y` استفاده نمائید و از کلیدهای `UP` و `DOWN` استفاده نمائید.

نکته مهم: اگر بخواهید از کلیدهای دیگر استفاده کنید باید بعد از این که فیلم را تست کردید از منوی `Control` گزینه `Disable keyboard shortcut` را فعال کنید

تمرین ۱۴- کنترل اشیاء با استفاده از رویداد خط زمان (`ENTER_FRAME`)

۱. فایل تمرین ۱۲ (`stageMouse.fla`) را باز کرده و با نام جدید `EnterFrame.fla` ذخیره نمائید.

۲. فریم ۱ لایه «اکشن» را انتخاب کرده و با استفاده کلید `F9` پنجره `Action Script` را فعال نمائید

۳. کدهای موجود را به صورت زیر تغییر دهید

```

1 stage.addEventListener(Event.ENTER_FRAME ,movePg);
2 function movePg (evt:Event):void
3 {
4     pg.x+=10;
5     pg.rotation+=10;
6 }
    
```

شکل ۹۴-۱۵

۴. تعداد fps را به ۱۰ تغییر دهید (در Time Line)
۵. فیلم خود را تست کنید خواهید دید بدون اینکه رویدادی از طرف شما به وقوع بپیوندد پنج ضلعی ضمن چرخش، به سمت راست حرکت می کند. در اینجا ما از رویداد ENTER_FRAME استفاده کرده ایم. بازای تعداد فریم‌ها در ثانیه پنج ضلعی شروع به حرکت می کند.

تمرین ۱۵: تایمر عقربه‌ای ۶۰ ثانیه‌ای با استفاده از یک فریم

۱. یک سند جدید ایجاد نموده و با نام ClockExample.fla ذخیره نمایید.
۲. سه لایه به نام‌های «عقربه»، «پس زمینه» و «کشن» ایجاد نمایید
۳. در لایه «پس زمینه» با استفاده از ابزار بیضی از بخش Oval Tool یک دایره به شعاع ۱۰۰ (پهنا و ارتفاع ۲۰۰ در ۲۰۰) رسم نموده و آن را به یک نماد (سمبل) به نام پس زمینه از نوع Graphic تبدیل کنید تا به Library اضافه شود (از منوی Modify گزینه Convert to Symbol ... انتخاب کنید یا از کلید F۸ استفاده نمایید)
۴. در لایه «عقربه» با استفاده از ابزار خط Line Tool یک خط عمودی به ارتفاع ۹۵ به مرکز دایره رسم نموده و آن را به یک نماد (سمبل) به نام عقربه از نوع Movie Clip تبدیل کنید تا به Library اضافه شو



شکل ۹۵-۱۵

۶ لایه «اکشن» را انتخاب نموده و برای فعال شدن پنجره کدنویسی کلید F9 را فشار دهید و کد زیر را بنویسید

```

1 var clockTimer:Timer = new Timer(1000,60);
2 // تعیین رویداد برای دکمه شروع
3 stage.addEventListener(MouseEvent.CLICK,startTimer);
4 // Add event listener for timer
5 clockTimer.addEventListener(TimerEvent.TIMER, moveHand);
6 clockTimer.addEventListener(TimerEvent.TIMER_COMPLETE, endTimer);
7 // تابع شروع تایمر
8 function startTimer(e:MouseEvent):void
9 {
10     clockTimer.start();
11     trace("شروع");
12 }
13 // تابع حرکت دادن عقربه
14 function moveHand(e:TimerEvent):void
15 {
16     secondHand.rotation = secondHand.rotation + 6;
17 }
18 // تابع خاتمه تایمر
19 function endTimer(e:TimerEvent):void
20 {
21     trace("پایان");

```

شکل ۹۶-۱۵

تشریح کدهای نوشته شده:

الف- در تعریف متغیر (شئی clockTimer)، آرگومان اول Timer زمان بر حسب میلی ثانیه می باشد و آرگومان دوم تعداد دفعات فراخوانی متد addEventListener می باشد آرگومان اول که ۱۰۰۰ قرار داده شده یعنی زمان تاخیر ۱ ثانیه در نظر گرفته شده است. اگر آرگومان دوم مشخص نشود تعداد فراخوانی نامحدود خواهد بود (آرگومان دوم اختیاری است)

```
var clockTimer:Timer = new Timer(1000,60);
```

ب- اگر در Timer از آرگومان دوم استفاده شود برای پایان Timer باید از رویداد TimerEvent.TIMER_COMPLETE به صورت زیر استفاده شود

```
clockTimer.addEventListener(TimerEvent.TIMER_COMPLETE, endTimer);
```

ج- از ویژگی rotation برای چرخاندن یک Movie Clip حول یک نقطه استفاده می شود. چون قرار است عقربه روی محیط دایره ۶۰ بار بچرخد پس باید ۳۶۰ را بر ۶۰ تقسیم نمود که عدد ۶ حاصل می شود

```
;secondHand.rotation = secondHand.rotation + 6
```

د- در این فیلم با کلیک کردن بر روی صحنه عقربه شروع حرکت می کند.

۲۹-۱۵ متن در Action Script

ما تا بحال تمام خروجی توابع و دستوراتی مانند trace را در پنجره output مورد مشاهده قرار می‌دادیم اکنون می‌خواهیم این خروجی‌ها را در پنجره swf ببینیم برای این کار از شیء text استفاده می‌نمائیم.

۱. تمرین ۱۶- می‌خواهیم وقتی پخش فیلم متوقف می‌شود عبارت «توقف» بر روی صحنه نمایش داده شود. فایل stop.fla مربوط به تمرین ۱ را باز نموده و با نام text.fla ذخیره نمائید.
۲. یک لایه جدید به نام «متن» ایجاد نموده و در فریم ۱ آن یک کادر متن بدون محتوی ترسیم نموده و نوع آن را Dynamic Text قرار دهید و نام نمونه آن را نیز txt1 قرار دهید.
۳. فریم ۳۰ لایه «اکشن» را انتخاب نموده و با استفاده از کلید F9 وارد محیط Action Script شده و کد زیر را اضافه نمائید.

`txt1 . text = «توقف» ;`

۴. حال فیلم خود را تست کنید به متوقف شدن فیلم در فریم ۳۰ عبارت «توقف» در کادر متن txt1 بر روی صحنه نمایش داده می‌شود.

تمرین ۱۷- نمایش محتوی متغیر عددی در Text

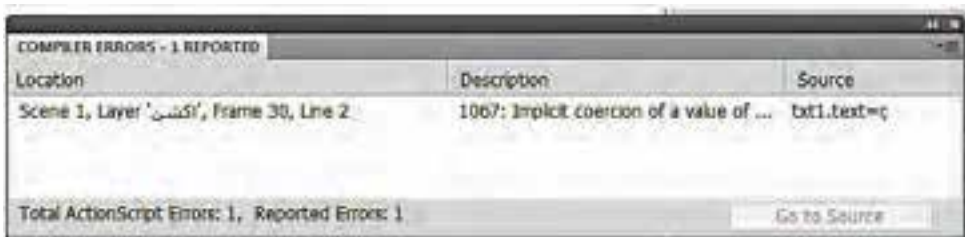
۱. فایل num.fla مربوط به تمرین ۱ را باز نموده و با نام txtnum.fla ذخیره نمائید.
۲. یک لایه جدید به نام «متن» ایجاد نموده و در فریم ۱ آن یک کادر متن بدون محتوی ترسیم نموده و نوع آن را Dynamic Text قرار دهید و نام نمونه آن را نیز txt1 قرار دهید.
۳. فریم ۳۰ لایه «اکشن» را انتخاب نموده و با استفاده از کلید F9 وارد محیط Action Script شده و کدهای نوشته شده را به صورت زیر تغییر دهید

```

1 i=i+1;
2 txt1.text=i;
3 gotoAndPlay(2);
    
```

شکل ۹۷-۱۵

۴. حال فیلم خود را تست کنید پیغام خطای زیر در پنجره Compiler Errors نمایش داده می‌شود.



شکل ۹۸-۱۵

نکته: فقط عبارت متنی را می‌توان به ویژگی text شیء txt1 انتساب داد و عدد را مستقیماً نمی‌توان به آن انتساب داد برای حل این مشکل از تابع String (که برای تبدیل عدد به رشته می‌باشد) استفاده نمائید.

کد `txt1.text=i;` را به صورت زیر تغییر دهید.

```
1 txt1.text=String(i);
```

شکل ۹۹-۱۵

حال فیلم خود را تست کنید بازای نمایش با راول فیلم عدد ۱ و بار دوم عدد ۲ و بار سوم عدد ۳ و الی آخر بر روی شئی `txt1` نمایش داده می شود.

تمرین ۱۸- دریافت دو عدد و محاسبه حاصلضرب آنها

۱. یک سند جدید ایجاد نموده و با نام `zarb.fla` ذخیره نمایید.

۲. دو لایه به نام های «متن» و «اکشن» ایجاد نمایید

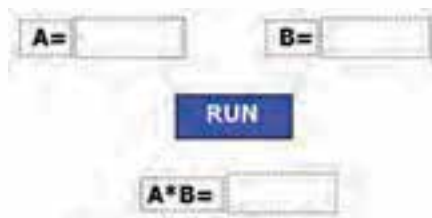
۳. در لایه «متن» با استفاده از ابزار متن شکل زیر را ایجاد نمایید.

`A=` و `B=` متن از نوع Static Text می باشد . جلوی `A=` متن از نوع Input Text با نام نمونه

`txtA` - جلوی `B=` متن از نوع Input Text با نام نمونه `txtB` و جلوی `A*B=` متن از نوع Dynamic Text

با نام نمونه `txtZ` و `txtZ` هم ترکیبی از مستطیل و متن که به نماد Button تبدیل شده و نام نمونه آن نیز

`btnZarb` می باشد



شکل ۱۰۰-۱۵

۴. فریم ۱ لایه «اکشن» را انتخاب نموده و با استفاده از کلید `F9` وارد محیط Action Script شود و کدهای

زیر را بنویسید

```

1 // Button btnZarb;
2 var a:Number=0;
3 var b:Number=0;
4 var c:Number=0;
5 btnZarb.addEventListener(MouseEvent.CLICK,zarb);
6 function zarb(ev:MouseEvent):void
7 {
8     a=Number(txtA.text);
9     b=Number(txtB.text);
10    c=a*b;
11    txtZ.text=String(c);
12 }

```

شکل ۱۰۱-۱۵

توانایی به کارگیری Action Script

۵. فیلم را تست نمایید و در جلوی A= عدد ۵ و در جلوی B= عدد ۳ را وارد نموده و برای نمایش نتیجه محاسبه بر روی دکمه RUN کلیک نمایید.

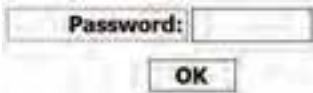


نکته: با توجه به اینکه متغیرهای a و b عددی هستند و محتوی txtA و txtB رشته‌ای هستند با استفاده از تابع Number آنها را به عدد تبدیل نمائید. به دستورهایی ردیف ۸ و ۹ دقت نمائید.

شکل ۱۰۲-۱۵

تمرین ۱۹- کادر دریافت رمز و بررسی رمز وارد شده (کنترل رمز عبور توسط Action Script)

۱. فایل btnStart.fla مربوط به تمرین ۱۱ را باز نموده و با نام txtPass.fla ذخیره نمائید.
۲. یک لایه جدید به نام «رمز» ایجاد نموده و آیتم‌های زیر را در فریم ۱ آن ایجاد نمائید. (شکل ۹۷-۱۵)
۳. کادر متن Password از نوع Dynamic Text با نام نمونه txtP می‌باشد. کادر متن جلوی Password از نوع Input Text با نام نمونه txtPass می‌باشد (ضمناً behavior آن را Password قرار دهید). کادر متن OK را به نماد Button تبدیل نموده و نام نمونه آن نیز btnOK قرار دهید.
۴. فریم ۱ لایه «اکشن» را انتخاب نموده و با استفاده از کلید F9 وارد محیط Action Script شده و کدها به صورت زیر را تغییر دهید



شکل ۱۰۳-۱۵

```

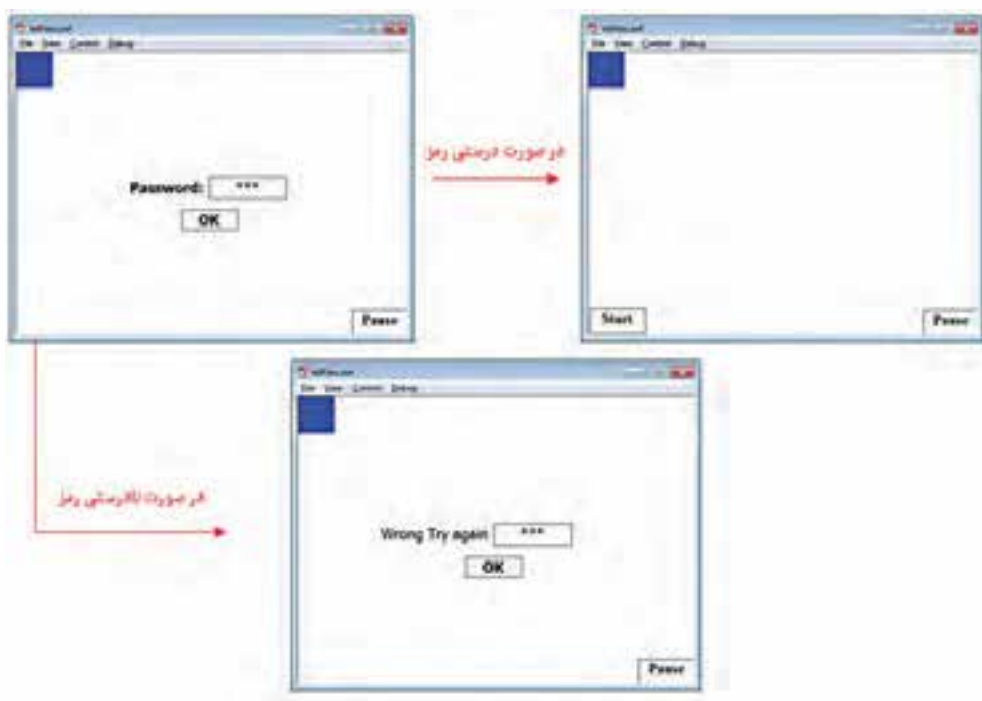
1 stop();
2 btnSt.visible=false;
3 btnSt.addEventListener(MouseEvent.CLICK,playMovie);
4 function playMovie (evt:MouseEvent) :void
5 {
6     play();
7 }
8 btnPs.addEventListener(MouseEvent.CLICK, pauseMovie);
9 function pauseMovie (evt:MouseEvent) :void
10 {
11     stop();
12 }
13 btnOK.addEventListener(MouseEvent.CLICK,okPass);
14 function okPass (evt:MouseEvent )
15 {
16     if (txtPass.text=="ali")
17     {
18         txtP.visible=false;txtPass.visible=false;btnOK.visible=false;
19         btnSt.visible=true;
20     }
21     else
22     {
23         txtP.text="Wrong Try again"
24     }
25 }

```

شکل ۱۰۴-۱۵

۱۵-۳۰ کدهای جدید

خط ۲: `btnSt.visible=false;` برای مخفی کردن دکمه Start در ابتدای نمایش فیلم می‌باشد. خطوط ۱۳ تا ۲۵: وقتی شما بر روی دکمه OK کلیک می‌کنید رمز وارد شده بررسی می‌شود در صورتی که برابر عبارت ali باشد دکمه Start نمایش داده شده و کادرهای متن به‌مراه دکمه OK مخفی می‌شوند اگر رمز وارد شده نادرست باشد متن Password به Wrong Try again تبدیل می‌گردد. ۵. فیلم را تست کنید در ابتدا دکمه Satrt مخفی می‌باشد پس از وارد کردن رمز (که به صورت ستاره نمایش داده می‌شود) بر روی دکمه OK کلیک نمائید.



شکل ۱۵-۱۰۵

۱۵-۳۱ مدیریت رنگ و سایز فونت متن با استفاده از Action Script

برای کنترل ویژگی متن باید یک شیء از نوع کلاس `TextFormat` تعریف نمائیم و برای اعمال تغییرات روی متن باید از متد `SetTextFormat` استفاده نمائید.

برای تعیین رنگ متن در کدنویسی از ویژگی `color` با قالب زیر استفاده می‌شود
`color=0xPRGGBB` . شئی متن

بجای RR می‌توان از 00 تا FF استفاده نمود که تعیین کننده مقدار رنگ قرمز می‌باشد
 بجای GG می‌توان از 00 تا FF استفاده نمود که تعیین کننده مقدار رنگ سبز می‌باشد
 بجای BB می‌توان از 00 تا FF استفاده نمود که تعیین کننده مقدار رنگ آبی می‌باشد
 هر چقدر عدد رنگ به سمت FF نزدیک شود رنگ روشن تر و به سمت 00 عدد را تغییر دهید رنگ تیره تر خواهد شد.
 اگر اعداد RR، GG و BB مثل هم باشد طیف رنگ بین سیاه و سفید تغییر خواهد نمود.

جدول رنگ ها

مقدار	رنگ
0xFF0000	قرمز
0x00FF00	سبز
0x0000FF	آبی
0x000000	سیاه
0xFFFFFFFF	سفید
0xFFFFF0	زرد
0xFF00FF	صورتی
0x00FFFF	فیروزه ای
0x000066	آبی تیره

- تمرین ۲۰- می‌خواهیم رنگ و سایز متن را با استفاده از رویدادهای ماوس تغییر دهیم
۱. یک سند جدید ایجاد نموده و با نام txtColor fla ذخیره نمائید.
 ۲. دو لایه به نام‌های «متن» و «اکشن» ایجاد نمائید
 ۳. در لایه «متن» با استفاده از ابزار متن عبارت Flash CS4 با سایز ۵۰ از نوع Dynamic Text و با نام نمونه txt ایجاد نمائید.
 ۴. در فریم یک لایه «اکشن» با استفاده از کلید F9 وارد پنجره Action Script شوید. و کدهای زیر را اضافه نمائید.

```

1 var tf:TextFormat=new TextFormat();
2 txt.addEventListener(MouseEvent.CLICK,redC);
3 txt.addEventListener(MouseEvent.CLICK,greenC);
4 txt.addEventListener(MouseEvent.CLICK,blueC);
5 txt.addEventListener(MouseEvent.CLICK,yellowC);
6 function greenC (eve:MouseEvent ): void
7 {
8     tf.color =0x00FF00;
9     txt.setTextFormat(tf);
10 }
11 function redC (eve:MouseEvent ): void
12 {
13     tf.color =0xFF0000;
14     txt.setTextFormat(tf);
15 }
16 function blueC (eve:MouseEvent ): void
17 {
18     tf.color =0x0000FF;
19     txt.setTextFormat(tf);
20 }
21 function yellowC (eve:MouseEvent ): void
22 {
23     tf.color =0xFFFF00;
24     txt.setTextFormat(tf);
25 }

```

شکل ۱۵-۱۰۶

نکته: برای تغییر سایز فونت می‌توانید از قالب کد زیر استفاده نمایید.

tf.size = عدد ;

۶ فیلم را تست کنید. وقتی ماوس را روی متن قرار می‌دهید رنگ متن زرد می‌شود در همان حالت دکمه ماوس را پائین نگه دارید رنگ متن قرمز خواهد شد و اگر دکمه ماوس را رها کنید رنگ متن سبز خواهد شد و اگر بخواهید ماوس را از متن خارج کنید رنگ متن آبی خواهد شد.

۱۵-۳۲ شناخت ساختار حلقه‌های تکرار

اگر بخواهیم یک یا چند دستور به دفعات معینی و یا به ازای برقراری شرطی تکرار شوند از ساختار حلقه‌های تکرار استفاده می‌کنیم

بر این اساس در Flash دو نوع حلقه تکرار وجود دارد، حلقه‌های تکرار شمارشی که با دستور For قابل انجام می‌باشند و حلقه‌های تکرار شرطی که با دستور While مورد استفاده قرار می‌گیرند. برای آشنایی هرچه بیشتر این دستورات در ادامه به بررسی شکل کلی آنها و نحوه اجرای این حلقه‌ها می‌پردازیم.

۱۵-۳۳ حلقه‌های for... In و for

همانطور که گفتیم از حلقه‌های For در مواردی استفاده می‌شود که تعداد دفعات حلقه‌های تکرار مشخص باشد. در Flash دو نوع حلقه تکرار شمارشی For In و For وجود دارد که در ادامه به بررسی آنها می‌پردازیم.

الف) حلقه تکرار شمارشی for

قبل از این که به بررسی اجزاء و نحوه کار این حلقه‌ها بپردازیم به شکل کلی این حلقه‌ها توجه کنید:

for (گام حرکت (پرش) ; شرط اتمام حلقه ; مقدار اولیه = نام شمارنده)

```
{
مجموعه دستورات
}
```

نکات مهم در مورد حلقه‌های تکرار For:

- نام شمارنده لازم است از قبل تعریف شده باشد
- چنانچه در گام حرکت بعد از نام متغیر از ++ استفاده شود گام مورد نظر افزایشی و در صورتی که از -- استفاده شود حلقه تکرار کاهش می‌یابد
- در این حلقه‌ها، شمارنده با توجه به میزان افزایش یا کاهش موجود در گام حرکت از یک مقدار اولیه تا نهایی تغییر کرده و دستورات داخل حلقه مرتباً تکرار خواهند شد.

تمرین ۲۱: قطعه برنامه زیر اعداد ۰ تا ۵ را روی پنجره خروجی نمایش می‌دهد. (حلقه افزایشی)

```
var i:int;
for (i = 0; i < 5; i++)
{
    trace(i);
}
```

شکل ۱۵-۱۰۷

نکته: ++i معادل i=i+1 در ریاضی بوده و با هر بار اجرای حلقه یک واحد به متغیر i اضافه خواهد شد.

تمرین ۲۲: قطعه برنامه زیر اعداد ۵ تا ۱ را روی پنجره خروجی نمایش می‌دهد. (حلقه کاهش‌دهنده)

```
var i:int;
for(i=5 ;i>0;i--)
{
    trace(i);
}
```

شکل ۱۵-۱۰۸

تمرین ۲۳: قطعه برنامه زیر ۵ بار عبارت Action Script را در پنجره خروجی نمایش می‌دهد

نکته: $i--$ معادل $i=i-1$ در ریاضی بوده و با هربار اجرای حلقه یک واحد از متغیر i کم خواهد شد.

```
var i:int;
for(i=1 ;i<=5;i++)
{
    trace('Action Script');
}
```

شکل ۱۰۹-۱۵

(ب) حلقه `for...in`

نوع جدیدی از حلقه‌های تکرار شمارشی می‌باشند که تکرار حلقه در آنها بر اساس تعداد ویژگی شیء مشخص شده انجام خواهد گرفت. شکل کلی این حلقه‌ها به صورت زیر است:

```
for (نام شیء in نوع داده‌ای : نام شمارنده)
{
    مجموعه دستورات
}
```

تمرین ۲۴: قطعه برنامه زیر به دلیل اینکه شیء مورد نظر دارای دو ویژگی می‌باشد ۲ بار عبارت `for...in` را در پنجره Output نمایش می‌دهد.

```
var myObj:Object = {x:20, y:30};
for (var i:String in myObj)
{
    trace('for...in');
}
```

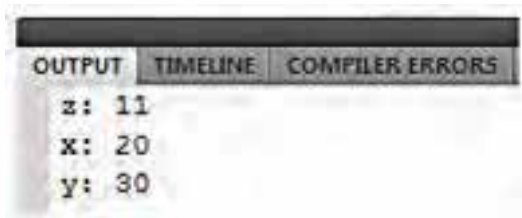
شکل ۱۱۰-۱۵

چون شیء `myObj` دارای دو ویژگی x و y می‌باشد حلقه `for` نیز دوبار تکرار شده و دو بار عبارت `for...in` در پنجره Output نمایش داده می‌شود

```
var myObj:Object = {x:20, y:30,z:11};
for (var i:String in myObj)
{
    trace(i + ": " + myObj[i]);
}
```

شکل ۱۱۱-۱۵

خروجی



شکل ۱۱۲- ۱۵

۱۵-۳۴ حلقه‌های تکرار شرطی while و do... while

در این نوع حلقه‌ها بر خلاف نوع قبلی، اجرای حلقه وابسته به برقراری شرط حلقه بوده و در صورت درست بودن این شرط حلقه تکرار خواهد شد. در Flash حلقه‌های While و Do...While جزء این دسته از حلقه‌ها محسوب می‌شوند.

الف) while

در این حلقه همانطور که در شکل کلی آن ملاحظه می‌کنید، ابتدا شرط حلقه بررسی شده و در صورت درست بودن شرط، حلقه تکرار خواهد شد.

while (شرط)
 {
 مجموعه دستورات
 }

تمرین ۲۵:

```
var i:int = 0;
while (i < 5)
{
    trace(i);
    i++;
}
```



خروجی

شکل ۱۱۳- ۱۵

۱-۳۴-۱۵ do...while

در این حلقه، بر خلاف حلقه While، ابتدا دستورات اجرا شده و سپس شرط بررسی می‌شود در صورت درست بودن شرط، حلقه مجدداً تکرار می‌گردد. شکل کلی آن به صورت زیر می‌باشد

```
do
{
مجموعه دستورات
} while (شرط)
```

نکته: به دلیل قرار گرفتن شرط حلقه در انتهای آن، تحت هر شرایطی حداقل یک بار تکرار خواهد شد.

تمرین ۲۶:

```
var i:int = 5;
do
{
trace(i);
i++;
} while (i < 5);
```

شکل ۱۱۴-۱۵

خروجی در حلقه فوق عدد ۵ می‌باشد. چرا؟

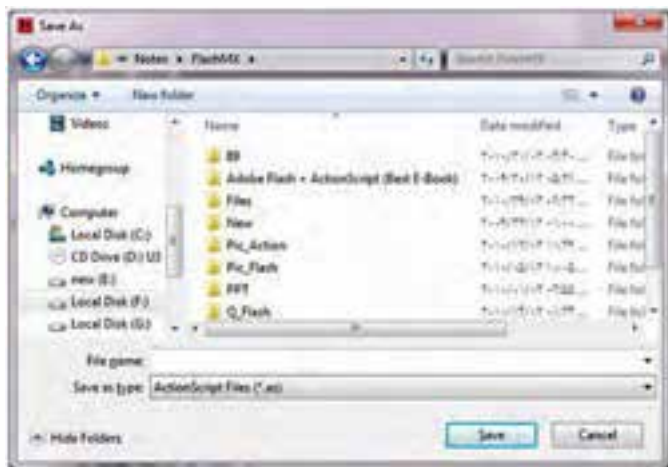
۳۵-۱۵ شناخت اصول استفاده از اسکریپت‌های خارجی

برای این که بتوانید از اسکریپت‌های خارجی استفاده کنید ابتدا باید بتوانید کدهای Action Script را در یک فایل جداگانه ذخیره کنید (فایل‌های اسکریپت دارای پسوند as. می‌باشند) پس از اینکه کدهای مورد نظر را در پانل Action Script وارد کردید از منوی پانل اکشن (Actions Panel Context Menu) که در زیر دکمه Close و بالای دکمه Help قرار دارد، گزینه Export Script... را انتخاب کنید (یا از کلید Ctrl+Shift+P استفاده کنید)



شکل ۱۱۵-۱۵

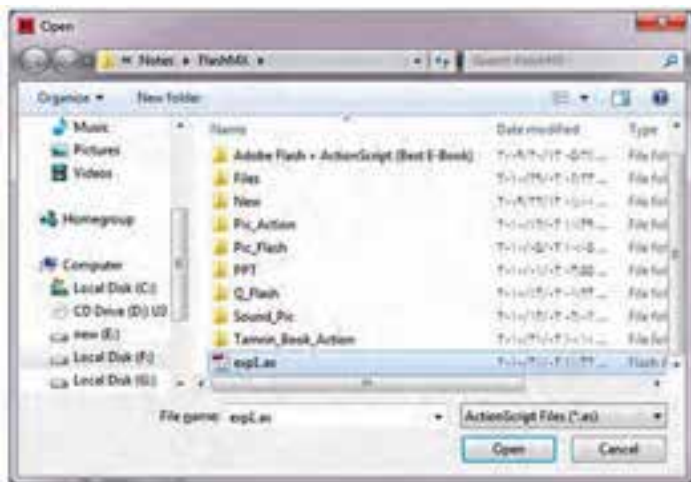
در پنجره Save As می‌توانید نام فایل Action Script را وارد کرده و پس از تعیین محل بر روی دکمه Save کلیک نمائید تا فایل as ذخیره شود. ضمن این که نام فایل را نیز exp1.as وارد کنید.



شکل ۱۱۶-۱۵

پس از این که کدهای Action Script را به صورت یک فایل با پسوند As ذخیره کردید نوبت آن است که از کدهای ذخیره شده در یک فایل Flash دیگر استفاده کنید برای این منظور مراحل زیر را انجام دهید.

۱. یک سند جدید ایجاد نموده و با نام ps3.fla ذخیره نمایید.
۲. برای باز شدن پنجره Action Script کلید F9 را فشار دهید.
۳. بر روی منوی پانل اکشن کلیک کرده و سپس گزینه Import Script... را انتخاب نمایید. (یا از کلید ترکیبی Ctrl+Shift+I استفاده کنید)
۴. در پنجره Open می‌توانید نام فایل Action Script مورد نظر را انتخاب کرده سپس بر روی دکمه Open کلیک نمایید.



شکل ۱۱۷-۱۵

با انجام مراحل فوق کدهای داخل فایل Action Script به فریم اصلی انتخاب شده اضافه می‌گردد.

۳۶-۱۵ تبدیل شی به کلاس و استفاده آن در Action Script

اگر شی را به کلاس تبدیل کنیم در Action Script می‌توان شی(های) جدیدی از روی کلاس ایجاد شده تعریف نمود سپس به stage اضافه نموده و کنترل‌های مورد نظر را نسبت به اشیای تعریف شده جدید اعمال نمود.

تمرین ۲۷- می‌خواهیم در زمان تست فیلم با کلیک در محل دلخواه یک پنچ ضلعی در محل کلیک ماوس اضافه شود.

۱. یک سند جدید به نام Class1.fla ایجاد کنید

۲. دو لایه به نامهای «محتوی» و «اکشن» ایجاد نمایید

۳. در لایه «محتوی» یک پنچ ضلعی رسم نموده و آن را به نماد MovieClip با نام پنچ ضلعی تبدیل نمایید.

۴. نام نمونه پنچ ضلعی را pg قرار دهید.

۵. در کادر Library بر روی پنچ ضلعی کلیک راست نموده و گزینه Properties را انتخاب انتخاب نمایید و

سپس گزینه Export for ActionScript را فعال نمایید. توجه داشته باشید که گزینه Export in frame 1

نیز فعال باشد. و نام کلاس را pgMC قرار دهید



شکل ۱۱۸-۱۵

۶. لایه جدیدی به نام «اکشن» ایجاد نمائید.

۷. اولین فریم لایه Actions را انتخاب نمود و با زدن کلید F9 وارد Action Script شوید و کد زیر را وارد نمائید.

```

1 stage.addEventListener(MouseEvent.CLICK ,CrPg);
2 function CrPg (evt:MouseEvent) :void
3 {
4     var newpg:pgMC=new pgMC();
5     addChild(newpg);
6     newpg.x=stage.mouseX;
7     newpg.y=stage.mouseY;
8 }
    
```

شکل ۱۱۹-۱۵

۸. با استفاده از Ctrl+Enter فیلم را تست کنید در هر نقطه‌ای کلیک کنید یک پنچ ضلعی ترسیم می‌شود.

Var newpg: pgMC=new pgMC(); شیء به نام newpg از نوع کلاس mypg تعریف شده است

addChild(newpg); یک شیء جدید از نوع newpg به صحنه اضافه می‌کند.

newpg.x=stage.mouseX مختصات افقی newpg را براساس محل کلیک ماوس تعیین می‌کند

newpg.y=stage.mouseY مختصات عمودی newpg را براساس محل کلیک ماوس تعیین می‌کند

می‌توان قبل از تبدیل شیء به کلاس، ابتدا به شیء جلوه حرکتی داد سپس به کلاس تبدیل نمود و در

Action Script اشیاء جدیدی از روی آن تعریف نمود

۹. فایل Class1 fla را با نام جدید Class2 fla ذخیره نمائید.

۱۰. بر روی پنچ ضلعی دابل کلیک نموده تا وارد خط زمان پنچ ضلعی شویم یک جلوه حرکتی به پنچ ضلعی

اضافه نمائید (کاری کنید پنچ ضلعی دور خودش بچرخد)

۱۱. از فیلم تست بگیرید حال با کلیک کردن بر روی stage یک پنچ ضلعی متحرک ایجاد می‌شود.

۳۷-۱۵ فراخوانی یک فایل swf خارجی

با استفاده از Action Script می‌توان یک یا چند فایل ایجاد شده با پسوند swf را به دلخواه فراخوانی نمود

برای فراخوانی ابتدا باید یک شیء از کلاس loader تعریف نمود سپس با استفاده از addChild شیء تعریف را

اضافه نموده و با استفاده از load و URLRequest فایل swf مورد نظر را فراخوانی نمود.

تمرین ۲۸- می‌خواهیم فایل ColorPicker.swf را که در مسیر جاری وجود دارد را در فلش فراخوانی نمائیم.

۱. یک سند جدید با نام loadSwf fla ایجاد نمائید

۲. دو لایه به نامهای «دکمه» و «اکشن» ایجاد نمائید

۳. در لایه «دکمه» یک متن با محتوی Play رسم نموده و آن را به نماد Button با نام دکمه تبدیل نمائید.

۴. نام نمونه دکمه را btnPlay قرار دهید.

۵. اولین فریم لایه Actions را انتخاب نمود و با زدن کلید F9 وارد Action Script شوید و کد زیر را وارد

نمائید.

```

1 btnPlay.addEventListener(MouseEvent.CLICK,playC);
2 function playC (evt:MouseEvent):void
3 {
4     var loadF:Loader=new Loader();
5     addChild(loadF);
6     loadF.load (new URLRequest("/ColorPicker.swf"));
7 }

```

شکل ۱۲۰-۱۵

ع از فیلم تست گرفته و بر روی دکمه Play کلیک کنید تا فایل ColorPicker.swf بر روی صحنه ظاهر شود.

نکته: برای آدرس دهی مسیر پوشه‌ها باید از علامت / استفاده نمائید برای مثال اگر فایل a.swf در فهرست myfolder درایو D: باشد پارامتر تابع URLRequest باید به صورت "D:/myfolder/a.swf" نوشته شود

جدول کلیدهای ترکیبی در Action Script

نام دستور یا تابع یا متد	کلید ترکیبی	نمایش در کد نویسی
trace	Esc+tr	trace()
function	Esc+fn	function () { }
for	Esc+fr	for () { }
do ...while	Esc+do	do { } while ();
if	Esc+if	if () { }

پروژه ۱: ساخت پازل اشکال

۱- سند جدیدی به نام dragdrop.fla ایجاد نمایید (w → 550 px h → 300 px)

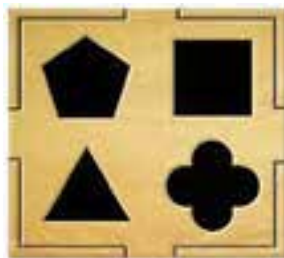
۲- ۵ لایه به نام های «اکشن»، «متن»، «اشکال»، «هدف» و «قاب» ایجاد نمایید.



شکل P-۱

۳- در فریم ۱ لایه «متن» یک فیلد متنی دینامیکی با محتوی Put the shapes away! ایجاد نمایید و نام نمونه آن را reply_txt قرار دهید

۴- تصویر toytop.png که در داخل هارد سیستم شما وجود دارد (قبلاً در فتوشاپ ایجاد شده است) به کتابخانه اضافه نمایید (ارتفاع تصویر 250 pt و عرض آن 280 pt می باشد)



شکل P-۲

از منوی File گزینه Import to library... را انتخاب نموده سپس فایل مورد نظر را انتخاب نمایید. و نام symbol را به grToyTop تغییر دهید

۵- در لایه «قاب» در فریم ۱ از کادر LIBRARY سمبل grToyTop را به صحنه درگ نمایید.

۶- در لایه «اشکال» در فریم ۱ یک مربع با مشخصات ذیل ترسیم نموده و آن را به سمبل MovieClip به نام «مربع» تبدیل نمایید و نام نمونه آن را mrb در نظر بگیرید

قرمز → color h → 74.7 w → 74.7 y → 30 X → 20

۷- در لایه «اشکال» در فریم ۱ یک پنج ضلعی با مشخصات ذیل ترسیم نموده و آن را به سمبل MovieClip به نام «پنج ضلعی» تبدیل نمایید و نام نمونه آن را pgz در نظر بگیرید

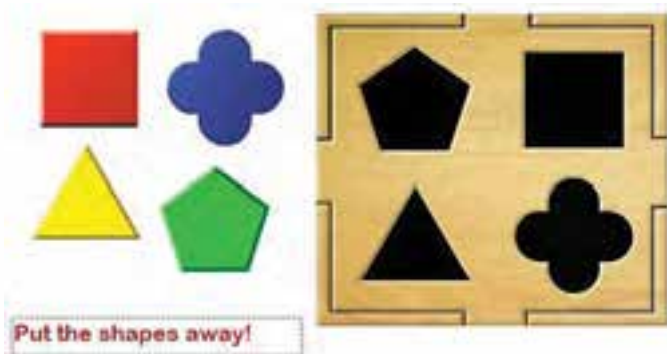
سبز → color h → 89.2 w → 93.8 y → 150 X → 150

۸- در لایه «اشکال» در فریم ۱ یک مثلث با مشخصات ذیل ترسیم نموده و آن را به سمبل MovieClip به نام «مثلث» تبدیل نمایید و نام نمونه آن را msl در نظر بگیرید

زرد → color h → 72 w → 84 y → 20 X → 120

۹- در لایه «اشکال» در فریم ۱ یک گل ۴ پر با مشخصات ذیل ترسیم نموده و آن را به سمبل MovieClip

به نام «گل» تبدیل نمائید و نام نمونه آن را glp در نظر بگیرید
 $X \rightarrow 135$ $y \rightarrow 20$ $w \rightarrow 90$ $h \rightarrow 90$ $color \rightarrow$ آبی



شکل ۳-P

۱۰- در لایه «هدف» در فریم ۱ یک مربع با مشخصات ذیل ترسیم نموده و آن را به سمبل MovieClip به نام «مربع هدف» تبدیل نمائید و نام نمونه آن را mrb_d در نظر بگیرید

$w \rightarrow 52$ $h \rightarrow 52$ $color \rightarrow \#00FF00$

۱۱- در لایه «هدف» در فریم ۱ یک پنج ضلعی با مشخصات ذیل ترسیم نموده و آن را به سمبل MovieClip به نام «پنج ضلعی هدف» تبدیل نمائید و نام نمونه آن را pgz_d در نظر بگیرید

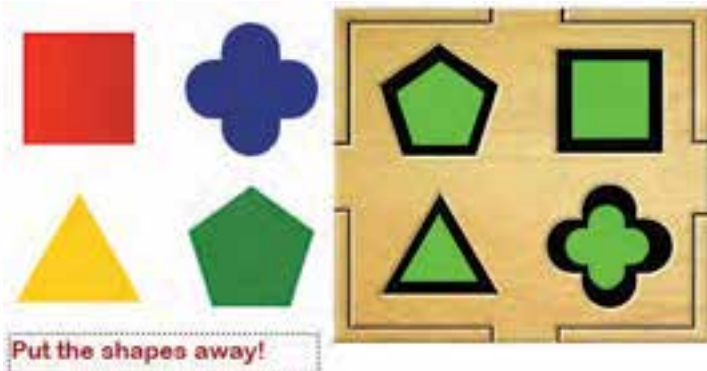
$w \rightarrow 93.8$ $h \rightarrow 89.2$ $color \rightarrow \#00FF00$

۱۲- در لایه «هدف» در فریم ۱ یک مثلث با مشخصات ذیل ترسیم نموده و آن را به سمبل MovieClip به نام «مثلث هدف» تبدیل نمائید و نام نمونه آن را msl_d در نظر بگیرید

$w \rightarrow 58$ $h \rightarrow 51$ $color \rightarrow \#00FF00$

۱۳- در لایه «هدف» در فریم ۱ یک گل چهارپر با مشخصات ذیل ترسیم نموده و آن را به سمبل MovieClip به نام «گل هدف» تبدیل نمائید و نام نمونه آن را glp_d در نظر بگیرید

$w \rightarrow 62$ $h \rightarrow 62$ $color \rightarrow \#00FF00$



شکل ۴-P

۱۴- شکل‌های لایه «هدف» که به رنگ سبز هستند را شفاف نمائید (alpha رنگ زمینه را به صفر تبدیل کنید)
نکته: برای درگ کردن (جابجایی) یک MovieClip با استفاده از ماوس از متد `startDrag` استفاده می‌شود و برای آزادسازی عمل درگ از متد `stopDrag` استفاده می‌گردد

۱۵- امکان درگ کردن مربع قرمز با استفاده از Action Script

در فریم ۱ لایه «اکشن» ابتدا کلید F9 را زده سپس کدهای زیر را برای قابل درگ نمودن مربع قرمز وارد نمائید

```

1 mrb.addEventListener(MouseEvent.CLICK, pickUp);
2 mrb.addEventListener(MouseEvent.CLICK, dropIt);
3 function pickUp(evt:MouseEvent):void
4 {
5     evt.target.startDrag(true);
6 }
7 function dropIt(evt:MouseEvent):void
8 {
9     evt.target.stopDrag();
10 }
11 mrb.buttonMode = true;
    
```

شکل P-۵

اگر با فشردن Ctrl+Enter خروجی را مورد مشاهده قرار دهید خواهید دید که فقط مربع قرمز قابل جابجا کردن می‌باشد. و کد خط ۱۱ برای این که وقتی ماوس را روی مربع قرار می‌دهیم حالت دکمه داشته باشد
 ۱۶- اگر دستور زیر را بعد از خط ۴ (بعد از دستور `event.target.startDrag(true);`) اضافه کنید در هنگام درگ نمودن عبارت Put the shapes away! نمایش داده نمی‌شود
`reply_txt.text = "";`

برای امکان درگ کردن اشکال دیگر کدهای زیر را بین خط دوم و سوم درج نمائید

```

mst.addEventListener(MouseEvent.CLICK, pickUp);
mst.addEventListener(MouseEvent.CLICK, dropIt);
glp.addEventListener(MouseEvent.CLICK, pickUp);
glp.addEventListener(MouseEvent.CLICK, dropIt);
pgz.addEventListener(MouseEvent.CLICK, pickUp);
pgz.addEventListener(MouseEvent.CLICK, dropIt);
    
```

شکل P-۶

کدهای زیر را نیز برای به حالت دکمه در آوردن سه شکل دیگر اضافه نمائید

```

glp.buttonMode = true;
mst.buttonMode = true;
pgz.buttonMode = true;
    
```

شکل P-۷

ما در این پروژه عمل مقایسه را بر اساس نام نمونه انجام می دهیم پس از ابتدا در نامگذاری دقت نمایید با استفاده از `evt.target.name` می توان نام شئی مقصد را که عمل درگ روی آن انجام می گیرد به دست آورد (evt همان منبع رویداد می باشد که آرگومان ورودی توابع تعریف شده می باشد)

۱۷- اگر کدهای تابع `dropIt` به صورت زیر تغییر دهید باعث می شود که در صورت قرار دادن مربع در کادر مربع هدف عبارت `Good Job` و در صورتیکه محل آن اشتباه باشد پیغام `Try Again` نمایش داده می شود

```
function dropIt(evt:MouseEvent):void {
    evt.target.stopDrag();
    var myTargetName:String = evt.target.name+ "_d";
    var myTarget:DisplayObject = getChildByName(myTargetName);
    if (evt.target.dropTarget != null && evt.target.dropTarget.parent == myTarget)
    {
        reply_txt.text = "Good Job!";
    }
    else
    {
        reply_txt.text = "Try Again!";
    }
}
```

شکل ۸-P

۱۸- کد زیر را بعد از خط `reply_txt.text = ""` در تابع `pickUp()` اضافه نمایید.

```
evt.target.parent.addChild(evt.target);
```

۱۹- حال می خواهیم زمانی که اشکال سمت چپ را در داخل حفره مورد نظر قرار می دهیم شکل در همان نقطه باقی بماند برای این کار باید مقدار `buttonMode` را برابر `false` قرار دهید و از متد `removeEventListener` نیز استفاده نمائید باید کدهای زیر را در تابع `dropIt()` بعد از خط `reply_txt.text="Good Job!"` اضافه نمائید.

```
evt.target.removeEventListener(MouseEvent.CLICK, pickUp);
evt.target.removeEventListener(MouseEvent.CLICK, dropIt);
evt.target.buttonMode = false;
```

شکل ۹-P

۲۰- برای منطبق شدن دقیق شکل ها بر روی حفره ها؛ مختصات `x` و `y` شکل در حال انتقال و مقصد (حفره) را یکسان نمائید کدهای زیر را بعد از خط `(evt.target.buttonMode = false;)` اضافه نمائید.

```
evt.target.x = myTarget.x;
```

```
evt.target.y = myTarget.y;
```

۲۱- با استفاده از کدهای زیر می خواهیم چنانچه محل قرار گرفتن اشکال درگ شده در حفره متناظر نبود به جای اولش برگردد

۱- تعریف متغیرهای مورد نیاز (در اولین سطر برنامه کدهای زیر اضافه شود) برای مختصات شروع حرکت (درگ) `startX` و `startY`

توانایی به کارگیری Action Script

```
var startX:Number;  
var startY:Number;
```

۲- مقدار دهی کردن متغیرها در تابع pickUp()

```
startX = evt.target.x;  
startY = evt.target.y;
```

۲۲- برگشت اشکال درگ شده به مبدا حرکت در صورتیکه در محل اصلی قرار نگرفته باشد. در تابع dropIt()

```
evt.target.x = startX;  
evt.target.y = startY;
```

۲۳- استفاده از شمارنده برای تعیین تکمیل شدن جابجایی ها

- تعریف و مقدار دهی اولیه متغیر شمارنده (محل قرار گرفتن کد بعد از تعریف دو متغیر قبلی)

```
var counter:Number = 0;
```

اضافه کردن شمارنده افزایشی در تابع dropIt()

```
counter++;
```

۲۴- چک کردن شمارنده و به روز رسانی متن متناسب با تکمیل جایگزینی در تابع dropIt() بعد از

```
if(counter == 4){  
    reply_txt.text = "Congrats, you're finished!";  
}
```

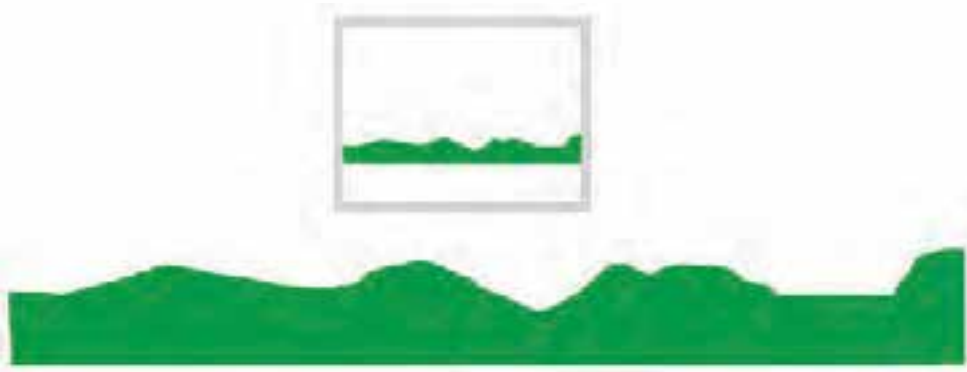
اگر شمارنده برابر ۴ باشد پیغام Congrats, you're finished! نمایش داده می شود
کد کامل

```
1 var startX:Number;  
2 var startY:Number;  
3 var counter:Number = 0;  
4 mrb.addEventListener(MouseEvent.CLICK, pickUp);  
5 mrb.addEventListener(MouseEvent.CLICK, dropIt);  
6 msl.addEventListener(MouseEvent.CLICK, pickUp);  
7 msl.addEventListener(MouseEvent.CLICK, dropIt);  
8 glp.addEventListener(MouseEvent.CLICK, pickUp);  
9 glp.addEventListener(MouseEvent.CLICK, dropIt);  
10 pgz.addEventListener(MouseEvent.CLICK, pickUp);  
11 pgz.addEventListener(MouseEvent.CLICK, dropIt);  
12 function pickUp(evt:MouseEvent):void  
13 {  
14     evt.target.startDrag(true);  
15     reply_txt.text = "";  
16     evt.target.parent.addChild(evt.target);  
17     startX = evt.target.x;  
18     startY = evt.target.y;  
19 }  
20 function dropIt(evt:MouseEvent):void  
21 {  
22     evt.target.stopDrag();  
23     var myTargetName:String = evt.target.name + "_d";  
24     var myTarget:DisplayObject = getChildByName(myTargetName);
```

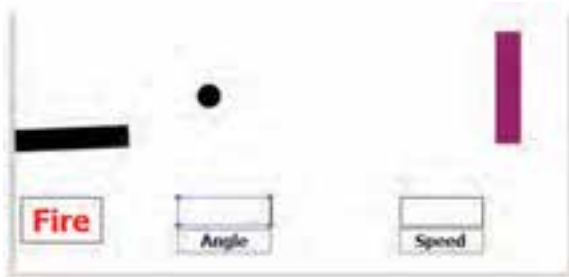
```
25 if (evt.target.dropTarget != null && evt.target.dropTarget.parent == myTarget)
26 {
27     reply_txt.text = "Good Job!";
28     evt.target.removeEventListener(MouseEvent.CLICK, pickUp);
29     evt.target.removeEventListener(MouseEvent.CLICK, dropIt);
30     evt.target.buttonMode = false;
31     evt.target.x = myTarget.x;
32     evt.target.y = myTarget.y;
33     counter++;
34 }
35 else
36 {
37     reply_txt.text = "Try Again!";
38     evt.target.x = startX;
39     evt.target.y = startY;
40 }
41 if(counter == 4)
42 {
43     reply_txt.text = "Congrats, you're finished!";
44 }
45 }
46 mrb.buttonMode = true;
47 glp.buttonMode = true;
48 msl.buttonMode = true;
49 pgz.buttonMode = true;
```

پروژه ۲: بازی پرتاب گلوله توپ با تعیین زاویه لوله توپ و سرعت حرکت توپ

- ۱- سند جدیدی به نام Ball_Fire.fla ایجاد نمائید
- ۲- سه لایه به نامهای «اکشن»، «زمینه» و «رابط» ایجاد نمائید.
- ۳- در فریم ۱ لایه «زمینه» شکل زیر را رسم نمائید. و آن را به سمبل MovieClip به نام «زمینه» تبدیل نمائید و نام نمونه آن را ground در نظر بگیرید



- ۴- در لایه «رابط» در فریم ۱
 - الف) یک دایره ترسیم نموده و آن را به سمبل MovieClip به نام «توپ» تبدیل نمائید و نام نمونه آن را ball در نظر بگیرید
 - ب) یک مستطیل به رنگ سیاه ترسیم نموده و آن را به سمبل MovieClip به نام «لوله توپ» تبدیل نمائید و نام نمونه آن را cannon در نظر بگیرید
 - ج) پنج کادر متن مطابق شکل زیر رسم کنید کادر متن Fire را به نماد Button تبدیل نموده و نام نمونه آن را btnfire قرار دهید و کادر متن بالای Angle را از نوع Input Text قرار داده و نام نمونه آن را angledeg قرار دهید و کادر متن بالای Speed را از نوع Input Text قرار داده و نام نمونه آن را speed قرار دهید



- د) یک مستطیل عمودی به رنگ ارغوانی ترسیم نموده آن را به سمبل MovieClip به نام «هدف» تبدیل نمائید و نام نمونه آن را target قرار دهید. حال بر روی مستطیل هدف دابل کلیک نموده تا وارد TimeLine شئی هدف شوید



فریم ۵ اولین لایه Timeline هدف را انتخاب نموده و یک فریم کلیدی ایجاد نمائید (با استفاده از کلید F6) سپس شکل زیر را در محل مستطیل ارغوانی ترسیم نمائید.



فریم ۱۰ اولین لایه Timeline هدف را انتخاب نموده و یک فریم کلیدی ایجاد نمائید (با استفاده از کلید F6) سپس شکل زیر را در محل شکل قلب ترسیم نمائید.



۱۵

فریم ۱ دومین لایه را انتخاب نموده و با استفاده از کلید F9 وارد پنجره Action Script شده و دستور زیر را بنویسید

```
stop();
```

فریم ۱۰ دومین لایه را انتخاب نموده و با استفاده از کلید F9 وارد پنجره Action Script شده و دستور زیر را بنویسید

```
stop();
```

۵- اولین فریم لایه «اکشن» را انتخاب نموده با استفاده از کلید F9 وارد پنجره Action Script شده و دستورات زیر را بنویسید
از کلاس Math ما دو متد مثلثاتی \sin و \cos و همچنین از ثابت π (برای تبدیل درجه به رادیان) استفاده شده است.

از متد `hitTestPoint` برای تطابق نقطه برخورد استفاده شده است به طوری که نقاط x و y شئی مبداء با شئی مقصد برخورد کند خروجی آن `true` خواهد بود و فرم کلی آن به صورت زیر می باشد

`hitTestPoint(X,Y,flag)` شئی مقصد

X, Y : مختصات شئی مبدا می باشد در صورت تطابق X, Y شئی مبدا با مقصد مقدار `flag` برابر `true` و گرنه برابر `false` خواهد بود

Action Script توانایی به کارگیری

```

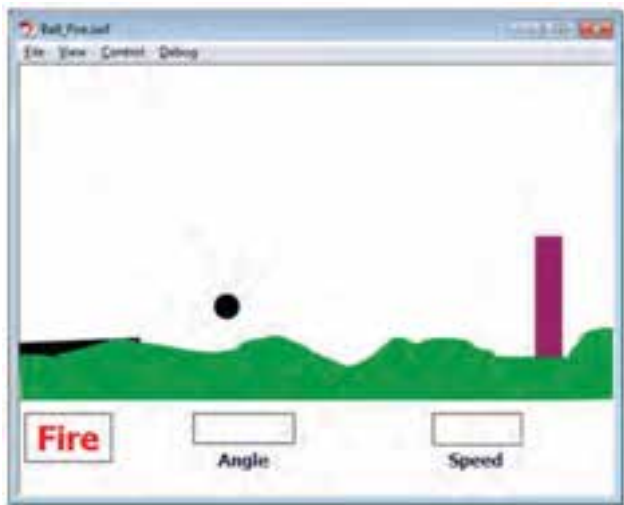
1 var xd:Number; var yd1:Number; var yd2:Number; var g:Number=2;
2 var mytimer:Timer = new Timer(50);
3 mytimer.addEventListener(TimerEvent.TIMER, moveball);
4 firebtn.addEventListener(MouseEvent.CLICK, fire);
5 function fire(ev:MouseEvent):void {
6     ball.visible = true;
7     var degrees = Number(angledeg.text);
8     cannon.rotation = -degrees;
9     var angler = degrees * Math.PI / 180;          var sp = Number(speed.text);
10    xd = Math.cos(angler)*sp;
11    yd2 = -Math.sin(angler)*sp;
12    target.gotoAndStop(1);
13    ball.x = cannon.x+ cannon.width;          ball.y = cannon.y-cannon.height;
14    mytimer.start();
15 }
16 function moveball(ev:TimerEvent):void {
17    ball.x += xd;    yd1=yd2;    yd2 = yd2+g;    ball.y +=.5*(yd1+yd2);
18    if (target.hitTestPoint(ball.x,ball.y, true)) {
19        mytimer.stop();          ball.visible = false;          target.gotoAndPlay(2);
20    }
21    if (ball.y > ground.y) {
22        ball.y = ground.y;          mytimer.stop();
23    }
24 }

```

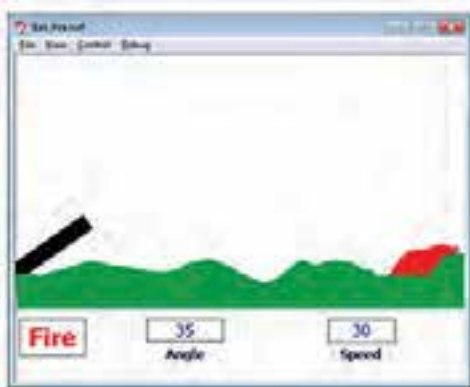
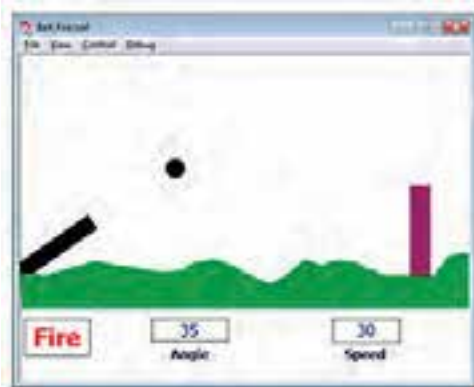
۱۵

X → 20 y → 30 w → 74.7 h → 74.7 قرمز → color

۷- در لایه «اشکال» در فریم ۱ یک پنچ ضلعی با مشخصات ذیل ترسیم نموده و آن را به سمبل MovieClip به

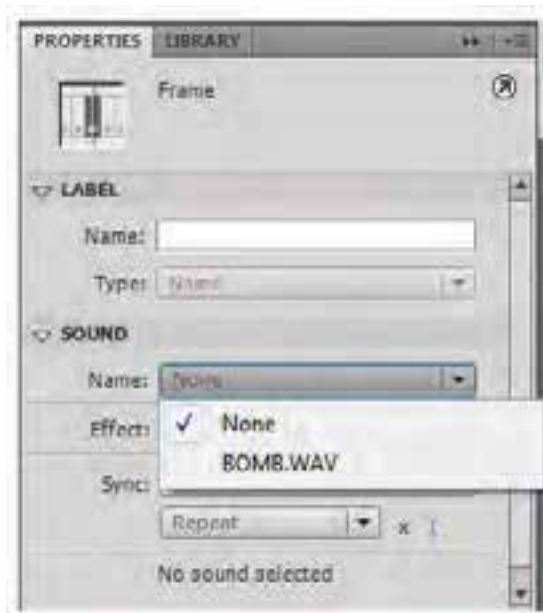


حال فیلم را تست کنید



برای صدا گذاری شلیک توپ

- ۱- از منوی File گزینه Import و سپس زیر گزینه Import to Library را انتخاب نمائید.
- ۲- فایل صوتی مورد نظر را انتخاب نموده (BOMB.WAV) و بر روی دکمه Open کلیک نموده تا فایل صوتی به کتابخانه (Library) اضافه شود.
- ۳- از کادر Library بر روی دکمه آتش دابل کلیک نمائید و فریم Down را انتخاب نموده و برای درج فریم کلیدی از کلید F6 استفاده نمائید.
- ۴- در کادر Properties در بخش Sound بر روی فلش جلوی Name کلیک کرده و گزینه BOMB.WAV را انتخاب نمائید. تا صدا به فریم Down اضافه شود.



- ۵- پس از اضافه شدن صدا به فریم Down علامت صدا در آن فریم ظاهر می‌شود.



حال اگر فیلم را تست کنید با هر بار کلیک بر روی دکمه Fire صدای شلیک توپ پخش می شود

پروژه ۳: کنترل حرکت ماشین با امکان دور زدن دایره‌ای

- ۱- سند جدیدی به نام moveCar.fla ایجاد نمایید
- ۲- دو لایه به نامهای «اکشن» و «محتوی» ایجاد نمایید.
- ۳- در فریم ۱ لایه «محتوی» شکل زیر را رسم نمایید. و آن را به سمبل MovieClip به نام «ماشین» تبدیل نمایید و نام نمونه آن را car_mc در نظر بگیرید



- ۴- در فریم ۱ لایه «محتوی» متن زیر را ایجاد نمایید.
Press Right and Left and Up keys to lead the car
- ۵- اولین فریم لایه «اکشن» را انتخاب نموده با استفاده از کلید F9 وارد پنجره Action Script شده و دستورات زیر را بنویسید




```
1 var myAngle:Number;
2 var xchange:Number;
3 var ychange:Number;
4 stage.addEventListener(KeyboardEvent.KEY_DOWN,ddnn);
5 function ddnn (evt:KeyboardEvent) :void
6 {
7     if (evt.keyCode== Keyboard.LEFT)
8     {
9         car_mc.rotation -= 10;
10    }
11    else if (evt.keyCode== Keyboard.RIGHT)
12    {
13        car_mc.rotation += 10;
14    }
15    myAngle = 90-car_mc.rotation;
16    xchange = Math.cos(Math.PI/180*myAngle)*5;
17    ychange = -Math.sin(Math.PI/180*myAngle)*5;
18    car_mc.x += xchange;
19    car_mc.y += ychange;
20    if (car_mc.x>stage.stageWidth)
21    {
22        car_mc.x = 0;
23    }
24    if (car_mc.y>stage.stageHeight)
25    {
26        car_mc.y = 0;
27    }
28    if (car_mc.x<0)
29    {
30        car_mc.x = stage.stageWidth;
31    }
32    if (car_mc.y<0)
33    {
34        car_mc.y = stage.stageHeight;
35    }
36 }
```

از کلاس Math ما دو متد sin و cos و همچنین از ثابت PI (برای تبدیل درجه به رادیان) استفاده شده است.

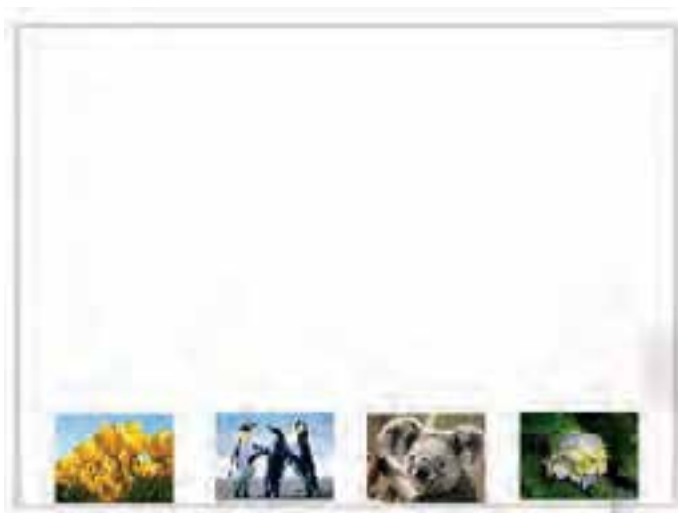
پروژه ۴: ساخت صفحه وب برای نمایش اسلایدی تصاویر

- ۱- سند جدیدی به نام SiteFlash.fla ایجاد نمائید
- ۲- ۵ لایه به نامهای «اکشن»، «گل ۱»، «گل ۲»، «کوالا» و «پنگوئن» ایجاد نمائید.
- ۳- چهار تصویر را که قبلاً به سایز 75*100 تبدیل نموده اید به کتابخانه اضافه نمائید (File > Import > Import to Library)
- ۴- هر چهار تصویر را در لایه مربوطه مطابق شکل زیر خارج از صفحه کاری قرار داده و آن ها را به نماد Button تبدیل نموده و نام نمونه آن ها را نیز به ترتیب btn1, btn2, btn3 و btn4 قرار دهید.



شکل ۱-P

- ۵- فریم ۱۵ هر چهار لایه را انتخاب نموده و با استفاده از کلید F6 یک فریم کلیدی ایجاد نمائید. و مطابق شکل زیر تصاویر را با فاصله مساوی در فریم ۱۵ قرار دهید.



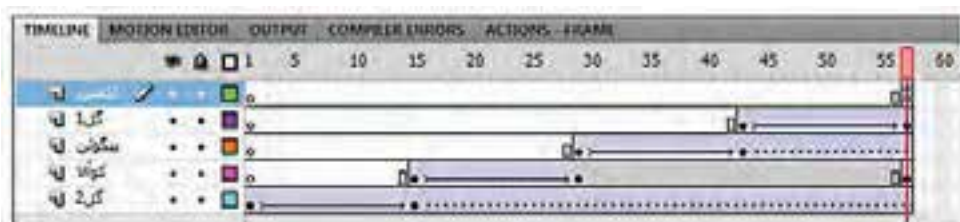
شکل ۲-P

۶- فریم ۱ تا ۱۵ هر چهار لایه را انتخاب کنید سپس کلیک راست نموده و گزینه Create Classic Tween را انتخاب نمائید.

۷- فریم ۱ هر چهار لایه را انتخاب نموده و Ease را معادل ۱۰۰ قرار دهید.

۸- فریم ۱۵ هر چهار لایه را انتخاب نموده و Ease را معادل ۱۰۰- قرار دهید.

۹- مطابق شکل زیر فریم های ۱ تا ۱۵ هر لایه را جابجا کنید.



شکل ۳- P

۱۰- فریم ۵۷ هر ۵ لایه را انتخاب نموده و با استفاده از کلید F5 یک فریم خالی ایجاد نمائید سپس فریم ۵۷ لایه «اکشن» را انتخاب نموده و با استفاده از کلید F9 وارد Action Script شده و متد stop() را برای توقف فیلم بنویسید.

۱۱- فیلم را تست کنید تصاویر به ترتیب وارد صحنه می شوند.

۱۲- فایل جدیدی به نام pic1 fla ایجاد نمائید.

۱۳- سه لایه به نامهای «تصویر»، «شکل» و «اکشن» ایجاد نمائید.

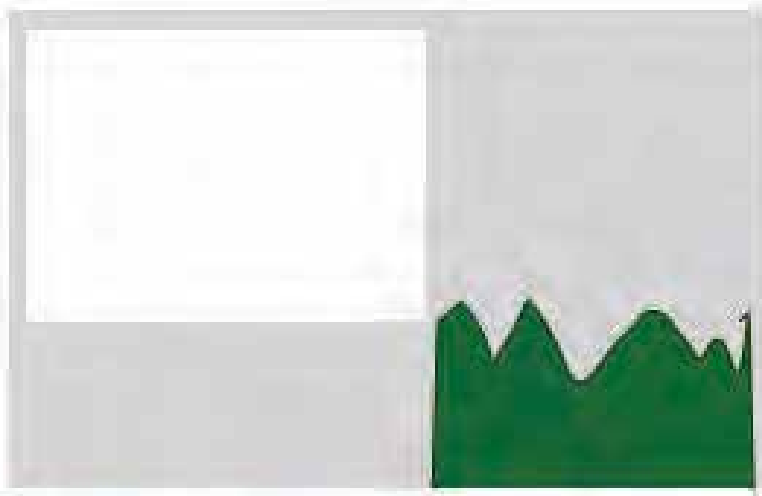
۱۴- تصویر به سایز 300*400 (همان تصاویر 75*100 که با اندازه 300*400 می باشند) را به کتابخانه اضافه نمائید

۱۵- در فریم ۱ لایه «شکل» با استفاده از ابزار Pen شکلی شبیه تصویر زیر را رسم کنید



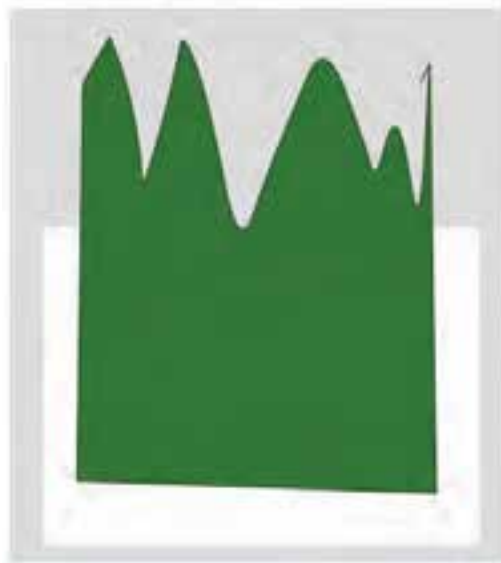
شکل ۴- P

۱۶- تصویر را به قسمت سمت راست پائین ، خارج از صفحه کاری انتقال دهید



شکل ۵-P

۱۷- فریم ۲۵ لایه «شکل» را انتخاب نموده و با استفاده از کلید F6 یک فریم کلیدی ایجاد نمائید سپس شکل ترسیمی را بر روی صفحه کاری انتقال داده و با استفاده از Free Transform ابعاد آن را مانند شکل زیر بزرگتر نمائید



شکل ۶-P

۱۸- بین فریم های ۱ تا ۲۵ لایه «شکل» کلیک راست نموده و گزینه Create Classic Tween را انتخاب نمائید.

۱۹- از کتابخانه Library تصویر اضافه شده را به لایه «تصویر» اضافه نمائید.



شکل ۷-P

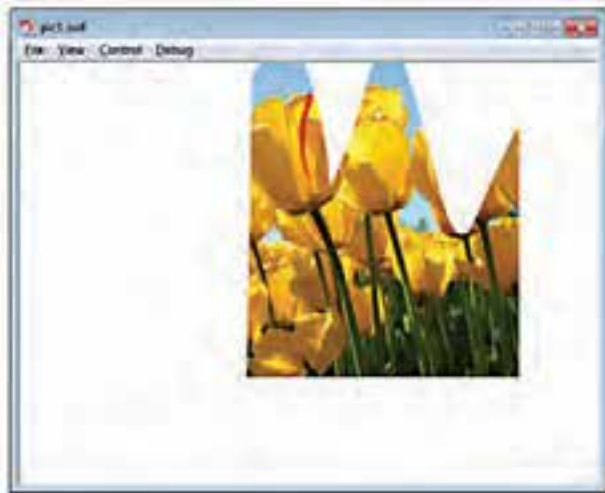
۲۰- بر روی لایه شکل کلیک راست نموده و گزینه Mask را انتخاب نمائید (توجه داشته باشد که لایه «تصویر» زیر لایه «شکل» قرار گرفته باشد).

۲۱- در فریم ۲۵ لایه «اکشن» با استفاده از کلید F6 یک فریم کلیدی ایجاد نمائید سپس وارد محیط Action Script شده و متد stop() را برای توقف فیلم وارد نمائید.



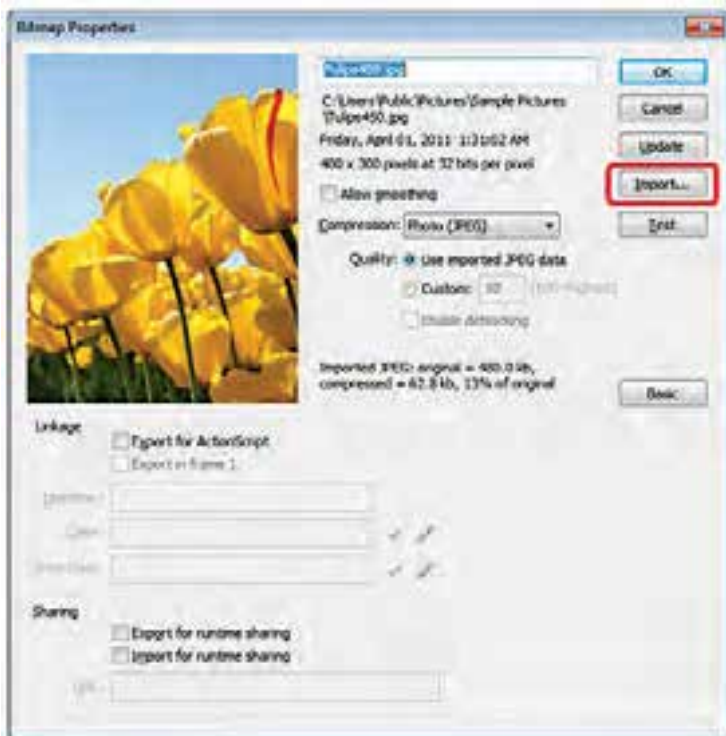
شکل ۸-P

۲۲- فیلم را تست کنید ابتدا تصویری روی صحنه دیده نمی شود به مرور از پائین سمت راست صحنه تصویر مانند شکل زیر رویت می شود تا کامل شود.



شکل ۹- P

۲۳- در Library بر روی تصویر دابل کلیک نموده تا کادر Bitmap Properties ظاهر گردد



شکل ۱- P

- ۲۴- بر روی گزینه Import کلیک کرده و فایل تصویری دوم را انتخاب نموده و فایل جاری را با نام جدید pic2.fla ذخیره نمائید به همین ترتیب فایل pic3.fla و pic4.fla را نیز ایجاد نمائید.
- ۲۵- حال فایل SiteFlash.fla را باز کرده و در فریم ۵۷ لایه «اکشن» کدهای زیر را اضافه نمائید

```

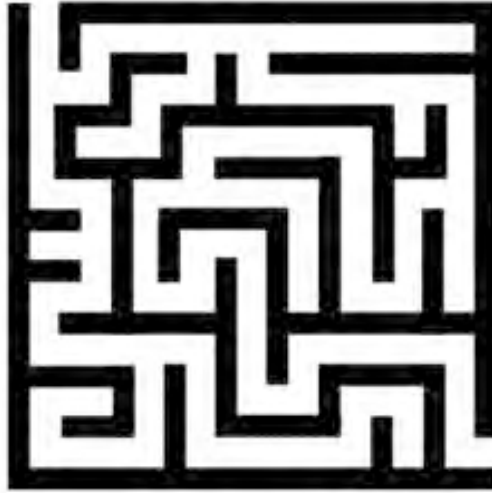
1 stop();
2 btn1.addEventListener(MouseEvent.CLICK,pic3);
3 btn2.addEventListener(MouseEvent.CLICK,pic2);
4 btn3.addEventListener(MouseEvent.CLICK,pic4);
5 btn4.addEventListener(MouseEvent.CLICK,pic1);
6 function pic1 (ev:MouseEvent):void {
7     var loadp:Loader=new Loader();
8     addChild(loadp);
9     loadp.load (new URLRequest("/pic1.swf"));
10 }
11 function pic2 (ev:MouseEvent):void {
12     var loadp:Loader=new Loader();
13     addChild(loadp);
14     loadp.load (new URLRequest("/pic2.swf"));
15 }
16 function pic3 (ev:MouseEvent):void {
17     var loadp:Loader=new Loader();
18     addChild(loadp);
19     loadp.load (new URLRequest("/pic3.swf"));
20 } function pic4 (ev:MouseEvent):void {
21     var loadp:Loader=new Loader();
22     addChild(loadp);
23     loadp.load (new URLRequest("/pic4.swf"));
24 }

```

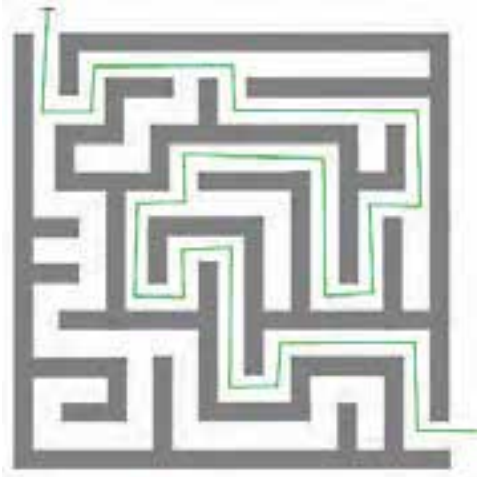
حال اگر فیلم را تست کنید ابتدا چهار تصویر کوچک بر روی صحنه ظاهر شده و با کلیک بر روی هر کدام از تصاویر فایل swf مربوط به pic1.fla , pic2.fla , pic3.fla , pic4.fla نمایش داده می شوند.

پروژه ۵: بازی Maze یا پیدا کردن مسیر

- ۱- سند جدیدی به نام maze.fla ایجاد نمایید
- ۲- سه لایه به نامهای «دکمه»، «اکشن»، «زمینه» و «متحرک» ایجاد نمایید.
- ۳- تصویر زیر را که قبلاً در نرم افزارهای گرافیکی ترسیم شده است وارد کتابخانه (Library) نمایید. و آن را در فریم ۱ لایه «زمینه» اضافه نمایید.



- ۴- در لایه «دکمه» در فریم ۱ متن مطابق شکل زیر رسم کنید کادر متن راهنما را ترسیم نموده و آن را به نماد Button تبدیل نموده و نام نمونه آن را btnhelp قرار دهید
- ۵- در فریم ۱ لایه «متحرک» در قسمت بالای سمت چپ maze دایره ای رسم کرده که اندازه آن کوچکتر از مسیر ورودی maze باشد و آن را به نماد MovieClip با نام توپ تبدیل نمایید
- ۶- بر روی شکل دایره دابل کلیک نموده تا وارد TIMELINE آن شوید و سه لایه به نام های «مسیر»، «توپ» و «اکشن ۲» ایجاد نمایید.
- ۷- در لایه «توپ» فریم ۶۰ را انتخاب نموده و با استفاده از دکمه F6 یک فریم کلیدی ایجاد نموده دایره را به پائین سمت راست maze (محل خروجی) انتقال دهید.
- ۸- بین فریم ۱ تا ۶۰ لایه «توپ» کلیک راست نموده و گزینه Create Classic Tween را انتخاب نمایید.
- ۹- فیلم را تست کنید دایره در یک مسیر مورب از بالا به پائین حرکت می کند.
- ۱۰- در لایه «مسیر» مطابق شکل زیر با استفاده Pen Tool مسیر سبز را رسم نمایید.



- ۱۰- بر روی نام لایه «مسیر» کلیک راست نموده و گزینه Guide را انتخاب نمائید.
- ۱۱- لایه «توپ» را به زیر لایه «مسیر انتقال داده» در فریم ۱ دایره را بر روی ابتدای مسیر سبز رنگ قرار داده و در فریم ۶۰ دایره را به انتهای مسیر سبز رنگ انتقال دهید تا حرکت توپ وابسته به مسیر سبز رنگ شود.
- ۱۲- فیلم را تست کنید دایره در مسیر مشخص شده سبز رنگ حرکت می کند.
- ۱۳- چون ما می خواهیم حرکت دایره بر روی مسیر در اختیار ما باشد فریم ۱ لایه «اکشن ۲» را انتخاب نموده و کد زیر را در پانل Action Script اضافه نمائید.

`stop();`

- ۱۴- بر روی Scene 1 کلیک نموده تا از TIMELINE نماد توپ خارج شده و وارد TIMELINE اصلی شوید
- ۱۵- فریم ۱ لایه «اکشن» را انتخاب و با استفاده از کلید F9 وارد محیط کد نویسی شوید و کدهای زیر را اضافه نمائید.

```

1 var x1:Number =tp.x;      var y1:Number =tp.y;
2 btnhelp.addEventListener(MouseEvent.CLICK,ply);
3 function ply (ev:MouseEvent):void
4 {
5     tp.x=x1;    tp.y=y1;    tp.play();
6 }
7 stage.addEventListener(MouseEvent.CLICK,mv);
8 function mv (evt:MouseEvent):void
9 {
10     switch (evt.keyCode)
11     {
12         case Keyboard.RIGHT :
13             tp.x+=1;
14             break;
15         case Keyboard.LEFT :
16             tp.x-=1;
17             break;
18         case Keyboard.UP :
19             tp.y-=1;
20             break;
21         case Keyboard.DOWN :
22             tp.y+=1;
23             break;
24     }
25 }

```

۱۶- فیلم را تست کنید با استفاده از کلیدهای جهتی می توانید دایره را حرکت دهید برای نمایش مسیر راهنما بر روی دکمه «راهنما» کلیک نمائید.

پروژه ۶- پیچ و خم ترسناک Scary Maze

- ۱- سند جدیدی به نام ScaryMaze.fla به ابعاد ۴۰۰ در ۳۰۰ با رنگ #32FF99 ایجاد نمائید
- ۲- دو لایه به نامهای «اصلی» و «اکشن» ایجاد نمائید.
- ۳- در فریم لایه «اصلی» یک متن با محتوی Scary Maze و یک متن با محتوی Play ایجاد نموده و متن Play را به نماد دکمه تبدیل نموده و نام نمونه آن را btnSt قرار دهید



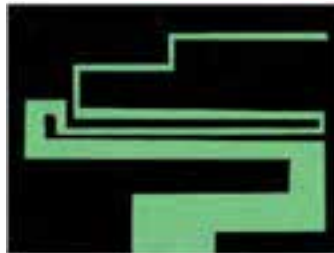
- ۴- لایه «اکشن» را انتخاب نموده و کد زیر را در پنجره Action Script وارد نمائید.

```

1 stop();
2 btnSt.addEventListener(MouseEvent.CLICK, nextSc);
3 function nextSc (ev:MouseEvent):void
4 {
5     gotoAndPlay("1", "Scene 2");
6 }
    
```

توجه: کد خط ۵ باعث می شود اولین فریم صحنه 2 پخش می شود که در ادامه 2 Scene را ایجاد نمائید.

- ۴- با انتخاب گزینه Scene از منوی Insert یک صحنه جدید ایجاد نمائید (نام صحنه 2 Scene باشد)
- ۵- سه لایه به نام های «طرح»، «هدف» و «اکشن ۲» ایجاد نمائید.
- ۶- فریم ۱ لایه «طرح» را انتخاب نموده و طرح زیر را که به رنگ سیاه است را با استفاده Pen Tool رسم کرده و آن را به نماد MovieClip تبدیل نموده و نام نمونه آن را wall_mc قرار دهید



- ۷- فریم ۱ لایه «هدف» را انتخاب نموده و مربع قرمز رنگ در انتهای مسیر سبز رنگ رسم نمائید و آن را به نماد MovieClip به نام هدف تبدیل نموده و نام نمونه آن را hit_mc قرار دهید

۸- لایه «اکشن ۲» را انتخاب نموده و کد زیر را در پنجره Action Script وارد نمایید.

```

1 stop();
2 stage.addEventListener(MouseEvent.CLICK, detectHits);
3 function detectHits(event:MouseEvent)
4 {
5     if (wall_mc.hitTestPoint(mouseX,mouseY,true))
6     {
7         gotoAndStop("1","Scene 1");
8         stage.removeEventListener(MouseEvent.CLICK, detectHits);
9     }
10    else if (hit_mc.hitTestPoint(mouseX,mouseY,true))
11    {
12        gotoAndPlay("1","Scene 3");
13        stage.removeEventListener(MouseEvent.CLICK, detectHits);
14    }
15 }

```

خط ۶ متد hitTestPont برای تطابق نقطه برخورد اشاره گر ماوس با دیواره سیاه رنگ استفاده شده است
خط ۸ متد removeEventListener برای حذف گوش کننده رویداد MOUSE_MOVE استفاده شده است
خط ۱۰ متد hitTestPont برای تطابق نقطه برخورد اشاره گر ماوس با هدف (مربع قرمز رنگ) استفاده شده است

۱۵

۹- با انتخاب گزینه Scene از منوی Insert یک صحنه جدید ایجاد نمایید (نام صحنه 3 Scene باشد)

۱۰- دو لایه به نام های «دکمه» و «اکشن ۳» ایجاد نمایید.

۱۱- فریم ۱ لایه «دکمه» یک تصویر دلخواه به صحنه اضافه نمایید (File > Import > Import to stage...)

۱۲- در فریم ۱ لایه «دکمه» یک متن با محتوی «دوباره؟» ایجاد نموده و آن را به نماد دکمه تبدیل نموده و نام نمونه آن را playAgain_btn قرار دهید



دوباره؟

۱۳- لایه «اکشن ۳» را انتخاب نموده و کد زیر را در پنجره Action Script وارد نمائید.

```

1 stop();
2 playAgain_btn.addEventListener(MouseEvent.CLICK, goBackScene);
3 function goBackScene (event:MouseEvent) {
4     gotoAndStop("1", "Scene 1");
5 }
    
```

۱۴- فیلم را تست کنید ابتدا Scene 1 ظاهر شده با کلیک بر روی دکمه Play وارد Scene 2 خواهید شد



چنانچه بتوانید بدون این که اشاره گر ماوس از محدوده سبز رنگ خارج نشود و آن را به انتها یعنی مربع قرمز رنگ برسانید وارد Scene 3 خواهید شد و گرنه مجدداً Scene 1 فعال خواهد شد



اگر در Scene 3 بر روی دکمه دوباره؟ کلیک کنید Scene 1 فعال خواهد شد

تمرین کاربردی برای حلقه **for**: تکرار یک **MovieClip** به تعداد مشخص با استفاده از حلقه **for**

- ۱- سند جدیدی به نام **for.fla** ایجاد نمائید
- ۲- دو لایه به نام‌های «اصلی» و «اکشن» ایجاد نمائید.
- ۳- تصویر زیر را که قبلاً در نرم‌افزارهای گرافیکی ترسیم شده است وارد کتابخانه (Library) نمائید. (ضمناً می‌توانید داخل محیط فلش نیز آن را رسم نمائید)



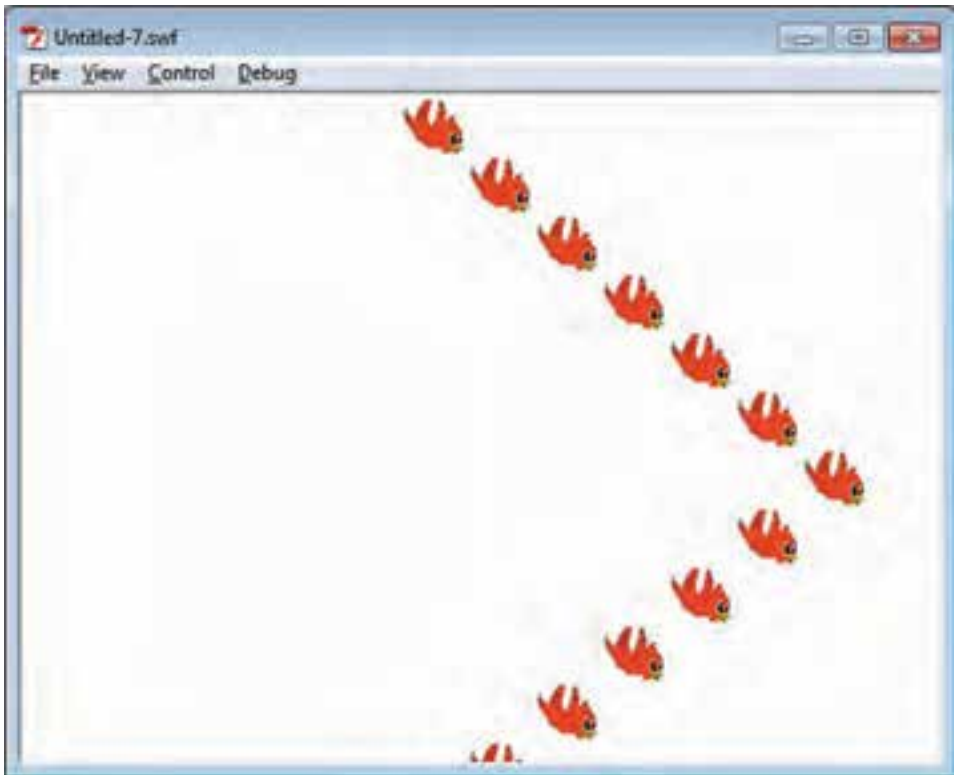
- ۴- تصویر را در گوشه بالای سمت چپ صفحه کاری با مشخصات زیر قرار دهید
 $X=0$ $Y=0$ $W=40$ $H=35$
- ۵- تصویر را انتخاب و به نماد **MovieClip** با نام **myMC** تبدیل نمائید
- ۶- بر روی تصویر پرنده دابل کلیک نموده تا وارد **TIMELINE** آن شوید فریم ۶۰ را انتخاب نموده و با استفاده از کلید **F6** یک فریم کلیدی ایجاد نمائید سپس پرنده را در انتهای سمت راست قرار دهید
- ۷- بین فریم ۱ تا ۶۰ کلیک راست نموده و گزینه **Create Classic Tween** را انتخاب نمائید.
- ۸- بر روی **Scene 1** کلیک نموده تا از **TIMELINE** نماد توپ خارج شده و وارد **TIMELINE** اصلی شوید
- ۹- فریم ۱ لایه «اکشن» را انتخاب و با استفاده از کلید **F9** وارد محیط کد نویسی شوید و کدهای زیر را اضافه نمائید.

```

1 var m:Number=0;
2 for (var i:Number=0; i<6 ; i++)
3 {
4     var my_mc:myMC = new myMC();
5     addChild(my_mc);
6     my_mc.y+=35*i;
7     my_mc.x+=40*i;
8 }
9 for (var k:Number=11; k>5 ; k--)
10 {
11     m++;
12     var my_mc2:myMC = new myMC();
13     addChild(my_mc2);
14     my_mc2.y+=35*k;
15     my_mc2.x+=40*m;
16 }

```

۱۰- فیلم را تست کنید . پرنده ها به صورت علامت > در حال حرکت خواهند بود



۱۵

خلاصه مطالب

- مجموعه کدهایی که در داخل یک فایل متنی قرار گرفته و قابلیت اجرا داشته باشند را اسکریپت می‌گویند
- آخرین نسخه اکشن اسکریپت در فلش CS4، Action Script 3.0 می‌باشد.
- برای نمایش پنجره اکشن اسکریپت می‌توان از منوی Window گزینه Actions را انتخاب کرده یا از کلید F9 استفاده کرد.
- Comments یا توضیحات به بخش هایی از یک برنامه گفته می‌شود که در اجرای برنامه هیچ تاثیری نداشته ولی برنامه نویسان از آنها برای افزایش درک برنامه و بررسی های احتمالی بعدی استفاده می‌کنند.
- دو نوع Comment یا توضیحات را می‌توان در Flash ایجاد کرد: ۱- // برای توضیح یک خطی و از علامت /* */ برای توضیح چند خطی مورد استفاده قرار می‌گیرد ۲- توضیحات برای فریم کلیدی در خط زمان
- برچسب‌ها نیز همانند توضیحات، خط زمان را قابل فهم تر می‌نمایند و به فریم های کلیدی اضافه می‌شوند ولی کاربرد دیگری نیز در هنگام استفاده از برنامه نویسی Action Script دارند. بعد اضافه کردن برچسب، علامت پرچم قرمز به فریم کلیدی انتخاب شده، اضافه شده و برچسب در ادامه نمایش داده می‌شود.
- از Instate name برای دسترسی به مشخصات اشیاء در اکشن اسکریپت استفاده می‌شود. در نام گذاری آن باید نکات زیر رعایت شود. ۱- نام باید با حروف یا زیرخط (Underscore) و یا علامت دلار (\$) شروع شود ۲- از علامت فاصله خالی Spacebar یا whitespace نمی‌توان استفاده نمود. ۳- نام نمونه باید منحصر به فرد (unique) باشد. ۴- از کلمات رزرو شده (reserved keyword) مانند if, for, else, and, or, ... نمی‌توان استفاده کرد.
- در اکشن اسکریپت هر عنصر درون فلش یک شیء محسوب می‌شود. به گروهی از اشیاء است که ویژگی های مشترکی داشته و رفتارهای یکسانی از خود نشان می‌دهند کلاس گفته می‌شود. شیء، نمونه خاصی از کلاس است. کلاس‌ها دارای سه مشخصه اصلی شامل: ویژگی، متد و رویداد بوده به طوری که این مشخصات اعضای یک کلاس محسوب می‌شوند. خواص Properties: از این مشخصه برای تعیین ویژگی های شیء در فلش استفاده می‌شود. متدها Methods: عملیاتی هستند که به صورت تابع نوشته می‌شوند و برای کنترل اشیاء به کار می‌روند. رویدادها Events: وقایعی هستند که شیء می‌تواند نسبت به آنها واکنش نشان دهد.
- بعد از نوشتن نام کلاس و گذاشتن علامت نقطه لیست اتومات مربوطه (Auto list) ظاهر می‌گردد
- برای کنترل حرکت در Timeline با استفاده از اسکریپت‌ها می‌توان از متدهای stop(), gotoAndPlay(), gotoAndStop(), nextFrame(), prevFrame(), play() استفاده کرد. (به قالب حروف‌ها دقت شود)
- کلمه کلیدی this یکی از دستورات اکشن اسکریپت است و به معنی ارجاع به خط زمانی است که کد در آن ظاهر شده است
- امکاناتی در اکشن اسکریپت برای قالب بندی کدها وجود دارد که به فهم راحت تر و خطایابی ساده تر ان کمک فراوان خواهد کرد. امکاناتی مانند: رنگ بندی کدها، قالب بندی اتومات، باز و بسته کردن کدها و... می‌باشد.

- فایل های اسکریپت دارای پسوند as، می باشند.
- متغیر (variable) مکانی در حافظه است که برای نگهداری یک مقدار مورد استفاده قرار می گیرد. قبل از استفاده از یک متغیر باید آن را با استفاده از var اعلان کنید. می توان با یک var چند متغیر تعریف نمود
- انواع داده ها در اکشن اسکریپت ۱- نوع داده ای String ۲- نوع داده ای uint ۳- نوع داده ای Boolean ۵- نوع داده ای Number
- منظور از دامنه (Scope) محدوده ای است که متغیر در آن شناخته شده و قابل مراجعه است. در اکشن اسکریپت متغیر در دو حالت محلی (Local) و سراسری یا فراگیر (Global) تعریف می شود. متغیر محلی فقط در محدوده ای که تعریف می شود قابل دسترس می باشد. متغیرهای محلی معمولاً بین دو آکولاد { } مربوط به توابع تعریف می شوند
- عملگرها علائمی هستند که موجب انجام عملیات بر روی عملوندهای خود می شوند. در Flash عملگرهای مختلفی وجود دارد که انواع آنها عبارتند از: الف) عملگرها ریاضی ب) عملگرهای مقایسه ای یا رابطه ای ج) عملگرهای انتساب د) عملگرهای منطقی ه) عملگر الحاق رشته
- ساده ترین دستور، برای تصمیم گیری در اکشن اسکریپت if می باشد اگر نیاز دارید چندین شرط با هم بررسی شود در این حالت می توانید از فرم if تودرتو استفاده کنید. در صورتیکه بخواهیم شرط به ازای مقادیر مختلف مورد بررسی قرار گرفته و گروهی از مجموعه دستورات اجرا شوند از دستور شرطی case ... switch استفاده می شود
- از حلقه های For در مواردی استفاده می شود که تعداد دفعات حلقه های تکرار مشخص باشد. در Flash دو نوع حلقه تکرار شمارشی For و For In وجود دارد در حلقه While و Do...While اجرای حلقه وابسته به برقراری شرط حلقه بوده و در صورت درست بودن این شرط حلقه تکرار خواهد شد. در Do...While، بر خلاف حلقه While، ابتدا دستورات اجرا شده و سپس شرط بررسی می شود در صورت درست بودن شرط، حلقه مجدداً تکرار می گردد.
- توابع، مجموعه ای از کدها برای وظایف خاص هستند که برای استفاده مجدد در برنامه به کار می روند. از توابع در اکشن اسکریپت برای سازماندهی و گروه بندی دستورات در یک برنامه استفاده می شود. در اکشن اسکریپت دو نوع تابع را می توان مورد استفاده قرار داد ۱- متدها ۲- بسته های تابعی
- (addEventListener): متدی برای تعیین رویدادها می باشد
- اگر شی را به کلاس تبدیل کنیم در اکشن اسکریپت می توان شی(های) جدیدی از روی کلاس ایجاد شده تعریف نمود

About ActionScript

The ActionScript scripting language lets you add complex interactivity, playback control, and data display to your application. You can add ActionScript in the authoring environment by using the Actions panel, Script window, or an external editor.

واژه‌نامه تخصصی	
Action	کنش
Break	شکستن-وقفه
Capability	توانایی-قابلیت
Carry out	انجام دادن
Chunks	مقدار قابل توجه-تکه
Collapse	به هم ریخته
Comment	توضیح
Conceptually	مفهومی
conditional	شرط
Container	ظرف
Content	محتوی
Convert	تبدیل کردن
Describe	شرح دادن
Elements	عناصر
Environment	محیط
Expand	گسترش دادن
Export	صادر کردن
External	خارجی
Facilitate	تسهیل کردن
Global	سراسری
Happen	رخ دادن
Hint	ایما-اشاره

واژه‌نامه تخصصی	
Identifier	علامت-نشان
Import	وارد کردن
Improvements	بهبود
Indentation	دندان-تورفتگی
Instance	نمونه
Instructions	دستورالعمل
Interactivity	تعامل
Label	برچسب
Legacy	میراث
Navigator	هدایتگر
Nevertheless	با این وجود
Object-oriented	شیء‌گرا
Operator	عملگر
Pieces	قطعات
Reserved keyword	کلمه رزرو شده
Retrieve	بازیابی
Reusable	قابل استفاده مجدد
Scope	دامنه
Sensitive	حساس
Transparency	شفافیت
Underscore	زیرخط
Variable	متغیر

خود آزمایی

- نحوه فعال ساختن پنجره Action Script در Flash را بیان کنید.
- روش‌های درج توضیحات در Action Script را بیان کنید.
- انواع متغیرهای عددی را نام ببرید.
- نحوه اضافه کردن برچسب به فریم را بیان کنید.
- چگونه می‌توان از اسکریپت خارجی در فلش استفاده نمود.
- با ذکر یک مثال فرمان Switch ... case را تشریح کنید.
- حلقه افزایشی و کاهش‌ی را با استفاده از حلقه for تشریح کنید.
- اصول درج تابع در Action Script را بیان کنید.

کارگاه flash

۱. با استفاده از برچسب و متدها یک شکل متحرک را کنترل نمائید. (بجای استفاده از شماره فریم از برچسب استفاده نمائید)
۲. دو دکمه بر روی stage قرار داده به طوری که با کلیک کردن بر روی دکمه اول یک صدای دلخواه پخش شود و با کلیک بر روی دکمه دوم صدا قطع شود
۳. یک ماشین را با استفاده از کلیدهای جهتی کنترل نمائید.



تمرین

در لوح فشرده تمرینات (در صورت موجود بودن) به فصل مربوطه رفته و هریک از محتویات پوشه result را در قالب فایل Fla ایجاد کنید.

پرسش‌های چهار گزینه‌ای

۱. اسکریپت چیست؟
 - الف) مجموعه کدهای قابل اجرا
 - ب) مجموعه کدهای غیر قابل اجرا
 - ج) مجموعه کدهای ایجاد شده پس از اجرا
 - د) Instance Name
۲. آخرین نسخه Action Script در فلش CS4 کدام گزینه است؟
 - الف) اکشن اسکریپت ۱
 - ب) اکشن اسکریپت ۲
 - ج) اکشن اسکریپت ۳
 - د) اکشن اسکریپت ۴

۳. برای نمایش پانل اکشن اسکریپت از کدام کلید تابعی استفاده می شود؟
 الف) F6
 ب) F7
 ج) F8
 د) F9
۴. برای نوشتن چندین خط توضیحات از کدام گزینه استفاده می شود؟
 الف) متن را بعد از علامت // می نویسیم
 ب) متن را بین دو علامت / / می نویسیم
 ج) متن را بین دو علامت /* */ می نویسیم
 د) متن را بین دو علامت /* / می نویسیم
۵. رنگ متن توضیحی در حالت پیش فرض کدام است؟
 الف) خاکستری
 ب) آبی
 ج) زرد
 د) سبز
۶. در کدام حالت پرچم قرمز رنگ به فریم اضافه می شود؟
 الف) بعد از اضافه کردن توضیحات
 ب) بعد از اضافه کردن برچسب
 ج) بعد از حذف برچسب
 د) بعد از حذف توضیحات
۷. کدام نام گذاری برای Instance Name مناسب می باشد؟
 الف) if
 ب) 3am
 ج) \$am
 د) ac scr
۸. کدام یک از دستورات زیر صحیح است؟
 الف) MyButton.Width:500;
 ب) MyButton.Width=500;
 ج) MyButton.Width(500);
 د) MyButton.Width{500};
۹. اگر در فریم جاری، کدهای Action Script اضافه شود کدام علامت در فریم قرار می گیرد؟
 الف) s
 ب) a
 ج) h
 د) as
۱۰. برای تعیین پهنای شیء، از کدام ویژگی استفاده می شود؟
 الف) x
 ب) y
 ج) height
 د) width
۱۱. برای درج تابع trace از کدام گزینه استفاده می شود؟
 الف) Ctrl+Esc
 ب) Esc+t
 ج) Esc+tr
 د) Ctrl+tr

۱۲. برای ارجاع به خط زمانی که کد در آن ظاهر شده است، از کدام گزینه استفاده می‌شود؟

الف) this

ب) frame

ج) tab

د) index

۱۳. کدام متد، آرگومان ورودی ندارد؟

الف) stop

ب) gotoAndPlay

ج) gotoAndStop

د) الف و ج

۱۴. کدام متد، برای پخش فیلم متوقف شده، مورد استفاده قرار می‌گیرد؟

الف) nextFrame

ب) gotoAndPlay

ج) gotoAndStop

د) play

۱۵. کدام متد، برای پخش فریم خاص مورد استفاده قرار می‌گیرد؟

الف) nextFrame

ب) gotoAndPlay

ج) gotoAndStop

د) play

۱۶. کلمات رزرو شده در Action Script در حالت پیش فرض با چه رنگی نمایش داده می‌شود؟

الف) سبز

ب) خاکستری

ج) آبی

د) سیاه

۱۷. برای بسته نمودن کدهای داخل علامت {} از کدام علامت استفاده می‌شود؟



د)



ج)



ب)



الف)

۱۸. در قالب‌بندی خودکار کدام عمل صورت می‌گیرد؟

الف) قرار دادن ؛ در انتهای هر دستور

ب) قرار دادن «» در دو طرف رشته‌ها

ج) حذف پرانتزهای اضافی

د) تبدیل حروف بزرگ به کوچک

۱۹. پسوند فایل‌های Action Script خارجی کدام است؟

الف) as

ب) swf

ج) fla

د) fls

۲۰. کدام نوع داده‌ای برای اعداد صحیح مثبت کمتر از ۱۰۰ مناسب می‌باشد؟

الف) int

ب) uint

ج) number

د) Boolean

۲۱. کدام نوع داده‌ای برای متغیرهایی که دارای دو حالت صحیح و غلط است، مناسب می‌باشد؟

الف) int

ب) uint

ج) number

د) Boolean

۲۲. کدام نام گذاری برای یک متغیر مناسب می باشد؟

- الف) \$actions
ب) int a
ج) 3int
د) for

۲۳. کدام گزینه عملگر مقایسه‌ای است؟

- الف) !=
ب) /=
ج) +=
د) % =

۲۴. کدام عملگر معادل and منطقی می باشد؟

- الف) \$\$
ب) &&
ج) ||
د) %%

۲۵. اگر a=5 باشد، خروجی عبارت a*=3 کدام گزینه است؟

- الف) 15
ب) 2
ج) 1
د) 0

۲۶. کدام حلقه تحت هر شرایطی حداقل یک بار تکرار می شود؟

- الف) for
ب) while
ج) do...while
د) switch...case

۲۷. حلقه زیر چند بار تکرار می شود؟

```
var i:int=3;
for (i=0;i<=5;i++)
{
    trace('Action');
}
```

- الف) دو بار
ب) سه بار
ج) پنج بار
د) شش بار

۲۸. خروجی قطعه برنامه زیر کدام است؟

```
var i:int=3;
if (!(i=0) && i<=5)
{trace('True');}
else
{trace(i!=3);}
```

- الف) true
ب) false
ج) 3
د) i!=3

۲۹. کدام رویداد مربوط به کاربر نمی باشد؟

- الف) KEY_UP
ب) MOUSE_DOWN
ج) CLICK
د) ENTER_FRAME

سوالات زبان تخصصی

1. You can add ActionScript in the authoring environment by using the
 - a) Actions panel
 - b) Script window
 - c) An external editor
 - d) a, b, c
2. Scripts thatthe FLA file.
 - a) Are embedded in
 - b) are separate from
 - c) is independent from
 - d) b, c
3. Which item is correct?
 - a) ActionScript is what's known as an object-oriented programming language.
 - b) A method represents one of the pieces of data that are bundled together in an object.
 - c) A property is an action that can be performed by an object
 - d) a, b
4. Which sign is between object and property?
 - a) Dash
 - b) point
 - c) Slash
 - d) Parenthesis