

الفصل الرابع عشر أساسيات ActionScript

نتعرف في هذا الفصل على مفهوم لغة ActionScript كلغة برمجة مستخدمة داخل برنامج Flash، كما نتعرف على الهدف من استخدام الكود داخل Flash وكيفية استخدام كود اللغة في تحقيق التفاعل بين المستخدم والفيلم.

بانتهاء هذا الفصل، سنتعرف على ما يلي:

- ◆ مفهوم لغة ActionScript
- ◆ فهم مصطلحات ActionScript.
- ◆ بعض مصطلحات ActionScript.
- ◆ المهام التي يمكنك تنفيذها من خلال كود ActionScript
- ◆ أين يتم تخزين كود ActionScript؟
- ◆ استخدام لوحة الإجراءات
- ◆ قواعد كتابة الكود.
- ◆ استخدام مستكشف الفيلم في الوصول للكود.
- ◆ استخدام Actionscript داخل الفيلم.
- ◆ ربط الأزرار بالإجراءات
- ◆ ربط الإطارات بالإجراءات

قبل أن نبدأ في هذا الفصل سنتناول بالشرح مقدمة عن **ActionScript** قبل أن نشرح كيفية ربط كل من الأزرار والإطارات بالإجراءات. وحيث أن موضوع **ActionScript** يستخدمه بصفة أساسية المبرمجون ذوي الخبرة بلغات البرمجة. لأنه يقف إلى جانب تلك اللغات ويستخدم كما تستخدم. فإننا في هذا الفصل نقدم فكرة مبسطة مع أمثلة بسيطة عن مفهوم **ActionScript** واستخدامها لربط كل من الأزرار والإطارات بالإجراءات.

ولمزيد من المعلومات عن لغة **ActionScript** راجع كتابنا استخدام



Flash مع ActionScript

مفهوم لغة ActionScript

يستخدم برنامج **Flash CS4** لغة **ActionScript** وهي لغة قوية تستخدم لتوسيع وظائف **Flash** وهي عبارة عن لغة برمجة كلغة **JavaScript** ولغة **Java** ولغة **C++** وغيرها من لغات البرمجة المختلفة، ويمكنك من خلال هذه اللغة إرسال الأوامر والتعليمات والأسئلة المختلفة إلى برنامج **Flash** عن سطور الوقت والمقطوعات الفيلمية والعناصر والكائنات الأخرى الموجودة داخل الفيلم كالأزرار والرموز والحالات. وعادةً يمكنك باستخدام **ActionScript** إجراء المهام المعقدة التي لا يمكنك بسهولة إنجازها من خلال نافذة التصميم أو ربما التي لا تستطيع من الأساس إنجازها من خلال هذه النافذة. ويمكن القول بأن عدم استخدامك لكوود **ActionScript** داخل أفلامك التي تقوم بإنشائها داخل **Flash** يعني عدم استخدامك لبرنامج **Flash** من الأساس. فأولى بك في هذه الحالة استخدام أحد برامج الرسوم البسيطة كبرنامج **PowerPoint** على سبيل المثال. فعلى سبيل المثال، كى تتمكن من التفاعل مع الفيلم من خلال أحد أزرار الأمر الموجودة بهذا الفيلم، يجب في هذه الحالة وجود كود **ActionScript** المصاحب لهذا الزر والذي يخبر البرنامج ما يجب فعله عند نقر هذا الزر. وكذلك الحال إذا أردت تنفيذ عملية أو أكثر عند ضغط مفتاح من لوحة المفاتيح، حيث يجب حينئذٍ تحديد كود **ActionScript** الذى يقوم بتنفيذ هذه العملية أو تلك العمليات. كما أن **ActionScript** هو الطريقة الوحيدة التي

يمكنك من خلالها نقل رأس التشغيل إلى إطار معين بسطر الوقت ثم الاستمرار من هذا الإطار أو ربما التوقف إذا أردت ذلك. وما ذكرته هنا ما هو إلا جزء بسيط من استخدام كود ActionScript داخل أفلامك التي تقوم بإنشائها داخل Flash، وسوف نتعرف من خلال هذا الفصل والفصول التالية على المزيد من هذه الاستخدامات.

إضافةً إلى ما سبق، يؤدي استخدامك لكود ActionScript داخل أفلامك إلى الحصول على ملفات SWF صغيرة الحجم مقارنةً بالملفات التي تقوم فيها بإنجاز نفس الأعمال من خلال الأدوات المختلفة الموجودة بنافذة التصميم، كما يمكنك أيضاً من خلال كود ActionScript إنجاز المهام بدقة متناهية، فيمكنك على سبيل المثال نقل إحدى المقطوعات الفيلمية من مكانها إلى مكانٍ آخر على مسرح العمل بدقة متناهية نظراً لأنك تقوم بتعيين إحداثي المكان الجديد وبالتالي لا يكون هناك مجال لأي تفاوت.

فهم مصطلحات ActionScript

كود ActionScript عبارة عن مجموعة من الكلمات باللغة الإنجليزية مصحوبةً ببعض الرموز الحسابية وعلامات الترقيم. ففي الكود التالي على سبيل المثال:

```
on (press) {  
    gotoAndPlay ("my frame");  
}
```

يمكنك بسهولة التعرف على وظيفة هذا الكود من خلال الكلمات الأساسية الموجودة بداخله. فكلمة **on** تعني "عند" أو "عندما" بينما تعني كلمة **press** "ضغط" أو "نقر"، وكلمة **gotoAndPlay()** عبارة عن كلمة مركبة وتعني الذهاب إلى مكان ما (الإطار **my frame** في هذه الحالة ثم الاستمرار في التشغيل. واستنتاجك صحيح مائة في المائة. فهذا الكود يخبر مفسر ActionScript بالذهاب إلى الإطار ذي العنوان **my frame** والاستمرار في تشغيل الفيلم عند نقر المستخدم للكائن الذي يحتوي على هذا الكود بزر الفأرة الأيسر. فإذا قمت بوضع الكود السابق مع زر أمر، يتم تنفيذ الكود بمجرد نقر المستخدم لزر الأمر هذا بزر الفأرة الأيسر.

بعض مصطلحات ActionScript

المتغير Variables

يمثل المتغير جزء معين من البيانات سواء كان هذا الجزء ثابتاً أو متغير، فعند إنشاء أو تعريف أى متغير فسنقوم بتحديد نوع البيانات **Data type**، التى يمثلها هذا المتغير، فمثلاً متغير السلسلة **String** يحمل أى سلسلة من الأحرف و الأرقام بينما المتغير **Number** يجب أن يحتوى رقم.

مثلاً بفرض أنك تريد تحديد متغير لعدد اللقطات الموجودة فى الفيلم، بفرض أن اسم المتغير هو **Frames** ، وأن القيمة الأساسية له هى ٠ وأن المطلوب هو ضبط الكود لإضافة ١ عندما ينتقل إلى لقطة أخرى. فى هذه الحالة سيبدو الكود اللازم لهذا الغرض هكذا:

```
Var frames =0;
Currentframes = frames + 1;
```

يجب أن تكون أسماء المتغيرات فريدة، وهى حساسة لحالة الأحرف، فالمتغير **frames** يختلف عن المتغير **Frames**، وقد تحتوى أسماء المتغيرات على أسماء أو أرقام أو أحرف أو شرطات ولا يمكن أن تبدأ برقم فيمكن تسمية المتغير **frames_2** أو **frames2** وليس **2frames**



الكلمة الأساسية Keywords

هى عبارة عن كلمة محجوزة تستخدم لإنجاز مهمة محددة ، فمثلاً **var** تعتبر كلمة أساسية تستخدم لإنشاء متغير للوصول إلى هذه الكلمات الأساسية يمكنك البحث فى تعليمات المساعدة **Flash** للتعرف عليها لإيجادها حيث ستجد صفحة كاملة لهذه الكلمات.

المعايير Parameters

البارميتر هى عبارة عن معايير تزود الكود بتفاصيل معينة ومحددة وهى القيم الموجودة بين أقواس () فى سطر الكود فمثلاً **gotoAndPlay(3)** فإن البارميتر يرشد الكود فى الذهاب إلى اللقطة رقم ٣.

الوظيفة Function

هي مجموعة من العبارات يمكن الإشارة إليها بالاسم، استخدام الوظيفة يمكن من تنفيذ نفس المجموعة من العبارات دون الحاجة لتكرار كتابتها في نفس الكود.

المهام التي يمكنك تنفيذها من خلال كود ActionScript

يتكون الفيلم داخل Flash كما تعلم من مجموعة من المشاهد، كما يحتوى كل مشهد على سطر وقت مستقل، حيث يبدأ كل سطر وقت بالإطار رقم ١ ويستمر إلى آخر إطار معرف بهذا السطر. والسلوك الطبيعي للأفلام داخل Flash هو التنقل إلى الأمام بمعدل ثابت معرف مسبقاً من الإطار الأول بالمشهد الأول إلى الإطار الأخير بنفس المشهد ثم الإطار الأول بالمشهد الثاني إلى الإطار الأخير بهذا المشهد وهكذا إلى أن نصل إلى نهاية المشهد الأخير بالفيلم. ويمكنك ضبط الفيلم للتوقف عند الوصول إلى آخر إطار أو الاستمرار من البداية مرة أخرى إلى أن يقوم المستخدم بإغلاق الفيلم بنفسه.

والهدف الأساسي لكود ActionScript هو تغيير هذا السلوك. فمن خلال كود ActionScript يمكنك إيقاف الفيلم عند إطار معين أو الرجوع مرة أخرى إلى إطار سابق أو تمكين المستخدم من اختيار الإطار الذي يرغب في تشغيله. كما يمكنك بالإضافة إلى ذلك استخدام ActionScript في أداء عدد من المهام والتي على رأسها ما يلي:

- إنشاء الرسوم المتحركة شديدة التعقيد. فعلى سبيل المثال، يمكنك من خلال ActionScript تدوير كرة حول الشاشة في مسار ثابت باستمرار من خلال إطار واحد فقط بدلاً من الحاجة إلى آلاف من الإطارات إذا أردت أداء ذلك من خلال نافذة التصميم وبدون استخدام كود ActionScript.
- يمكنك من خلال كود ActionScript التحكم في حركة الفيلم إلى الأمام أو إلى الخلف وتمكين المستخدم من أداء ذلك من خلال مجموعة من خيارات إحدى القوائم أو من خلال مجموعة من أزرار الأمر.

- يمكنك من خلال كود **ActionScript** استقبال البيانات من المستخدم أثناء تشغيل الفيلم ومن ثم استخدام هذه البيانات داخل الفيلم أو إرسالها إلى الخادم.
- يمكنك من خلال كود **ActionScript** إحضار البيانات من الخادم وعرضها للمستخدم من خلال الفيلم.
- يمكنك من خلال كود **ActionScript** إجراء العمليات الحسابية المختلفة.
- يمكنك من خلال كود **ActionScript** تغيير حجم ودوران وميل ولون المقطوعات الفيلمية الموجودة بالفيلم أثناء التشغيل، كما يمكنك استنساخ هذه المقطوعات أو حذفها من على مسرح العمل أثناء التشغيل.
- يمكنك من خلال كود **ActionScript** لاختبار بيئة تشغيل الفيلم والتعرف وقت وتاريخ تشغيله ووقت الجزء المنقضى منه.
- يمكنك من خلال كود **ActionScript** تشغيل الأصوات والتحكم في أحجامها واتزانها.

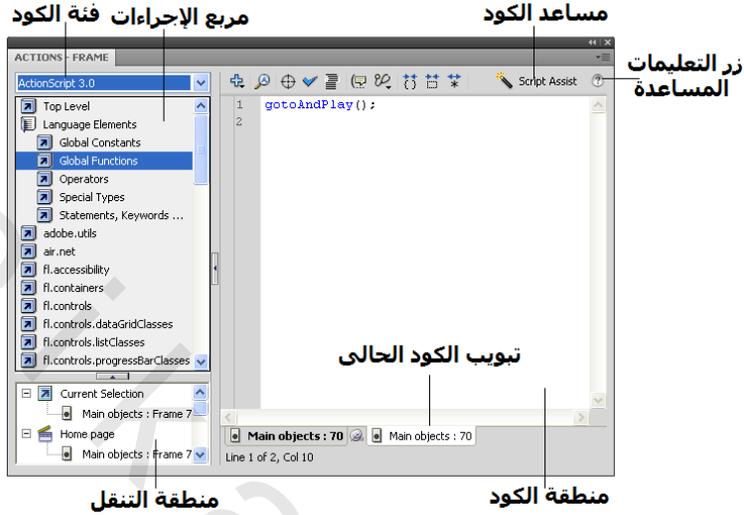
أين يتم تخزين كود **ActionScript** ؟

- يمكنك إدخال كود **ActionScript** إلى الفيلم من خلال لوحة الإجراءات **Actions panel** كما سنعرف بعد قليل. ولكن أين يتم تخزين هذا الكود ؟.
- يتم حقيقةً تخزين كود **ActionScript** داخل أحد الأماكن الأربعة التالية:
- داخل سطر الوقت، حيث يمكنك تخزين كود **ActionScript** بأى إطار رئيسى بسطر الوقت. انقر الإطار داخل سطر الوقت لتنشيطه ثم قم بإدخال الكود إلى لوحة الإجراءات. ويتم تنفيذ الكود الموجود بسطر الوقت بمجرد وصول رأس التشغيل إلى الإطار الرئيسى الذى يحتوى على هذا الكود. فإذا قمت على سبيل المثال بوضع الإجراء **stop()** داخل أحد الأطارات الرئيسية، يتم إيقاف تنفيذ الفيلم بمجرد وصول رأس التشغيل إلى هذا الإطار.

- مع أزرار الأمر وهي الرموز ذات النوع **Button**. قم بتنشيط الزر على مسرح العمل ثم قم بإدخال الكود إلى لوحة الإجراءات. ويتم تنفيذ الكود المصاحب لزر الأمر بمجرد نقر المستخدم لهذا الزر أثناء تشغيل الفيلم.
- مع المقطوعات الفيلمية وهي الرموز ذات النوع **MovieClip**. قم بتنشيط المقطوعة الفيلمية على مسرح العمل ثم قم بإدخال الكود.
- مع المكونات **Components** وهي أدوات التحكم كزر الأمر ومربع الاختيار وزر الاختيار والقائمة المنسدلة وغيرها. قم بتنشيط الأداة على مسرح العمل ثم قم بإدخال الكود المصاحب إلى لوحة الإجراءات مع تحديد نوع الحدث المستخدم.

استخدام لوحة الإجراءات

- يمكنك تعيين كود **ActionScript** داخل **Flash** من خلال إحدى اللوحات الهامة الموجودة داخل البرنامج وهي لوحة الإجراءات **Actions Panel**. لإظهار أو تكبير أو تنشيط لوحة الإجراءات، استخدم إحدى الطرق التالية:
- افتح قائمة **Window** من شريط القوائم ثم اختر **Development Panels** من القائمة الفرعية الناتجة ومنها اختر **Actions**.
 - انقر شريط عنوان اللوحة **Actions** بالجزء السفلي من نافذة التصميم.
 - اضغط مفتاح **F9** من لوحة المفاتيح.
- وفي جميع الحالات تظهر النافذة مكبرة أو واضحة أو في مقدمة اللوحات الأخرى بحيث يمكنك التعامل معها وتعريف كود **ActionScript** من خلالها. (انظر شكل ١٤-١) انقر داخل الإطار أو الزر أو المقطوعة الفيلمية التي ترغب في وضع كود **ActionScript** بداخلها ثم انتقل إلى لوحة الإجراءات وقم بإدخال الكود المناسب إلى الجانب الأيمن من اللوحة.



شكل ١-١٤ ١-١٤ يمكنك من خلال لوحة الإجراءات تعيين كود ActionScript

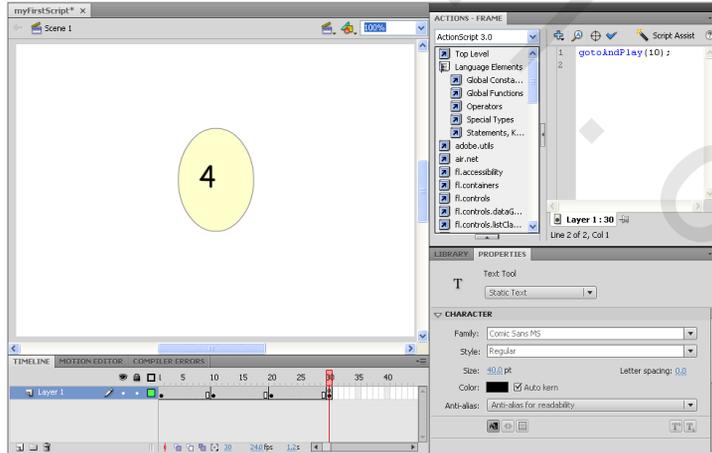
وتحتوى لوحة الإجراءات كما هو واضح من الشكل السابق على العناصر التالية:

- مربع الإجراءات ويوجد بالجانب الأيسر العلوى من اللوحة ويحتوى على قائمة بالإجراءات المختلفة الموجودة داخل ActionScript مقسمةً في عدد من المجموعات.
 - منطقة التنقل وتوجد بالجانب الأيسر السفلى من اللوحة وتحتوى على أجزاء الفيلم المختلفة التى تحتوى على كود بحيث يمكنك التنقل بسهولة بين هذه الأجزاء لعرض الكود المصاحب.
 - منطقة الكود وتوجد بالجانب الأيمن من اللوحة ومن خلالها يتم إدخال الكود سواءً بكتابته مباشرةً أو باختيار الإجراء المناسب من مربع الإجراءات بالجانب الأيسر.
- للتعرف على كيفية استخدام لوحة الإجراءات فى تعيين كود ActionScript داخل الفيلم، تابع معنا الخطوات الآتية:

١. قم بإنشاء فيلم جديد باسم مناسب وليكن myFirstScript.fl
٢. قم بإضافة ثلاثة إطارات أساسية للطبقة الموجودة بسطر الوقت. لأداء ذلك، انقر داخل كل إطار ثم اضغط مفتاح F7 من لوحة المفاتيح.

٣. استخدم أدوات الرسم المختلفة الموجودة بلوحة الأدوات في رسم شكل مميز داخل كل إطار من هذه الإطارات. فعلى سبيل المثال، قم برسم دائرة تحتوي على رقم يعبر عن ترتيب الإطار.
 ٤. قم الآن بتجربة الفيلم سواءً داخل نافذة التطوير بضغط مفتاح Enter أو من خلال مشغل Flash الداخلي بضغط الاختصار **Ctrl+Enter**، يبدأ الفيلم في العمل بحيث تظهر محتويات الإطار الرئيسي الأول ثم محتويات الإطار الرئيسي الثاني ثم محتويات الإطار الرئيسي الثالث وأخيراً محتويات الإطار الرئيسي الرابع.
 ٥. يمكنك تغيير سلوك الفيلم من خلال كود **ActionScript** كما يلي:
- على فرض أننا نرغب في توجيه الفيلم من الإطار الرئيسي الرابع إلى الإطار الرئيسي الثاني وهكذا إلى أن يقوم المستخدم بإغلاق الفيلم. لأداء ذلك، انقر الأطار الرئيسي الرابع وتأكد من ظهور لوحة الإجراءات وإلا قم بإظهارها باستخدام أي من الطرق الثلاثة التي ذكرناها منذ قليل ثم قم بإدخال العبارة التالية إلى منطقة الكود بالجزء الأيمن من لوحة الإجراءات (انظر شكل ١٤-٢):

gotoAndPlay(10);



شكل ١٤-٢ إدخال الكود إلى الإطار من خلال لوحة الإجراءات المصاحبة

وهذا على فرض أن الإطار الرئيسي الثاني يقع بالإطار رقم ١٠ بالطبقة. قم الآن بتشغيل الفيلم، تلاحظ ظهور محتويات الإطار الرئيسي الأول ثم محتويات الإطار الرئيسي الثاني ثم محتويات الإطار الرئيسي الثالث وما أن نأتي إلى الإطار الرئيسي الرابع، يتم الذهاب مباشرةً إلى الإطار الرئيسي الثاني وهكذا إلى أن يتم إغلاق الفيلم وبالتالي تظهر دائماً محتويات الإطارين الرئيسيين الثاني والثالث فقط.

- على فرض أننا نرغب في تكرار نفس العملية السابقة ولكن بتوجيه الفيلم إلى الإطار الرئيسي الثاني ثم إيقافه. لأداء ذلك، انقر الإطار الرئيسي الرابع ثم قم بإدخال العبارة التالية إلى منطقة الكود بدلاً من العبارة السابقة:

gotoAndStop(10);

قم الآن بتشغيل الفيلم، تلاحظ ظهور محتويات الإطار الرئيسي الأول ثم محتويات الإطار الرئيسي الثاني ثم محتويات الإطار الرئيسي الثالث وما أن نأتي إلى الإطار الرئيسي الرابع، يتم الذهاب مباشرةً إلى الإطار الرئيسي الثاني وإيقاف الفيلم.

- على فرض أننا نرغب في إيقاف الفيلم بمجرد الوصول إلى الإطار الرئيسي الأخير وعدم تكرار التشغيل مرةً أخرى، لأداء ذلك، انقر الإطار الرئيسي الرابع ثم قم بإدخال العبارة التالية إلى منطقة الكود بدلاً من العبارة السابقة:

stop();

قم الآن بتشغيل الفيلم، تلاحظ ظهور محتويات الإطار الرئيسي الأول ثم محتويات الإطار الرئيسي الثاني ثم محتويات الإطار الرئيسي الثالث ثم محتويات الإطار الرئيسي الرابع ثم يتوقف الفيلم.

قواعد كتابة الكود

لاشك أن لكل مبرمج أسلوبه الخاص في كتابة الكود. ولكن هناك العديد من القواعد والأعراف التي ينبغي اتباعها وعدم تجاهلها حتى نحصل على كود مقروء يستطيع الآخرون قراءته وفهمه بسهولة وتستطيع أنت كذلك فهمه وقراءته إذا ما تركته لفترات طويلة ثم عدت إليه مرةً أخرى لتعديله أو تطويره. وللحصول على كود مقروء، يجب مراعاة

المسافات بين الكلمات والهوامش والمسافات البادئة ومراعاة أيضاً متى يتم كتابة الحرف بحالة كبيرة **Capital** أو حالة صغيرة **Small**، إلى جانب استخدام التعليقات لتوضيح المهام المختلفة للكود الذي تقوم بكتابته وأهداف كتابة هذا الكود.

الحفاظ على المسافات وحالة الأحرف

تتميز لغة **ActionScript** بالمرونة الشديدة في التعامل مع المسافات بين الكلمات والمسافات البادئة وحالة الأحرف ما عدا بعض الاستثناءات البسيطة. فعلى سبيل المثال، بالنسبة لمفسر **ActionScript**، تتكافأ الدالة التالية:

```
function addTwo(a, b) {return a+b;}
```

تماماً مع هذه الدالة:

```
function addTwo(a, b) {  
    return a+b;  
}
```

إلا أن الحالة الثانية للدالة أفضل من الحالة الأولى نظراً لاستخدام المسافات ووجود مسافة بادئة قبل العبارة **return** الموجودة داخل الدالة مما يعمل على سهولة قراءتها وفهمها مقارنةً بالحالة الأولى.

إضافة التعليقات إلى الكود

يمكنك كتابة التعليقات داخل **ActionScript** باستخدام زوج من الشرط المائلة الأمامية (**//**) هكذا مثلاً:

```
// هذا تعليق
```

حيث تستخدم هذه الصيغة في حالة التعليقات البسيطة التي لا تتعدى السطر الواحد. أما في حالة الرغبة في كتابة التعليقات الطويلة، فيمكنك وضع التعليق بين رمزي الفتح **/*** ورمزي الإغلاق ***/** هكذا مثلاً:

/* يستخدم المبرمجون التعليقات لتوضيح بعض المعلومات عن الكود الذي يقومون بكتابته حتى يتمكنوا من تحديثه فيما بعد بسهولة تامة، وحتى يسهل على الآخرين قراءة هذا الكود وفهمه في أي وقتٍ آخر. لذا اجتهد دائماً في كتابة التعليقات المناسبة. ***/**

إلا أن البعض الآخر يفضل وضع التعليق الطويل في عدة سطور هكذا مثلاً:

```

////////////////////////////////////
//يستخدم المبرمجون التعليقات لتوضيح بعض المعلومات عن الكود الذى يقومون بكتابته
حتى يتمكنوا من تحديثه فيما بعد بسهولة تامة، وحتى يسهل على الآخرين قراءة هذا //
الكود وفهمه في أى وقتٍ آخر. لذا اجتهد دائماً في كتابة التعليقات المناسبة. //
////////////////////////////////////
ولا يشترط وضع التعليق في سطر مستقل أو في بداية السطر وإنما يمكنك وضعه أيضاً بعد
كتابة الكود هكذا مثلاً:

```

تقوم هذه العبارة بتخصيص الرقم ١٠ للمتغير `x` // `x = 10;`
 ويتم إظهار التعليقات داخل لوحة الإجراءات بلون مخالف للون الكود (اللون الافتراضى
 للتعليقات هو اللون الرمادى الباهت)، لذا يمكنك والآخرين التفرقة بين الكود والتعليق
 بسهولة شديدة بمجرد النظر. أما مفسر `ActionScript` (والذى يقوم بتحويل الفيلم إلى
 ملف قابل للتنفيذ من خلال مشغل `Flash`)، فلا ينظر أساساً لهذه التعليقات أو يعرها أى
 اهتمام.

إنشاء أسماء مقروءة

يحتوى كود `ActionScript` غالباً على أنواع مختلفة من العناصر مثل الدوال والمصفوفات
 والكائنات والمتغيرات والتي يحتاج كل منها إلى اسم مميز يقوم بتعريفها وتمييزها عن بقية
 العناصر الأخرى. ويمكنك حقيقةً إنشاء أسماء سهلة القراءة من خلال عدد من الوسائل،
 والتي رأينا واحدةً منها منذ قليل حينما قمنا بتعريف الدالة `addTwo()`. فكلمة
`addTwo` (عن طريق كتابة الحرف `T` بحالة كبيرة) أفضل في القراءة من كلمة `addtwo`
 (والتي تحتوى على جميع الحروف بحالة صغيرة)، فالحالة الأخيرة ليس لها معنى، أما الحالة
 الأولى فتدل من الوهلة الأولى على أن هذه الدالة تقوم بإضافة (من كلمة `add`) عنصرين
 (من كلمة `Two`)، عرفنا هذه المعلومة دون الدخول في تفاصيل الدالة. ومن المتعارف عليه

داخل Flash بدء جميع الأسماء التي تقوم بإنشائها داخل أفلامك بحروف صغيرة ما عدا التصنيفات **Classes**، فتبدأ عادةً بحروف كبيرة.

استخدام مستكشف الفيلم في الوصول للكود

يمكنك استخدام مستكشف الفيلم **Movie Explorer** في البحث عن كود **ActionScript** الموجود داخل ملفات **Flash** المصدرية ذات الامتداد **FLA**.

للبحث عن الكود من خلال لوحة مستكشف الفيلم، تابع معنا الخطوات الآتية:

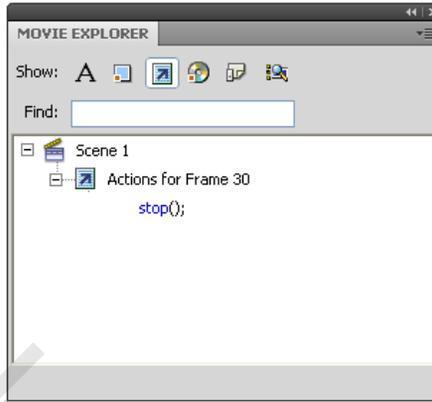
١. قم بإظهار لوحة مستكشف الفيلم إذا لم تكن ظاهرة بالفعل، وذلك باستخدام إحدى الطريقتين التاليتين:

- افتح قائمة **Window** من شريط القوائم ثم اختر **Movie Explorer** من القائمة الفرعية.
- اضغط الاختصار **Alt+F3** من لوحة المفاتيح.

وفي الحالتين يتم إظهار لوحة المستكشف إذا كانت مخفية أو إخفاؤها إذا كانت ظاهرة.

٢. من الجزء **Show** بالجزء العلوي من اللوحة، قم بتنشيط الأزرار التي ترغب في عرض كائناتها المصاحبة داخل المستكشف. فمثلاً الرمز **A** لعرض أو إخفاء النصوص، بينما الرمز  لعرض أو إخفاء الرموز (**Buttons** أو **Graphics** أو **Movie Clip**).

لإظهار كود **ActionScript** المصاحب لأحد الأزرار أو إحدى المقطوعات الفيلمية، اختر الزر أو المقطوعة الفيلمية ثم انقر زر التوسيع + المجاور حتى تصل إلى كود **ActionScript** الذي تبحث عنه. (انظر شكل ١٣-٣).



شكل ١٣-٣ استخدام مستكشف الفيلم في الوصول لكود ActionScript

استخدام ActionScript داخل الفيلم

قمنا فيما سبق من فصول الكتاب بإنشاء المشاهد (أو الصفحات) المختلفة للفيلم بما في ذلك صفحات المستوى الثاني التي تنفرع من الصفحة الرئيسية للفيلم، كما قمنا أيضاً بإنشاء الأزرار المستخدمة لفتح هذه الصفحات إلا أننا لم نقم بتعيين أى أحداث لهذه الأزرار، حيث يتم تعيين هذه الأحداث من خلال كود ActionScript. فباستخدام الأوامر المختلفة لهذه اللغة يستطيع المستخدم التنقل بحرية تامة بين الصفحات المختلفة للفيلم، كما يمكنه التوقف أو الاستمرار في أى لحظة كما رأينا منذ قليل.

ربط الأزرار بالإجراءات

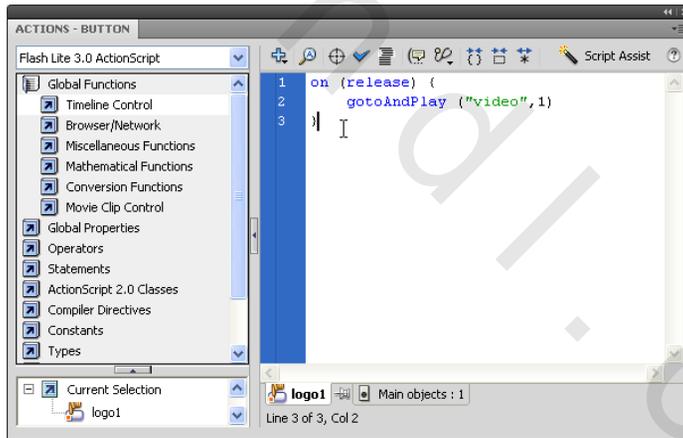
الهدف من إنشاء الأزرار في الصفحات المختلفة هو إظهار الصفحة المصاحبة بمجرد نقر المستخدم لأحد تلك الأزرار. إذاً يجب إنشاء رابط بين الزر والصفحة التي يقوم بفتحها وذلك من خلال كود لغة ActionScript.

يمكنك من خلال كود ActionScript إنشاء الإجراء `GotoAndPlay()` الذي يقوم بربط كل زر بالصفحة المصاحبة. دعنا نبدأ بالصفحة الرئيسية `Home Page`، تابع معنا الخطوات الآتية:

١. قم بتنشيط الصفحة الرئيسية باستخدام لوحة المشاهد أو القائمة المنسدلة من زر تحرير المشاهد الموجود بالركن الأيمن العلوي من نافذة مساحة العمل.
٢. انقر الزر **logo** لتنشيطه على مسرح العمل ثم افتح قائمة **Window** من شريط القوائم ثم اختر **Actions** من القائمة المنسدلة، تظهر لوحة الإجراءات **Actions** المصاحبة للزر.
٣. قم بإدخال الكود التالي إلى منطقة الكود بالجانب الأيمن من لوحة الإجراءات (انظر شكل ١٤-٤):

```
on (release) {  
gotoAndPlay("Video", 1);  
}
```

وهذا الكود معناه، عند نقر الزر **Logo** ثم تحرير زر الفأرة (الحدث **release**)، اذهب إلى الإطار الأول بالمشهد **Video** وقم بتشغيله.



شكل ١٤-٤ ربط الزر **logo** بالصفحة المصاحبة باستخدام كود **ActionScript**

لاختبار الفيلم الآن، تابع معنا الخطوات الآتية:

١. افتح قائمة **Control** من شريط القوائم ثم قم بتنشيط الخيار **Enable Simple Buttons**. قم أيضاً بتنشيط الخيار **Play All Scenes** داخل نفس القائمة.

٢. انقر زر **Logo** بزر الفأرة الأيسر، فإذا كنت قد أدخلت كود **ActionScript** بطريقة صحيحة، يتم الذهاب مباشرة إلى المشهد **Video** وتشغيله بدءاً من الإطار الأول، حيث يستمر الفيلم بعد ذلك في التشغيل حتى يصل إلى المشهد الأخير وذلك لأنك قمت بتنشيط الخيار **Play All Scenes**. لا تقلق، سنتغلب على هذه المشكلة بعد قليل.

باتباع نفس الخطوات السابقة يمكنك إنشاء الروابط بين الأزرار الأخرى الموجودة على مسرح العمل سواءً بالصفحة الرئيسية أو الصفحات الفرعية الأخرى.

يمكنك استخدام عمليات النسخ واللصق المعروفة لنسخ كود الأزرار المتشابهة بدلاً من إعادة تعيينها مع كل زر توفيراً للوقت والجهد.

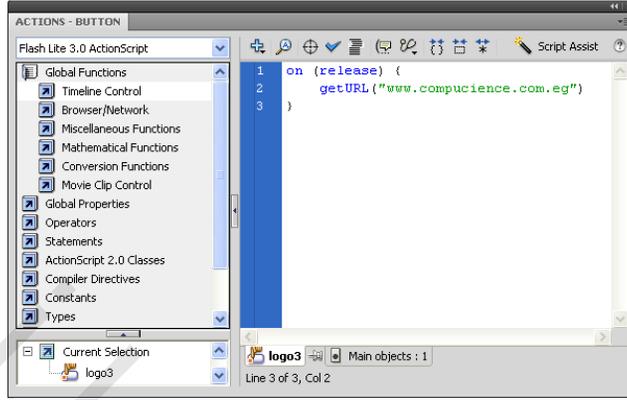


استخدام الإجراء `getURL()`

يمكنك استخدام الإجراء `getURL()` بنفس طريقة استخدام الإجراء `gotoAndPlay()` ويستخدم في حالة الرغبة في فتح صفحة ويب داخل أحد مواقع الإنترنت. للتعرف على طريقة استخدام الإجراء `getURL()`، تابع معنا الخطوات الآتية:

١. انقر الزر الذي ترغب في ربطه بإحدى صفحات أو مواقع الويب لتنشيطه على مسرح العمل ثم افتح قائمة **Window** من شريط القوائم ثم اختر **Actions** من القائمة المنسدلة (أو اضغط مفتاح **F9** من لوحة المفاتيح)، تظهر لوحة الإجراءات.
٢. قم بإدخال الكود التالي إلى منطقة الكود بالجانب الأيمن بلوحة الإجراءات (انظر شكل ١٤-٥):

```
on (release) {
    getURL("www.compuscience.eg.com");
}
```



شكل ١٤-٥ فتح أحد مواقع الويب باستخدام الإجراء gotoURL()

ربط الإطارات بالإجراءات

بعد أن قمنا بربط كل زر من الأزرار بالصفحة المصاحبة، نريد الآن إضافة المزيد من الكود للتحكم في مدى تقدم رأس التشغيل داخل الفيلم. وقبل أن نبدأ في إضافة هذا الكود، افتح قائمة **Control** من شريط القوائم ثم اختر **Enable Simple Frame Actions** لتنشيطه حتى يتم تمكين الإجراءات الخاصة بالإطارات داخل بيئة تشغيل Flash.

استخدام الإجراء Stop

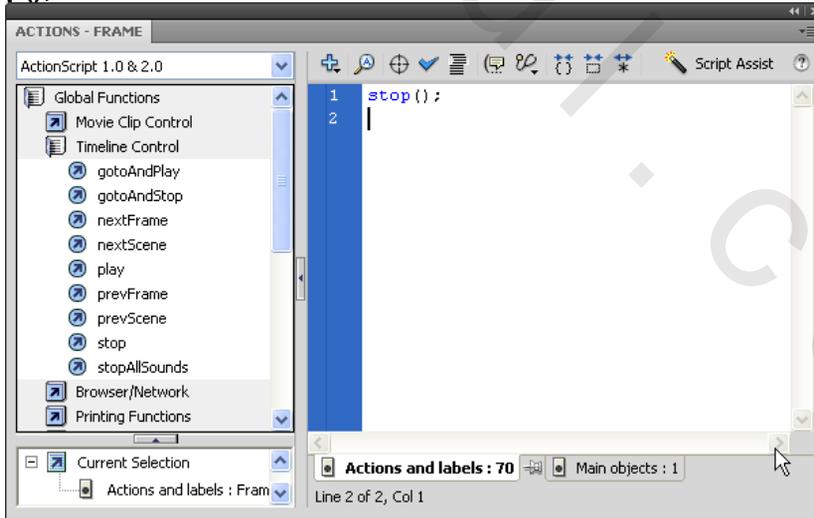
إذا أردت إيقاف تنفيذ الفيلم بانتهاء الصفحة الرئيسية، يجب أن تقوم بإضافة الإجراء **Stop** بالإطار الأخير من مشهد الصفحة الرئيسية. ومن الأشياء المتعارف عليها دائماً إنشاء طبقة جديدة تحتوي على الأحداث الخاصة بتشغيل الإطارات حتى يمكنك صيانتها وحل مشاكلها بسهولة تامة. انقر الطبقة العليا داخل منطقة الطبقات في سطر الوقت ثم انقر زر إضافة طبقة جديدة وأعد تسمية الطبقة الجديدة إلى **Actions and Labels** (انظر شكل ١٤-٦).



شكل ١٤-٦ إضافة الطبقة الجديدة إلى منطقة الطبقات داخل نافذة سطر الوقت لإنشاء الكود اللازم لإيقاف الفيلم بمجرد الوصول إلى الإطار الأخير بالصفحة الرئيسية، تأكد من أنك تقبع الآن داخل مشهد الصفحة الرئيسية من الفيلم ثم تابع معنا الخطوات الآتية:

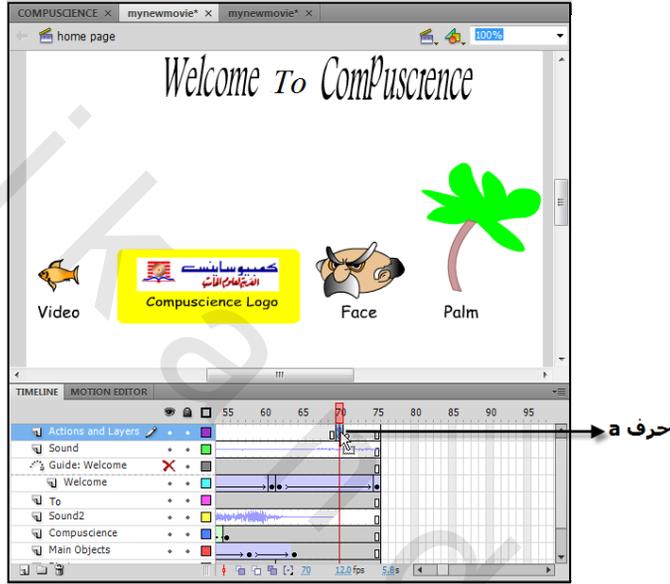
١. انقر الإطار رقم ٧٠ من الطبقة الجديدة **Actions and Labels** لتتشيته ثم اضغط مفتاح **F6** لإضافة إطار رئيسي جديد.
٢. تأكد من ظهور لوحة الإجراءات وإلا اضغط مفتاح **F9** من لوحة المفاتيح لإظهارها ثم قم بإضافة الإجراء **stop()** هكذا (انظر شكل ١٤-٧):

stop();



شكل ١٤-٧ إضافة الإجراء **stop()** لإيقاف تشغيل الفيلم بمجرد الوصول إلى نهاية الصفحة الرئيسية

٣. قم بإلقاء نظرة خاطفة على نافذة سطر الوقت، تلاحظ ظهور الحرف a بالإطار الرئيسي بالإطار رقم ٧٠ داخل طبقة **Actions and Labels** (انظر شكل ١٤-٨) دلالة على أن هناك إجراء ما مرتبط بهذا الإطار.



شكل ١٤-٨ ظهور الحرف a داخل الإطار المرتبط بمحدث معين

٤. قم بتشغيل الفيلم بعد أن تتأكد من تنشيط الخيارات **Enable Simple Frame Actions** و **Enable Simple Buttons** حتى يمكنك التعامل مع هذه الأزرار.

إضافة الإجراء **Stop** إلى الصفحات الفرعية

إذا قمت الآن بتشغيل الفيلم، تلاحظ البدء بتشغيل الصفحة الرئيسية إلى أن تصل إلى الإطار الأخير بالصفحة وحينئذٍ تقف عملية العرض. فإذا قمت بنقر أحد الأزرار الموجودة على الصفحة، يتم فتح الصفحة المصاحبة ثم الصفحات التابعة داخل الفيلم بسرعة شديدة حتى ترجع إلى الصفحة الرئيسية مرةً أخرى وذلك لأننا لم نقم بعد بتعيين إجراءات إيقاف هذه الصفحات بعد تنفيذها. لأداء ذلك، تابع معنا الخطوات الآتية:

١. اختر صفحة الشعار من لوحة المشاهد أو من القائمة المنسدلة عن الزر الموجود بالركن الأيمن العلوى من نافذة سطر الوقت.
 ٢. قم بإضافة طبقة جديدة كى تحتوى على الإجراءات والأسماء الموجودة بالصفحة كما سبق ثم أعد تسمية الطبقة إلى **Actions and Labels**.
 ٣. انقر الإطار الرئيسى الموجود بالإطار الأول من الطبقة الجديدة لتنشيطه ثم اضغط مفتاح **F9** من لوحة المفاتيح لفتح لوحة الإجراءات إذا لم تكن مفتوحة بالفعل.
- يقوم **Flash** تلقائياً بإضافة إطار رئيسى إلى الطبقة الجديدة، لذا فلست فى حاجة إلى إضافة الإطار الرئيسى مرةً أخرى
٤. قم بإضافة الإجراء **stop()** إلى منطقة الكود بالجزء الأيمن من لوحة الإجراءات كما سبق.
 ٥. قم بإعادة دوران الفيلم كى يبدأ من البداية ثم قم بتشغيله، تلاحظ توقف الفيلم بمجرد الوصول إلى الإطار الأخير داخل مشهد الصفحة الرئيسية. انقر زر الشعار، تلاحظ فتح صفحة الشعار المصاحبة ثم التوقف بدلاً من الاستمرار فى عرض المشاهد (الصفحات) التالية كما كان من قبل.
 ٦. قم بتكرار الخطوات السابقة مع كل صفحة من صفحات المستوى الثانى.
 ٧. قم بإعادة تشغيل الفيلم لترى أنك الآن قادر على التحكم التام فى طريقة عرض الفيلم.

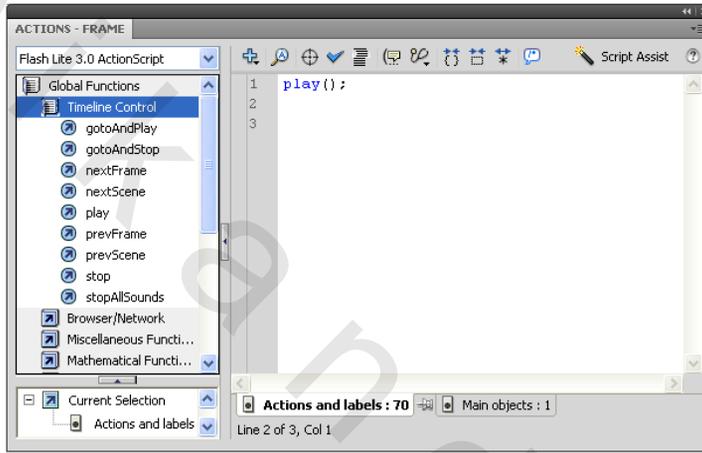
استخدام إجراء التشغيل **Play()**

- يمكنك استخدام الإجراء **play()** بنفس طريقة استخدام الإجراءات السابقة، حيث يمكنك استخدام هذا الإجراء فى أى إطار من الإطارات لتشغيل مشهد أو إطار أو مقطوعة صوتية فضلاً عن أنه يتحكم بشكل كبير مثل الإجراء **stop()** فى طريقة وترتيب عرض الفيلم.
- لإضافة الإجراء **play()** على أحد الإطارات، تابع معنا الخطوات الآتية:
١. قم بإضافة إطار رئيسى إلى الإطار الذى ترغب فى استخدام الإجراء معه.

الفصل الرابع عشر: أساسيات ActionScript

٢. انقر الإطار الرئيسي لتنشيطه ثم اضغط مفتاح F9 من لوحة المفاتيح لفتح لوحة الإجراءات إذا لم تكن مفتوحة بالفعل.
٣. قم بإضافة الإجراء `play()` إلى منطقة الكود بالجزء الأيمن من لوحة الإجراءات هكذا (انظر شكل ١٤-٩):

`play();`



شكل ١٤-٩ لا يحتوي الإجراء `play()` على أي معاملات

