11. DISEÑO DE PAGINAS WEB.

11.4 PAGINA WEB UTILIZANDO MULTIMEDIA.

|  |  |  |  |
| --- | --- | --- | --- |
| [1. Introducción](http://www.aulaclic.es/flash8/t_1_1.htm) [2. Mi primera animación Flash](http://www.aulaclic.es/flash8/t_2_1.htm) [3. El Entorno de Trabajo](http://www.aulaclic.es/flash8/t_3_1.htm) [4. Dibujar](http://www.aulaclic.es/flash8/t_4_1.htm)  [5. Trabajar con Textos](http://www.aulaclic.es/flash8/t_5_1.htm)  [6. Sonidos](http://www.aulaclic.es/flash8/t_6_1.htm) | [7. Trabajar con Objetos](http://www.aulaclic.es/flash8/t_7_1.htm) [8. Las Capas](http://www.aulaclic.es/flash8/t_8_1.htm) [9. Símbolos](http://www.aulaclic.es/flash8/t_9_1.htm) [10. Gráficos](http://www.aulaclic.es/flash8/t_10_1.htm) [11. Clips de película](http://www.aulaclic.es/flash8/t_11_1.htm) [12. Botones](http://www.aulaclic.es/flash8/t_12_1.htm) | [13. Animaciones de Movimento](http://www.aulaclic.es/flash8/t_13_1.htm) [14. Transformaciones de Forma](http://www.aulaclic.es/flash8/t_14_1.htm) [15. Efectos sobre Animaciones](http://www.aulaclic.es/flash8/t_15_1.htm) [16. Generar y Publicar Películas](http://www.aulaclic.es/flash8/t_16_1.htm)  [17. Introducción a ActionScript 2.](http://www.aulaclic.es/flash8/t_17_1.htm) [18. Ejemplos de uso de ActionScript 2.](http://www.aulaclic.es/flash8/t_18_1.htm) | [19. Navegación](http://www.aulaclic.es/flash8/t_19_1.htm) [20. Formularios](http://www.aulaclic.es/flash8/t_20_1.htm) [21. Los Filtros](http://www.aulaclic.es/flash8/t_21_1.htm) [22. Vídeo](http://www.aulaclic.es/flash8/t_22_1.htm)  [23. Juegos en Flash.](http://www.aulaclic.es/flash8/t_23_1.htm) [24. Animaciones avanzadas](http://www.aulaclic.es/flash8/t_24_1.htm) |

|  |  |
| --- | --- |
| ¿Qué es FLASH? |  |

|  |  |
| --- | --- |
| Flash 8 es una potente herramienta creada por Macromedia que ha superado las mejores expectativas de sus creadores.  Inicialmente Macromedia Flash fue creado con el objeto de realizar animaciones vistosas para la web, así como para crear GIFs animados.  Los motivos que han convertido a Flash 8 en el programa elegido por la mayoría de los diseñadores web profesionales y aficionados son varios. Veamos pues, porque es interesante Flash 8.  Desde la web de [Macromedia](http://www.macromedia.com) te puedes descargar una versión de evaluacion de Flash 8 válida para 30 días.  Los logotipos de Flash son propiedad de Macromedia. |  |

|  |  |
| --- | --- |
| ¿Por qué usar FLASH 8? |  |

|  |
| --- |
| Las posibilidades de Flash son extraordinarias, cada nueva versión ha mejorado a la anterior, y el actual Flash 8 no ha sido menos. Aunque su uso más frecuente es el de crear animaciones (a lo largo de este curso veremos lo sencillo que puede resultar) sus usos son muchos más. Son tantos, que todos los diseñadores web deberían saber utilizar Flash.  Flash ha conseguido hacer posible lo que más se echa en falta en Internet: Dinamismo, y con dinamismo no sólo nos referimos a las animaciones, sino que Flash permite crear aplicaciones interactivas que permiten al usuario ver la web como algo atractivo, no estático (en contraposición a la mayoría de las páginas, que están realizadas empleando el lenguaje HTML). Con Flash podremos crear de modo fácil y rápido animaciones de todo tipo.  Flash es fácil de aprender, tiene un entorno amigable que nos invita a sentarnos y pasar horas y horas creando lo que nos dicte nuestra imaginación, pero esto no es suficiente para ser el preferido por los diseñadores profesionales... ¿Entonces qué es? |

|  |  |
| --- | --- |
| De Flash MX 2004 a Flash 8 |  |

|  |
| --- |
| Hay compañías que lanzan mejoras en sus productos por pura necesidad económica; Cuando sucede esto, los usuarios lo notamos rápidamente por las pocas mejoras ofrecidas por las nuevas versiones. No es este el caso de Flash 8, que continua con la costumbre de Macromedia de aplicar considerables mejoras a las nuevas versiones de sus productos ...  Si pensabais que Flash MX 2004 era ya insuperable, no os perdais las mejoras que aporta Flash 8: Mejoras en cuanto a facilidad de manejo, mayor potencia gráfica y de integración con programas de edición de imágenes, facilidad para importar vídeo, posibilidad de emular tus películas dirigidas a dispositivos móviles y para los menos avanzados, se recupera el asistente de ActioScript que había desaparecido en la versión anterior... Analicemos estas ventajas y otras muchas con un poco más de detalle:    **http://www.aulaclic.es/flash8/comunes/orangeball.gifDiseños más atractivos:** Flash 8 permite el uso de efectos visuales que nos facilitarán la creación de animaciones, presentaciones y formularios más atractivos y profesionales, así mismo, pone a nuestra disposición mecanismos para hacer este trabajo más cómodo y rápido, tales como la existencia de filtros y modos de mezcla añadidos en esta versión.  **http://www.aulaclic.es/flash8/comunes/orangeball.gifOptimización de fuentes:** Incorpora también opciones de legibilidad para fuentes pequeñas, haciendo la lectura de nuestros textos más agradables y de alta legibilidad. Además de poder modificar la optimización, Flash permite también la selección de configuraciones preestablecidas para textos dinámicos y estáticos.  **http://www.aulaclic.es/flash8/comunes/orangeball.gifBibliotecas integradas:** Ahora podemos buscar rápidamente cualquier objeto existente en nuestras películas, navegando por las bibliotecas de todos los archivos abiertos desde un único panel.  **http://www.aulaclic.es/flash8/comunes/orangeball.gifMayor potencia de animación:** Flash 8 permite un mayor control de las interpolaciones habilitando un modo de edición desde el que se podrá modificar independientemente la velocidad en la que se apliquen los diferentes cambios de rotación, forma, color, movimiento, etc, de nuestras interpolaciones.  **http://www.aulaclic.es/flash8/comunes/orangeball.gifMayor potencia gráfica:** Evita la repetición innecesaria de la representación de objetos vectoriales simplemente señalando un objeto como mapa de bits. Aunque el objeto se convierta al formato de mapa de bits, los datos vectoriales se mantienen tal cual, con el fin de que, en todo momento, el objeto pueda convertirse de nuevo al formato vectorial.  **http://www.aulaclic.es/flash8/comunes/orangeball.gifMejoras en la importación de vídeo:** Para facilitar el resultado con formatos de vídeo, Flash 8 incluye un códec independiente de calidad superior capaz de competir con los mejores códecs de vídeo actuales con un tamaño de archivo mucho más pequeño. Además de una gran posibilidad de revestimientos para los controles de éste en nuestra película.  **http://www.aulaclic.es/flash8/comunes/orangeball.gifCompatiblidad Metadatos:** Incluye tus SWF en buscadores de internet con la nueva característica de definición de archivo con un título, una descripción y/o palabras clave para que los motores de búsqueda reflejen con más precisión el contenido representado por el archivo.  **http://www.aulaclic.es/flash8/comunes/orangeball.gifEmulador para dispositivos móviles:** Prueba tus películas destinadas a dispositivos móviles compatibles con Flash Lite con el nuevo emulador que incorpora Flash 8. Podrás probar tus películas de un modo eficiente antes de publicarlas.  **http://www.aulaclic.es/flash8/comunes/orangeball.gifAsistente de ActionScript:** El Asistente de ActionScript ha vuelto. Fue eliminado en la versión anterior, pero se ha vuelto a recuperar, y de forma mejorada, en esta última. Ahora ActionScript al alcance de tu mano. |

|  |  |
| --- | --- |
| **Unidad 2.**  Mi primera Animación Flash |  |

|  |  |
| --- | --- |
| Comenzando |  |

|  |  |
| --- | --- |
| Una de las características principales de Flash 8 es su sencillez, esta sencillez en su utilización permite crear animaciones de un modo efectivo y rápido.  Pongámonos en situación, supongamos que queremos crear una animación en la que una pelota realice un movimiento de "bote", es decir, que baje y suba. Puede parecer un trabajo de horas, pero nada más lejos de la realidad. Veamos cómo Flash nos lo pone fácil. | http://www.aulaclic.es/flash8/graficos/pelota_azul1.jpg |

|  |  |
| --- | --- |
| Realizando la Animación | [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/mi_primera_animacion.htm');) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| En este apartado no pretendemos enseñar cómo crear una animación Flash (pues eso se verá más adelante), el objetivo es que entendamos cómo crea Flash las animaciones y de que modo nos facilita el trabajo.  A primera vista, lo lógico parece ser dibujar la pelota en cada instante del movimiento, de modo que cuantos más instantes dibujemos, más real parecerá el movimiento. Pues bien, con Flash bastará con crear 3 fotogramas: en el primero dibujaremos la pelota en el instante inicial (arriba del todo), en el segundo se dibujará la pelota en el momento de tocar el suelo y en el tercero la pelota volverá a estar arriba (en realidad este tercer fotograma no hará falta crearlo, pues bastará con copiar el primero). Por tanto, vemos que el trabajo más largo (dibujar los objetos) ya está hecho.   |  |  | | --- | --- | | **Unidad 3.**  El Entorno de Trabajo (I) |  |  |  |  | | --- | --- | | La Interfaz de Flash 8 |  |  |  |  | | --- | --- | | Flash 8 cuenta con un entorno o interfaz de trabajo de lo más manejable e intuitiva. Además, tiene la ventaja de que es similar a la de otros programas de Macromedia (Dreamweaver, Freehand, Director...), todo esto hace más fácil aprender Flash y más rápido su manejo y dominio. Esto es lo que nos encontraremos al abrir Flash 8 por primera vez: | | | http://www.aulaclic.es/flash8/graficos/entornomx2004.jpg |  | | *A lo largo de todo el curso trabajaremos con la versión de evaluación de Flash 8 en castellano. Podeis descargar esta versión de forma gratuita desde* [*AQUÍ*](http://www.macromedia.com/es/software/trial_download/)  En la imagen se puede ver la interfaz que nos encontraremos la primera vez que abramos el programa Flash. Flash recordará nuestras preferencias y abrirá el programa tal y como lo dejamos la última vez que lo utilizamos.  En esta imagen faltan algunos menús. Todos se irán viendo a lo largo de este curso aunque no aparezcan en esta imagen, que sólo pretende mostrar las partes principales de la interfaz de Flash. Veamos cuáles son: |  |  |  |  | | --- | --- | | La Barra de Menús |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gifLa **Barra de Menús** http://www.aulaclic.es/flash8/graficos/barra_menusmx2004.giftiene como propósito facilitar el acceso a las distintas utilidades del programa. Es similar a la de cualquier otro programa de diseño web o gráfico, aunque tiene algunas particularidades. Veamos los principales **Submenús** a los que se puede acceder:  **Archivo:** Permite crear nuevos archivos, abrirlos, guardarlos... Destaca la potencia de la utilidad **Importar** que inserta en la película actual casi todo tipo de archivos (sonidos, vídeo, imágenes e incluso otras películas Flash), o la de **Configuración de Publicación** desde donde se pueden modificar las características de la publicación. También permite configurar la impresión de las páginas, imprimirlas...  **Edición:** Es el clásico menú que te permite **Cortar, Copiar, Pegar...** tanto objetos o dibujos como fotogramas; también permite personalizar algunas de las opciones más comunes del programa.  **Ver:** Además de los típicos Zooms, te permite moverte por los fotogramas y por las escenas. También incluye la posibilidad de crear una cuadrícula y unas guías. Esto se puede seleccionar desde los submenús **Cuadrícula y Guías** desde donde también se pueden configurar sus opciones.  **Insertar:** Te permite insertar objetos en la película, así como nuevos fotogramas, capas, acciones, escenas...  **Modificar:** La opción **Transformar** permite modificar los gráficos existentes en la película, y la opción **Trazar Mapa de Bits** convierte los gráficos en mapas vectoriales (este tema se tratará más adelante). El resto de opciones permite modificar características de los elementos de la animación **Suavizar, Optimizar** o de la propia película **(Capa, Escena ...)**.  **Texto:** Sus contenidos afectan a la edición de texto. Más adelante se tratará en profundidad.  **Comandos:** Permite administrar los Comandos (conjunto de sentencias almacenadas que permiten emular lo que un usuario pueda introducir en el entorno de edición) que hayamos almacenado en nuestra animación, obtener otros nuevos de la página de Macromedia o ejecutar los que ya tengamos.  **Control:** Desde aquí se modifican las propiedades de reproducción de la película. **Reproducir, Rebobinar, Probar Película** ....  **Ventana:** Este menú, además de las opciones clásicas acerca de cómo distribuir las ventanas, incluye accesos directos a **TODOS** los Paneles.  **Ayuda:** Desde aquí podemos acceder a toda la ayuda que nos ofrece Macromedia, desde el manual existente, hasta el diccionario de Action Script, pasando por tutoriales, lecciones guiadas | |

|  |  |
| --- | --- |
| **Unidad 4.** Dibujar y Colorear (I) |  |

|  |  |
| --- | --- |
| El Dibujo en Flash 8 |  |

|  |
| --- |
| Cuando se diseña una página web o una animación el diseñador pasa por muchas fases. Tras la fase de "Qué quiero crear y cómo va a ser" viene (normalmente) la fase de diseño gráfico. Llevar lo que ha producido tu imaginación al papel (en este caso al papel de Flash).  No conviene engañarnos, Flash no es un programa de diseño gráfico, pero su potencia en este ámbito es casi tan grande como la de éstos programas. Vamos a ver cómo emplear cada herramienta de dibujo para sacarle el máximo partido. |

|  |  |
| --- | --- |
| La Barra de Herramientas. Herramientas Básicas. |  |

|  |  |
| --- | --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gifLa **Barra de Herramientas** contiene todas las Herramientas necesarias para el dibujo. Veamos cuáles son las más importantes y cómo se usan:  **Herramienta Selección (flecha):** http://www.aulaclic.es/flash8/graficos/herr_flecha2004.gif. Es la herramienta más usada de todas. Su uso principal es para seleccionar objetos. permite seleccionar los bordes de los objetos, los rellenos (con un sólo clic), los bordes (con doble clic), zonas a nuestra elección... Además, su uso adecuado puede ahorrarnos tiempo en el trabajo.  **Herramienta Línea: http://www.aulaclic.es/flash8/graficos/herr_linea2004.gif**Permite crear líneas rectas de un modo rápido. Las líneas se crean como en cualquier programa de dibujo, se hace clic y se arrastra hasta donde queramos que llegue la línea recta. Una vez creada la podemos modificar sin más que seleccionar situar el cursor encima de los extremos para estirarlos y en cualquier otra parte cercana a la recta para curvarla.  **Herramienta Texto: http://www.aulaclic.es/flash8/graficos/herr_texto2004.gif**Crea un texto en el lugar en el que hagamos clic. Sus propiedades se verán en el tema siguiente.  **Herramienta Óvalo: http://www.aulaclic.es/flash8/graficos/herr_ovalo2004.gif**La herramienta **Óvalo** permite trazar círculos o elipses de manera rápida y sencilla.  http://www.aulaclic.es/flash8/comunes/greenball.gifPara practicar el manejo de esta Herramienta, te aconsejamos realizar [Ejercicio Crear Óvalo](http://www.aulaclic.es/flash8/epp_4_1_1.htm)  http://www.aulaclic.es/flash8/comunes/greenball.gifPara practicar el manejo de esta Herramienta, te aconsejamos realizar [Ejercicio Dar color a un Óvalo](http://www.aulaclic.es/flash8/epp_4_1_2.htm)  **Herramienta Rectángulo: http://www.aulaclic.es/flash8/graficos/herr_rectangulo2004.gif**Su manejo es idéntico al de la Herramienta Óvalo, tan solo se diferencian en el tipo de objetos que crean.  **Herramienta Lápiz:** http://www.aulaclic.es/flash8/graficos/herr_lapiz2004.gifEs la primera Herramienta de dibujo propiamente dicho. Permite dibujar líneas con la forma que decidamos, modificando la forma de estas a nuestro gusto. El color que aplicará esta Herramienta se puede modificar, bien desde el Panel **Mezclador de Colores** o bien desde el subpanel **Colores** que hay en la **Barra de Herramientas.**  **Herramienta Brocha:** http://www.aulaclic.es/flash8/graficos/herr_brocha2004.gifSu funcionalidad equivale a la del lápiz, pero su trazo es mucho más grueso. Se suele emplear para aplicar rellenos. Se puede modificar su grosor y forma de trazo.  **Herramienta Cubo de Pintura: http://www.aulaclic.es/flash8/graficos/herr_cubo2004.gif**Permite aplicar rellenos a los objetos que hayamos creado. Al contrario que muchos otros programas de dibujo, no permite aplicar rellenos si la zona no está delimitada por un borde. El color que aplicará esta Herramienta se puede modificar, bien desde el Panel Mezclador de Colores o bien desde el subpanel **Colores** que hay en la **Barra de Herramientas.**  **Herramienta Borrador: http://www.aulaclic.es/flash8/graficos/herr_borrador2004.gif**Su funcionamiento es análogo a la Herramienta Brocha. Pero su función es la de eliminar todo aquello que "dibuje". | http://www.aulaclic.es/flash8/graficos/barra_herramientasmx2004.jpg |

|  |  |
| --- | --- |
| La Barra de Herramientas. Herramientas Avanzadas. |  |

|  |
| --- |
| **Herramienta Lazo:** http://www.aulaclic.es/flash8/graficos/herr_lazo2004.gifSu función es complementaria a la de la Herramienta **Flecha**, pues puede seleccionar cualquier cosa, sin importar la forma, (la Herramienta Flecha sólo puede seleccionar objetos o zonas rectangulares o cuadradas). En contrapartida, la Herramienta **Lazo** no puede seleccionar rellenos u objetos (a menos que hagamos la selección a mano).  Al seleccionar esta Herramienta, en el Panel Opciones aparecen estas imágenes: http://www.aulaclic.es/flash8/graficos/herr_varitas.gif. Esto, es la Herramienta Varita Mágica, tan popular en otros programas de dibujo. Permite hacer selecciones según los colores de los objetos. El tercer dibujo que aparece es este: http://www.aulaclic.es/flash8/graficos/herr_lazo2_2004.gifpermite hacer selecciones poligonales.  **Herramienta Pluma: http://www.aulaclic.es/flash8/graficos/herr_pluma2004.gif**Crea polígonos (y por tanto rectas, rectángulos...) de un modo sencillo. Mucha gente encuentra esta herramienta complicada, aunque es una de las más potentes que ofrece Flash. Su empleo consiste en hacer clic en los lugares que queramos definir como vértices de los polígonos, lo que nos asegura una gran precisión. Para crear curvas, hay que señalar los puntos que la delimitan y posteriormente trazar las tangentes a ellas. Con un poco de práctica se acaba dominando.  **Herramienta Subseleccionador: http://www.aulaclic.es/flash8/graficos/herr_subselec2004.gif**Esta Herramienta complementa a la Herramienta Pluma, ya que permite mover o ajustar los vértices que componen los objetos creados con dicha herramienta.  **Herramienta Bote de Tinta: http://www.aulaclic.es/flash8/graficos/herr_tinta2004.gif**Se emplea para cambiar rápidamente el color de un trazo. Se aplica sobre objetos, si tienen borde, cambia al color mostrado de dicho borde, por el mostrado en el Panel Mezclador de Colores (que coincide con el subpanel Colores que hay en la **Barra de Herramientas.**) |

|  |  |
| --- | --- |
| **Unidad 5.** Trabajar con Textos (I) |  |

|  |  |
| --- | --- |
| Comenzando | [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/a_5_1_1_peli.html',699,500);) |

|  |
| --- |
| Flash abarca todo aquello que nos pueda hacer falta a la hora de crear una animación, y por tanto, también todo aquello relativo a los textos. Sin embargo, Flash fue concebido para crear animaciones gráficas, de modo que tratará cualquier texto como si de un objeto más se tratase, listo para ser animado a nuestro gusto. Esto nos permitirá posteriormente animar textos y crear espectaculares animaciones con muy poco esfuerzo. Flash distigue entre 3 tipos de texto, texto estático o normal, texto dinámico y texto de entrada (para que el usuario introduzca sus datos, por ejemplo), también se puede crear texto que soporte formato HTML etc... |

|  |  |
| --- | --- |
| Propiedades de los Textos |  |

|  |
| --- |
| Para poder escribir, deberemos hacer clic en la Herramienta**Texto** y posteriormente en el punto del escenario en el que queramos empezar a escribir.  Para saber más acerca de **Cómo escribir textos**visita nuestra Página Avanzada [http://www.aulaclic.es/flash8/comunes/ico_mas.gif](http://www.aulaclic.es/flash8/a_5_1_1.htm)  http://www.aulaclic.es/flash8/comunes/orangeball.gif El Panel**Propiedades**contiene las principales propiedades de todos los objetos que empleemos durante nuestra película, de modo que si seleccionamos un texto, podremos ver en él todo lo que nos hace falta conocer sobre nuestro texto. Si tenemos experiencia usando Flash 5, notaremos que todas las propiedades que antes se encontraban en los Paneles**Carácter** y **Párrafo**, se encuentran agrupadas ahora en el Panel**Propiedades.** |
| http://www.aulaclic.es/flash8/graficos/panel_prop_textos2004.jpg  Panel Propiedades |
| Veamos a fondo el Panel**Propiedades**:  **Fuente:**http://www.aulaclic.es/flash8/graficos/char_font2004.gif Desde aquí, al igual que en los editores de texto más comunes podemos seleccionar el tipo de letra o "fuente" que más nos guste.  **Altura: http://www.aulaclic.es/flash8/graficos/prop_car_altmx.JPG** Determina el espaciado entre los caracteres. Útil cuando la tipografía que empleamos muestra las letras muy juntas o para dar efectos concretos al texto.  **Dirección del Texto:** http://www.aulaclic.es/flash8/graficos/char_direc_txt.gif Cambia la orientación del texto de horizontal a vertical, tanto de izquierda a derecha como de derecha a izquierda.  **Ajuste automático entre caracteres: http://www.aulaclic.es/flash8/graficos/prop_car_ajusmx.JPG** Activar esta casilla provoca que la separación entre caracteres se realice de modo automático.  **Posición: http://www.aulaclic.es/flash8/graficos/char_posic2004.gif**Nos permite convertir nuestro texto en subíndices o en superídices (o dejarlo normal).  **URL: http://www.aulaclic.es/flash8/graficos/prop_car_urlmx.JPG**Si queremos que al pulsar sobre nuestro texto, el usuario vaya a una dirección web, nada más fácil que escribirla ahí. Flash la asociará al texto que estés escribiendo en ese momento.  **Destino:** http://www.aulaclic.es/flash8/graficos/char_destino2004.gif Determina si la URL a la que el texto hace referencia se cargará en la misma ventana del navegador, en una nueva ...  **Tipo de Línea:**http://www.aulaclic.es/flash8/graficos/prop_car_tipolineamx.JPG Si el texto que vamos a incluir es dinámico (de lo contrario aparece sin activar), esta opción nos permite determinar cómo queremos que aparezcan las líneas (línea única, multilínea o multilínea sin ajuste).  **Configuración: http://www.aulaclic.es/flash8/graficos/prop_car_opcmx.JPG** Son las clásicas opciones que permiten convertir el texto en Negrita (B), Cursiva (I), cambiar el color del texto y el tamaño.  **Otras Propiedades: http://www.aulaclic.es/flash8/graficos/prop_car_otraspropmx.JPG**Dado que Flash trata los textos como objetos, éstos también tienen anchura, altura y coordenadas. Podemos modificarlos a nuestro gusto, pero debemos tener en cuenta que los caracteres no deben estirarse sin motivo, pues las tipografías son una parte muy importante y deben respetarse. |

|  |
| --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif Un **Párrafo**no es más que un conjunto de caracteres con propiedades comunes a todos ellos. Estos párrafos admiten ciertas opciones que nos permiten trabajar con bloques de texto. El Panel **Propiedades** nos proporciona las siguientes opciones para trabajar con párrafos (entre otras).  **A la Izquierda: http://www.aulaclic.es/flash8/graficos/char_alig1.gif**Todas las líneas empezarán tan a la izquierda como sea posible (dentro del recuadro de texto que hayamos definido).  **Centrar:** http://www.aulaclic.es/flash8/graficos/char_alig2.gif Las líneas se distribuyen a la derecha y a la izquierda del punto medio del Párrafo.  **A la derecha:** http://www.aulaclic.es/flash8/graficos/char_alig3.gif Todas las líneas empezarán tan a la derecha como sea posible (dentro del recuadro de texto que hayamos definido).  **Justificado:**http://www.aulaclic.es/flash8/graficos/char_alig4.gif El texto se ensancha si es necesario con tal de que no quede "dentado" por ninguno de sus límites.  http://www.aulaclic.es/flash8/comunes/orangeball.gif El resto de opciones nos permiten determinar los márgenes (izquierdo y derecho), los sangrados de párrafo y el espacio interlineal. |

|  |  |
| --- | --- |
| trans_770 | trans_160 |
| |  |  | | --- | --- | | **Unidad 6.** Trabajar con Sonidos (I) |  |  |  |  | | --- | --- | | Comenzando |  |  |  | | --- | | ¿Quién sería capaz de ver una película muda? ¿Y una animación espectacular sin sonido?  Hasta hace poco, los únicos sonidos que oíamos en las páginas web eran los famosos "midis", de escaso tamaño y de escasa calidad. De hecho, eran sólo instrumentos musicales sin voz, de ahí su pequeño tamaño (y calidad). Aún así, siempre existía algún creador de páginas web que se aventuraba a poner algún sonido complejo (.wav o .mp3) en su página web, por desgracia, su carga es tan lenta, que la mayoría de los visitantes se irían de la página sin llegar a escucharla.  Hoy, gracias a las conexiones de banda ancha orientadas a contenido multimedia, poner sonido a las páginas web es un poco más fácil, ¿qué aporta FLASH?  Flash nos permite insertar cualquier sonido que queramos en nuestras películas (.wav, .aiff y .mp3) de forma fácil y muy efectiva, ya que es capaz de acelerar la descarga del sonido siempre y cuando se descargue junto con nuestra película.  Podemos dar a la película efectos simples (el típico "clic" al pulsar un botón), efectos complejos (música de fondo) e incluso podemos hacer que la animación se desarrolle conjuntamente con una música (si bien esto último necesitaría que toda la película estuviera descargada previamente, para evitar "atascos" durante el desarrollo de la animación). En definitiva, Flash nos lo vuelve a poner fácil. |  |  |  | | --- | --- | | Importar Sonidos | [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/sonidos.htm',699,500);) |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Si alguna vez habeis intentado añadir un sonido a vuestra animación Flash probablemente os hayais llevado una gran decepción, no conseguirlo. Esto se debe a que no se tiene en cuenta que para poder emplear un objeto en nuestra película, o bien lo creamos nosotros (como llevamos haciendo hasta ahora) o bien lo conseguimos de cualquier otro modo y lo insertamos en nuestra película.  A esto último se le llama **"Importar"**y se puede hacer con sonidos, gráficos, e incluso con otras películas Flash. Importar por tanto, no es más que decirle a Flash que añada un determinado archivo a nuestra película, para que podamos usarlo cuando queramos. En realidad, lo añade a nuestra **Biblioteca**, que es el Panel en el que están todos los objetos que participan en la película (este Panel se verá más adelante).  Así pues si queremos manejar un sonido en nuestra película, deberemos importarlo previamente.  Una vez esté importado, podremos usarlo con total libertad.    Para importar un sonido haz clic en el menú **Archivo** → **Importar** → **Importar a biblioteca**.  http://www.aulaclic.es/flash8/graficos/sonidos_pago.gif  Se abrirá el cuadro de diálogo de **Importar a biblioteca**. Allí deberás seleccionar en **Tipo** de archivo **Todos los formatos de sonido**.  Navega por las carpetas hasta encontrar el archivo de audio que quieras incluir en tu película. Seleccionalo haciendo clic sobre él y haz pulsa el botón **Aceptar**.  El sonido estará listo para usarlo donde quieras, podrás encontrarlo en la **Biblioteca** (menú **Ventana** → **Biblioteca**). |  |  |  | | --- | --- | | Propiedades de los Sonidos |  |  |  | | --- | | En Flash 8, al igual que en Flash 8 todo lo referente a los sonidos lo podemos editar desde el Panel **Propiedades**. Aquí tenemos todo lo necesario para insertar, modificar y editar el sonido que acabamos de importar. Si no hemos importado ningún sonido, nos daremos cuenta de que no podemos seleccionar nada en dicho panel, basta insertarlo para que esto cambie.  Para que aparezca la posibilidad de trabajar con sonidos, deberemos hacer clic en algún fotograma de nuestra película, tras hacer esto, el Panel **Propiedades** toma el siguiente aspecto:  http://www.aulaclic.es/flash8/graficos/panel_prop_sonidos2004.jpg  Veamos las partes que tiene este panel.  **http://www.aulaclic.es/flash8/comunes/orangeball.gif Sonido:** En esta pestaña nos aparecerán las canciones que tenemos importadas, deberemos seleccionar la canción que pretendamos añadir a nuestra película (en el siguiente punto veremos cómo insertarlo). | | http://www.aulaclic.es/flash8/comunes/orangeball.gif **Efecto:** Desde aquí podremos añadir algún efecto a nuestro sonido, como por ejemplo que el sonido pase del canal izquierdo al derecho (esto crea la sensación de que el sonido te envuelve ya que lo oyes por un sitio y luego por otro, aunque queda muy lejos de los verdaderos sonidos envolventes), que el volumen aumente progresivamente etc... Si deseamos añadir complejos efectos sonoros, deberemos tratar adecuadamente el sonido con algún programa creado específicamente para este propósito antes de importarlo.  En el punto **Editar Sonidos**se tratará en más profundidad estos efectos.  **http://www.aulaclic.es/flash8/comunes/orangeball.gif Sinc:** Esta opción nos permite determinar en qué momento comenzará a actuar nuestro sonido, estas son las opciones que tenemos:  **Evento**: Sincroniza nuestro sonido con un evento determinado. Es la opción por defecto y provoca que el sonido se empiece a reproducir al pasar la película por el fotograma en el que está situado. También se puede sincronizar el sonido con botones y los demás tipos de símbolos.   * **Inicio:** Su funcionamiento es equivalente al de "Evento", se diferencian en que si está seleccionado Inicio en vez de Evento y se reproduce otra vez el mismo sonido u otro distinto, lo hace "encima" del sonido actual. Puede ser un efecto muy atractivo, o puede provocar que se forme "ruido" en nuestra película. * **Detener:** Detiene el sonido seleccionado. * **Flujo:** Esta opción sincroniza el sonido con el o los objetos con los que esté asociado, por tanto, si la carga de la película es lenta y las imágenes no fluyen adecuadamente, el sonido se detendrá para sincronizarse con ellas. Este efecto puede dar la sensación de que la película se corta de un modo muy brusco (pensar en que se considera normal que una imagen tarde en cargarse, pero si se detiene un sonido mientras se reproduce, produce una reacción muy negativa en los que pueden estar viendo nuestra película). Por otra parte, es un efecto muy adecuado para algunas situaciones, por ejemplo, el efecto de que un personaje hable durante una película. En esta situación, es muy recomendable que el sonido y las imágenes estén sincronizadas.   http://www.aulaclic.es/flash8/comunes/orangeball.gif**Repetir**: Determina el número de veces que se reproducirá el sonido según lo que escribas en la caja de texto de la derecha. También puedes seleccionar **Reproducir indefinidamente** para que el sonido se reproduzca en un bluce hasta llegar al siguiente fotograma clave.  No recomendamos insertar sonidos con la opción Flujo y a la vez la de Repetir, ya que al estar sincronizados los sonidos con las imágenes, provocaríamos que las imágenes (y los fotogramas que las contengan) se dupliquen también, aumentando considerablemente el tamaño de la película. | | |  | |  |  | | --- | |  | |  |
| trans_770 | trans_160 |
| |  |  | | --- | --- | | **Unidad 7.** Trabajar con Objetos (I) |  |  |  |  | | --- | --- | | Los Objetos. Iniciación |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Independientemente de si estamos trabajando en una animación, en una página web, en un catálogo para un CDRom o en cualquier otra cosa, tendremos que trabajar con objetos. A grandes rasgos, podremos considerar un objeto todo aquello que aparezca en nuestra película y sea visible, de modo que podamos trabajar con él,un objeto sería, por ejemplo, cualquier imagen que creemos o importemos, un botón, un dibujo creado por nosotros mismos etc...  Los objetos así considerados tienen 2 partes fundamentales:  **El Borde:** Consiste en una delgada línea que separa el objeto del exterior del escenario. Puede existir o no, según nos convenga. Cuando creamos un objeto, el borde se crea siempre y su color será el indicado en el **Color de Trazo** (dentro del Panel **Mezclador de Colores**). Si queremos dibujar creando Bordes deberemos emplear las Herramientas **Lápiz**, **Línea** o **Pluma** y si queremos que nuestro dibujo no tenga borde, bastará con seleccionar el borde y suprimirlo (ver siguiente punto).  **El Relleno:**El relleno no es más que el propio objeto sin borde. Es, por tanto, la parte interna del objeto. Su existencia también es arbitraria, ya que podemos crear un objeto cuyo color de relleno sea transparente, como ya se vió en el tema de Dibujar, y por tanto, parecerá que dicho objeto no tiene relleno, aunque en realidad sí que exista pero sea de color transparente. Para dibujar Rellenos (sin borde) podemos usar herramientas tales como el **Pincel** o el**Cubo de Pintura**.  http://www.aulaclic.es/flash8/graficos/relleno_borde.gif |  |  |  | | --- | --- | | Seleccionar | [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/objetos.htm');) |  |  |  |  | | --- | --- | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Para poder trabajar con objetos, es fundamental saber seleccionar la parte del objeto que queramos modificar (mover, girar, cambiar de color...). Podremos observar que las partes de un objeto seleccionadas toman una apariencia con textura para indicar que están seleccionadas. | | | | http://www.aulaclic.es/flash8/graficos/selecc_rell_bord1.gif Objeto SIN seleccionar | http://www.aulaclic.es/flash8/graficos/selecc_rell_bord2.gif Objeto con el BORDE seleccionado | http://www.aulaclic.es/flash8/graficos/selecc_rell_bord3.gif Objeto con el RELLENO seleccionado | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Veamos cómo seleccionar las diferentes partes de un objeto:  **Seleccionar un Relleno o un Borde**: Basta hacer clic 1 vez en el Relleno o en el Borde que queramos seleccionar.  **Seleccionar el Relleno y el Borde de un objeto**: Hacer doble clic en el Relleno.  **Seleccionar todos los bordes (o líneas) de un mismo color que estén en contacto**: Hacer doble clic sobre una de las líneas que tenga el color que pretendamos seleccionar.  **Seleccionar un Símbolo, un texto, o un grupo**: Clic en el Símbolo, en el texto o en el grupo. Estos tipos de objetos mostrarán un borde de color azul (por defecto) al estar seleccionados.  **Seleccionar Varios elementos**: Mantendremos pulsada la tecla SHIFT mientras seleccionamos los objetos que queramos.  **Seleccionar los objetos que se encuentran en una determinada zona**: Para ello usaremos la Herramienta **Selección (Flecha)**. Haremos clic en una parte del escenario y arrastraremos el cursor hasta delimitar el área que contenga los objetos que queremos seleccionar. Si esta área corta algún objeto, sólo quedará seleccionada la parte del objeto que esté dentro del área que hemos delimitado. Si quisiéramos incluir dicho objeto, bastaría con usar la tecla **SHIFT** y seleccionar la parte del objeto que falta por seleccionar.  Este modo de seleccionar objetos permite seleccionar muchos objetos rápidamente, además de permitirnos seleccionar determinadas zonas de los objetos, para cortarlas, pegarlas ...  Además de la Herramienta Flecha, también podemos usar la Herramienta **Lazo**, más útil aún si cabe. Nos permite seleccionar cualquier zona de cualquier forma (la forma la determinamos nosotros) del objeto que sea. Al contrario que la Herramienta Flecha, las áreas seleccionadas no tienen porque ser rectángulares.  **Seleccionar a partir de la Línea de Tiempo:** Si seleccionamos un determinado fotograma en la línea de tiempo, se seleccionan automáticamente todos los objetos que estén en dicho fotograma. Esto es útil para modificar de un modo rápido todos los elementos del fotograma.  **Seleccionar Todo**: La forma más natural de seleccionar todo, consiste en hacer clic en el menú **Edición**→ **Seleccionar Todo.** | | |  |  | | --- | |  | |  |

|  |  |
| --- | --- |
| trans_770 | trans_160 |
| |  |  | | --- | --- | | **Unidad 8.** Capas (I) |  |  |  |  | | --- | --- | | Las Capas. Entendámoslas |  |  |  |  | | --- | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Todo el mundo ha visto alguna vez cómo trabajan los dibujantes de dibujos animados. Y todos hemos visto cómo colocan una hoja semitransparente con dibujos sobre otras y la superposición de todas forman el dibujo final. ¿Por qué no dibujan todo en una misma hoja? ¿Por qué trabajan con varios niveles y con varios dibujos si van a acabar todos juntos?  Los motivos son muchos, y estos niveles que emplean los dibujantes, equivalen a las **Capas**que utiliza Flash. Cada capa es, por tanto, un nivel en el que podemos dibujar, insertar sonidos, textos... con **INDEPENDENCIA** del resto de capas. Hay que tener en cuenta que todas las capas comparten la misma Línea de Tiempos y por tanto, sus distintos fotogramas se reproducirán simultáneamente. | | | Clarifiquemos esto con un ejemplo:    http://www.aulaclic.es/flash8/comunes/orangeball.gif Supongamos que tenemos 2 capas. En una de ellas los fotogramas del 1 al 10 contienen el dibujo de una portería de fútbol. En la otra los fotogramas del 1 al 5 contienen el dibujo de un portero (del 5 en adelante están vacíos).  Pues bien, esta película nos mostrará inicialmente (durante el tiempo que duren los primeros 5 fotogramas) la portería con el portero, para después (durante los fotogramas del 5 al 10) mostrar la portería sin portero.  De este modo la portería es independiente del portero, y podemos tratar estos objetos con total libertad, ya que no interfieren entre ellos para nada. | http://www.aulaclic.es/flash8/graficos/porteromx2004.jpg | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Otra razón para separar los objetos en capas, es que Flash nos obliga a colocar cada animación distinta en una capa. De lo contrario, todos los objetos que se encuentren en dicha capa formarán parte de la animación. Si queremos que un objeto NO forme parte de una animación, deberemos quitarlo de la capa en la que se produce dicha animación.  Siguiendo con el ejemplo del portero, si quisiéramos crear un movimiento que haga que el portero se desplace hacia un lado no hay ningún inconveniente, pero si la portería estuviera en la misma capa que el portero, entonces **AMBOS** objetos se moverían hacia dicho lado, con lo que resultaría imposible que sólo se moviera el portero. La solución es separar los objetos en 2 capas, como ya hemos hecho.  Las capas además, tienen otras utilidades, nos permiten ordenar nuestra película de forma lógica, y nos ayudan en la edición de dibujos (evitando que se "fundan" en uno sólo, o bloqueando el resto de capas de modo que sólo podamos seleccionar la capa que nos interese). | |  |  |  | | --- | --- | | Trabajar con Capas | [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/trabajar_capas.htm',699,500);) |  |  |  | | --- | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif La vista standard de una capa es la que muestra la imagen. Veamos para qué sirven los distintos botones y cómo usarlos.  **Insertar Capas**http://www.aulaclic.es/flash8/graficos/capa_anyadirmx.JPG : Como su nombre indica, sirve para **Insertar** capas en la escena actual. Inserta capas normales (en el siguiente punto se verán los distintos tipos de capas).  **Añadir Capa Guía http://www.aulaclic.es/flash8/graficos/capa_anyguiamx.JPG** : Inserta una capa de tipo guía. Se tratan en profundidad el siguiente punto. | http://www.aulaclic.es/flash8/graficos/capasmx2004.gif | | **Borrar Capa**http://www.aulaclic.es/flash8/graficos/capas_basura.gif : **Borra** la capa seleccionada.  **Cambiar Nombre:**Para **cambiar el nombre**a una capa, basta con hacer doble clic en el nombre actual.  **Propiedades de Capa:**Si hacemos doble clic en el icono http://www.aulaclic.es/flash8/graficos/capas_tipo.gif, podremos acceder a un panel con las **propiedades** de la capa en la que hayamos hecho clic. Podremos modificar todas las opciones que hemos comentado anteriormente y alguna más de menor importancia.  http://www.aulaclic.es/flash8/graficos/trabajar_capas_pago.gif  Aquí puedes cambiar diferentes opciones sobre la capa, como su nombre o su color. También puedes bloquearla u ocultarla.  Pero haremos especial hincapié en la opción **Tipo** cuyas opciones, **Guía** y **Máscara**, veremos **más** adelante en el curso. | | | |  |

|  |  |
| --- | --- |
| **Unidad 9.**  Símbolos (I) |  |

|  |  |
| --- | --- |
| Qué son los símbolos |  |

|  |
| --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif Los **Símbolos** provienen de objetos que hemos creado utilizando las herramientas que nos proporciona Flash 8.      Estos objetos al ser transformados en símbolos, son incluidos en una biblioteca en el momento en que son creados, lo que permite que sean utilizados en varias ocasiones, ya sea en la misma o en otra película. |

|  |  |
| --- | --- |
| Cómo crear un símbolo |  |

|  |
| --- |
| La acción de crear un nuevo símbolo es una de las más usadas en Flash ya que **es uno de los primeros pasos para crear una animación**, como veremos más adelante.      El procedimiento es el siguiente:  **Seleccionamos** el objeto que queramos convertir en símbolo. Abrimos el panel de Propiedades del Símbolo, accediendo al menú **Insertar** → **Nuevo Símbolo** o simplemente pulsando **Ctrl + F8**o**F8**.  http://www.aulaclic.es/flash8/graficos/crear_simb2004.jpg      Una vez hecho esto nos aparecerá una ventana como la mostrada en la imagen. **Introducimos el nombre** del símbolo que vamos a crear.      Esto al principio y mientras tengamos poco símbolos no será muy importante, pero más adelante nos servirá para hacer referencia al objeto.      Sólo nos queda seleccionar el **tipo de símbolo o comportamiento**en que queremos convertir nuestro objeto. Podemos elegir entre Clip de Película, Botón y Gráfico. Sus características y las diferencias entre ellos las veremos en temas posteriores.      Bastará con pulsar **Aceptar** para tener nuestro símbolo creado. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| http://www.aulaclic.es/flash8/comunes/greenball.gif Para practicar estas operaciones te aconsejamos realizar [Ejercicio Crear Símbolo.](http://www.aulaclic.es/flash8/epp_9_1_1.htm)   |  |  | | --- | --- | | Las Bibliotecas |  |  |  |  | | --- | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif En Flash 8 podemos encontrar dos tipos de bibliotecas, las bibliotecas comunes y de ejemplos y aquellas asociadas a las películas que hemos creado. Todas ellas las tenemos a nuestra disposición para utilizar los símbolos que contienen.  Para acceder a las bibliotecas comunes que nos ofrece Flash simplemente tenemos que ir a la Barra de Menús, **Ventana** → **Bibliotecas Comunes** y seleccionar alguna de las que se nos ofrecen. Las hay de todo tipo de símbolos: botones, clips o gráficos.  Para acceder a la líbrería de símbolos de la película que estamos creando de nuevo vamos a la Barra de Menús, **Ventana** → **Biblioteca**. En esta biblioteca aparecerán todos los símbolos que hemos creado hasta el momento.  Podemos comprobar como el nuevo símbolo que hemos creado en el ejercicio anterior (Ejercicio Crear Símbolo) se ha añadido a nuestra biblioteca accediendo a ella como acabamos de indicar. | http://www.aulaclic.es/flash8/graficos/ventana_bibl2004.gif  Los símbolos contenidos en las bibliotecas están identificados por su nombre y por un icono que representa el tipo de símbolo que representan:  http://www.aulaclic.es/flash8/graficos/icono_clip2004.gif **Clip      http://www.aulaclic.es/flash8/graficos/icono_boton2004.gif Botón      http://www.aulaclic.es/flash8/graficos/icono_grafico2004.gif Gráfico** |   Para utilizar un símbolo de una biblioteca basta con **pulsar en el nombre** de dicho símbolo y **arrastrarlo** a cualquier lugar del área de trabajo. Observa cómo podemos hacerlo.  [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/simbolos.htm',699,500);)     |  |  | | --- | --- | | Diferencia entre símbolo e instancia |  |  |  |  | | --- | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Como hemos comentado anteriormente, cuando creamos un símbolo, Flash lo almacena en una biblioteca. Pues bien, cada vez que utilicemos ese objeto en una película, éste se convierte en una **instancia.**  Aunque parece que sean lo mismo, la importancia de esta distinción es que cuando utilicemos un símbolo que hayamos creado previamente en una película, al modificarlo **se modificará la instancia**, mientras que el objeto seguirá intacto, tal y como era en el momento de su creación, de manera que podremos volverlo a utilizar en otro momento.  A la derecha tenéis un ejemplo de una biblioteca, en este caso una biblioteca estándar de Flash. Cada elemento de la biblioteca es un **símbolo.**  http://www.aulaclic.es/flash8/comunes/greenball.gif Para entender mejor este concepto te aconsejamos realizar [Ejercicio Modificar Instancia](http://www.aulaclic.es/flash8/epp_9_1_2.htm). | http://www.aulaclic.es/flash8/graficos/bibl_botones2004.jpg | |

|  |  |
| --- | --- |
| **Unidad 10.**  Gráficos (I) |  |

|  |  |
| --- | --- |
| ¿Qué es un gráfico? |  |

|  |
| --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif Los **Gráficos** son símbolos que nos permiten representar objetos estáticos y animaciones sencillas.      En caso de que utilicemos un símbolo gráfico para realizar una animación, debemos tener en cuenta que ésta estará ligada a la línea de tiempo de la película en la que se encuentre. Es decir, la animación se reproducirá siempre y cuando la película original también se esté reproduciendo. Esto  hace que, pese a tener su propia línea de tiempo, no puedan contener sonidos, controles  ni otros símbolos gráficos.      Así pues, normalmente utilizaremos los gráficos para imágenes estáticas o para cuando nos convenga que una animación se reproduzca sólo cuando determinado frame de la línea de tiempo de la película esté en marcha, ya que para los casos que hemos comentado anteriormente en los que un gráfico no nos es útil, Flash nos ofrece otro tipo de símbolos como veremos en temas posteriores. |

|  |  |
| --- | --- |
| Tipos de Gráficos |  |

|  |  |
| --- | --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif Los gráficos pueden ser:    **a) Estáticos:**estos gráficos se mantienen sin cambios cuando pasa el tiempo. Estos gráficos son los típicos en los fondos y en los objetos que no desempeñan ninguna función especial. Su tamaño y por tanto, el tiempo de carga de este tipo de gráficos, aunque siempre dependerá de la resolución, de sus dimensiones y de la forma en la que estén creados \*, será en general reducido.    **b) Animaciones:**este tipo de gráfico varía su forma, posición u otras propiedades a medida que va pasando el tiempo. Puesto que para realizar la animación se deben usar varios gráficos más además del original o bien realizar determinadas acciones que modifiquen el estado inicial, el tamaño de esta clase de gráficos, para las mismas dimensiones y forma de creación, será mucho mayor que uno estático.           Por esto, aunque las animaciones dan a nuestra web un aspecto más bonito y espectacular tienen dos inconvenientes:  **1)** Si se trata de un Mapa de Bits (ahora veremos que significa esto) la web puede llegar a tener un tamaño excesivamente grande.  **2)** Aunque no se traten de mapas de bits, por ejemplo, si son animaciones típicas de Flash, cuyo tamaño no es excesivo, el hecho de poner muchas animaciones puede llegar a "marear" un poco al visitante de nuestro sitio y desviar su atención de lo que realmente importa, su contenido.    (\*) Los tipos de gráfico anteriores pueden ser, a su vez de dos tipos, según la forma en la que estén creados: Gráfico Vectorial o Mapa de Bits. Para saber más acerca de estos dos tipos de gráfico, échale un vistazo a nuestro tema básico.[[http://www.aulaclic.es/flash8/comunes/ico_menos.gif](http://www.aulaclic.es/flash8/b_10_1_1.htm#b1).](http://www.aulaclic.es/flash8/b_10_1_1.htm" \l "b1) | http://www.aulaclic.es/flash8/graficos/estatico.gif  **ESTO ES UN GRÁFICO ESTÁTICO**    http://www.aulaclic.es/flash8/graficos/animacio.gif  **ESTO ES UNA ANIMACIÓN** |

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | Creando un gráfico y comprobando sus propiedades |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Como explicar teóricamente las propiedades de un gráfico resulta un tanto confuso, vamos a ver de manera práctica lo que queremos explicar. Vamos a crear un gráfico en Flash y a comprobar las propiedades que hemos comentado en el primer apartado del tema. Para ello, abre una nueva película Flash (**Archivo** → **Nuevo**).  Ahora tenemos que crear el objeto que queremos convertir en un símbolo Gráfico. Dibujemos, por ejemplo, un óvalo en cualquier lugar del área de trabajo con la herramienta **Óvalo http://www.aulaclic.es/flash8/graficos/herr_ovalo2004.gif**de la barra de herramientas de dibujo y démosle un color de relleno que será lo que después animemos(Selecciona el fondo del óvalo y dale el color azul mediante la herramienta **Relleno de color**. **http://www.aulaclic.es/flash8/graficos/relleno_color2004.gif** ).  Ya hemos creado nuestro objeto, vamos a convertirlo en un símbolo gráfico ¿lo recuerdas ?  Selecciona el objeto, ves al Menú **Insertar → Convertir en Símbolo**y conviértelo en un símbolo como ya habíamos visto y dándole el **nombre** "GráficoAnimado"y seleccionando el**Comportamiento** **Gráfico**:  http://www.aulaclic.es/flash8/graficos/panel_insertargrafico2004.jpg  Ahora vamos a crear la animación del gráfico. Puesto que todavía no hemos visto a fondo las animaciones, la vamos a realizar de una forma que quizá no sería la más apropiada la mayoría de las veces, pero nos servirá muy bien para este ejemplo. Para elloselecciona nuestro gráfico y pulsa el botón derecho del ratón. Se desplegará un menú, en el que seleccionaremos la opción **Editar** para modificar el gráfico y acceder a su línea de tiempos. Comprueba que estás en la línea de tiempos del gráfico (justo encima del escenario existe una secuencia que nos indica en qué nivel nos encontramos. En la imagen inferior, se puede apreciar que nos encontramos en "Escena1 - Gráfico Animado" y, por tanto estamos DENTRO del gráfico (y la línea de tiempos que vemos es la del gráfico, y no la de la película principal)  http://www.aulaclic.es/flash8/graficos/nivel_grafico2004.jpg  Crearemos a continuación nuevos fotogramas clave **seleccionando uno a uno** los frames número 2, 3 y 4 y **pulsando F6**cuando los seleccionemos.  Pulsa sobre el frame 2 y cámbiale el color de fondo al óvalo como hicimos antes. Haz lo mismo en los dos siguientes frames.    Tu línea de tiempos debería tener este aspecto:  http://www.aulaclic.es/flash8/graficos/linea_graf2004.jpg  Pulsa donde pone Escena 1 justo encima del escenario y de este modo volveremos al nivel inicial (Película principal) y podremos ver nuestro gráfico "desde fuera".    Ahora ya tenemos completo nuestro gráfico animado. **¿Qué crees que sucederá si reproducimos la película?** Comprobémoslo pulsando **Control + Intro**, se visualizará la película.    ¿Y bien? Nada. Nada en absoluto. **El óvalo sigue tal cual**. ¿Por qué? La respuesta está en las características de los gráficos que habíamos explicado: la línea de tiempos del gráfico está ligada a la de la película.  En este caso la línea de tiempos de **la película tenía un solo frame**, mientras que la del **gráfico tenía 4**, por tanto, **no le hemos dado tiempo al gráfico a desarrollar su animación**; sólo ha reproducido un frame, el primero.  ¿Cómo podemos solucionarlo?  Muy sencillo. Basta con recordar lo que hemos comentado, **Cierra** la ventana del reproductor de Flash,**selecciona** el fotograma 5 de la película principal y **pulsa F6**. **Ahora tenemos 5 fotogramas en la película principal.**  Pulsa de nuevo **Control + Intro**.  Ahora sí que se ve la animación. [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/a_10_1_1.html',460,440);)  Guarda este archivo porque lo utilizaremos más adelante. | |

|  |  |
| --- | --- |
| **Unidad 11.**  Clips de Película (I) |  |

|  |  |
| --- | --- |
| ¿Qué es un Clip de Película? |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif Un **Clip de Película** o simplemente Clip, es una película en si misma, como cualquiera de las que podamos haber creado hasta el momento en este curso, pero que está incluida dentro de otra película y, a su vez puede contener también películas insertadas en él.      Al igual que los otros tipos de símbolos de Flash, los clips de película tienen su propia línea de tiempo. Sin embargo, y a diferencia de los Gráficos (como veremos a continuación), esta línea temporal no está ligada a la línea de tiempos del documento que lo contiene, de tal forma que su ejecución es independiente.      Este tipo de símbolos puede contener cualquier otro tipo de símbolo: gráfico, clip o botón, así como cualquier objeto creado con Flash, ya que  un clip es realmente una película.      Otra de las ventajas de los Clips la encontramos cuando realizamos películas de gran complejidad y tamaño, en la que intervienen un número muy elevado de fotogramas, debido a que en la vista general del documento, nosotros sólo veremos un fotograma por clip, el cual puede estar compuesto por muchos frames, lo que nos permitirá tener una mejor visión de cómo se desarrolla nuestra animación, y una barra de tiempos más clara y "limpia".      Los Clips son una de las herramientas que dan mayor potencia a Flash 8, permitiéndonos crear películas de gran complejidad y multiplicar los efectos visuales, ya que se pueden crear múltiples movimientos independientes entre sí y crear conexiones entre los diferentes Clips de un documento.      Todas aquellas cosas que no podíamos hacer con un símbolo de tipo Gráfico, lo podemos hacer con un Clip, además de poder realizar también todo aquello que nos permitía dicho símbolo. Por esto, normalmente se utilizan los clips para cualquier tipo de animación debido a su gran flexibilidad, dejando los gráficos sólo para imágenes estáticas.     |  |  |  |  |  | | --- | --- | --- | --- | --- | | |  |  | | --- | --- | | Comprobar las propiedades de un Clip |  |  |  |  | | --- | --- | | http://www.aulaclic.es/flash8/graficos/ejmvclipsmx2004.jpg | http://www.aulaclic.es/flash8/comunes/orangeball.gifComo hicimos con los gráficos, vamos a ver como la línea de tiempo de un clip no está ligada a la línea de tiempo del documento que lo contiene y, a la vez, veremos la diferencia entre los clips y los símbolos Gráficos, que supone una de las razones de la mayor utilización de los primeros en lo que se refiere a la creación de animaciones.   Para ello abriremos una nueva película Flash (**Archivo** → **Nuevo**). Después importaremos una imagen cualquiera o bien crearemos una. La convertimos en Símbolo (**Insertar** → **Nuevo Símbolo...**) y seleccionamos en Comportamiento "Clip de Película"  Arrastramos al escenario (si no está ya allí) nuestro clip de película.  Ahora tenemos nuestro Clip, insertado en nuestro nuevo documento Flash. Es decir, tenemos dos películas, cada una con su línea de tiempos.   Ahora veamos que son independientes, siguiendo un proceso similar al que utilizamos con los símbolos de tipo Gráfico. | | |

|  |
| --- |
| Observa la línea de tiempos de la película principal y comprueba que sólo tiene un fotograma.  http://www.aulaclic.es/flash8/graficos/e_lineatiempo2004.jpg    Editemos el Clip que insertamos en el documento **(doble clic sobre el avión)** y examinemos su línea de tiempos. Aparecerá un único movieclip. Podemos realizar algo similar a lo realizado con los gráficos, si bien vamos a crear una Animación de Movimiento (que se verá en el Tema 13). Nos quedará así  :  http://www.aulaclic.es/flash8/graficos/e_lineatiempo2004_2.jpg      Como vemos, la duración del clip que hemos insertado es mucho mayor que la película nueva que lo contiene. Si el pez fuera un gráfico ya hemos visto que si intentáramos reproducir la película no ocurriría nada. Veamos qué sucede siendo en este caso un Clip (esto también es válido si la animación se ha creado del modo visto con los gráficos). **(Control + Intro).**  En este caso **sí que vemos la animación** porque en el momento en que el reproductor ha pasado por el fotograma que contenía el clip, éste ha empezado a reproducirse y, puesto que su línea de tiempos es **independiente**, ha seguido reproduciéndose a pesar de que la línea de tiempos principal haya terminado, y lo continuará haciendo hasta que acabe su línea temporal propia. |

|  |  |
| --- | --- |
| **Unidad 12.**  Botones (I) |  |

|  |  |
| --- | --- |
| ¿Qué es un Botón? |  |

|  |
| --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif Los símbolos de tipo **Botón** son los que aportan la mayor parte de la interactividad de las películas Flash con aquel que la está visualizando. Un botón, en Flash, es igual que cualquier botón de cualquier entorno informático, sea web o cualquier otro.  Son elementos que se prestan a que el usuario los presione, desencadenando al hacerlo una serie de acciones. También es habitual ver como este tipo de elementos reaccionan cuando se les pasa el ratón por encima o cuando se les está pulsando, por ejemplo.  Pues bien, para conseguir los efectos interactivos que acabamos de mencionar en otros lenguajes orientados a la web, que es el caso que nos ocupa, debemos crear programas relativamente grandes. Esto es un inconveniente bastante grande ya que el uso de los botones es una práctica muy habitual en el diseño en Internet. Sin embargo, en Flash no ocurre así. Su interfaz está diseñada de manera especial para la creación de botones, lo que nos permite crear todos estos efectos de una manera muy sencilla.  Al igual que los otros símbolos de Flash 8, los botones tienen su propia línea de tiempos. Esta es independiente pero, sin embargo, está formada únicamente por cuatro fotogramas, uno para cada estado posible del botón.  http://www.aulaclic.es/flash8/graficos/convertir_botonmx2004.jpg  **Reposo.**Aspecto por defecto del botón, es decir, cuando el puntero del ratón no está situado sobre él.  **Sobre.**Aspecto del botón cuando situamos el puntero sobre él.  **Presionado.**Apariencia que deseamos tenga nuestro botón mientras lo mantengamos pulsado.  **Zona activa.** Aquí debemos indicar el área real en la que queremos que actúe nuestro botón. Esto es importante sobre todo en botones compuestos sólo por texto como veremos más adelante.  Parece que la limitación de fotogramas podría implicar una limitación en la capacidad de espectacularidad y utilidad de estos símbolos, pero no es así.  Los botones pueden contener a su vez otros símbolos, como clips o gráficos (también de tipo Bitmap). La unión de las posibilidades de todos los símbolos dota a los botones de gran espectacularidad. |

|  |  |
| --- | --- |
| Creación de un Botón |  |

|  |
| --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif En la creación de un botón podemos considerar dos fases. En la primera vamos a convertir nuestro objeto a símbolo de tipo botón y posteriormente veremos cómo completarlo internamente, lo que nos ayudará a entender mejor dicha estructura. |

|  |
| --- |
| Primeramente crearemos el objeto que representará el aspecto por defecto de nuestro botón con las herramientas que nos ofrece Flash 8.  Seleccionaremos el objeto y accederemos al menú **Insertar** → **Convertir en Símbolo**, le daremos el comportamiento **Botón** y asignaremos un nombre a nuestro nuevo símbolo.  http://www.aulaclic.es/flash8/graficos/linea_t_botones2004.jpg  De esta forma ya tenemos transformado el objeto para que se comporte como un botón. Ahora lo completaremos internamente. |

|  |  |
| --- | --- |
| Para determinar cómo debe reaccionar el botón en función de las acciones del ratón, lo editaremos haciendo clic con el botón derecho del ratón sobre nuestro nuevo botón y seleccionando la opción **Editar.**  Cuando tengamos delante la línea de tiempos del botón (observa que tiene el aspecto que hemos mostrado anteriormente), seleccionaremos cada uno de los frames (sobre, reposo, presionado y zona activa) y pulsaremos **F6** para crear un fotograma clave en cada uno de ellos.  http://www.aulaclic.es/flash8/graficos/linea_t_botones2004_2.jpg | |
| Ahora ya podemos modificar el aspecto inicial del botón para cada posición del cursor y marcar el área de acción del botón (fotograma Hit) en la que simplemente podremos dejar la misma figura que la inicial (en este caso sólo es importante la forma del objeto, no los colores u otras cosas) o bien dibujar con las herramientas de dibujos de Flash una nueva figura, en cuya superficie "se sentirá aludido" nuestro botón. |  |
| Aquí podemos ver una muestra de creación de un botón ovalado. [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/crear_boton.htm',660,500);) Como resultado obtenemos el botón que continuación mostramos.    Este es un botón muy básico, como veremos se pueden complicar mucho, pero para empezar nos servirá con este.  Si una vez creado el botón queremos observar sus distintos estados y todavía no hemos terminado la película entera y por tanto no deseamos tener que reproducirla toda podemos hacerlo accediendo a la Biblioteca de nuestra película y seleccionando el botón creado. Para ver lo que comentábamos bastará con pulsar la tecla http://www.aulaclic.es/flash8/graficos/play.gif situada a la derecha de la vista previa del símbolo. | |

|  |  |
| --- | --- |
| Formas en los Botones |  |

|  |
| --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif Los botones son símbolos que pueden tener multitud de formas. Si bien lo más habitual es ver botones rectangulares, cuadrados y circulares, cuya creación es inmediata como vimos en el punto anterior, también hay otros muchos tipos de botones que, pese a ser menos utilizados, es muy habitual verlos en multitud de páginas web.  Entre estos están los creados mediante formas poligonales, aquellos que están formados por texto únicamente, dibujos con diferentes motivos, etc. Es interesante su uso para dar más vistosidad ya que algunos resultan más expresivos, y en esto Flash nos ayuda mucho, debido a la relativa sencillez de creación de botones que sus herramientas de dibujo nos ofrece.  Hay varias formas de botón también muy extendidas, como el botón con relieve sencillo o los botones en forma de píldora. Puesto que existen muchas formas de conseguir estos efectos, a continuación tienes un ejercicio paso a paso que muestra una forma de conseguir el relieve en un botón rectangular. [Ejercicio Botón en Relieve.](http://www.aulaclic.es/flash8/epp_12_1_1.htm) |

|  |  |
| --- | --- |
| trans_770 | trans_160 |
| |  |  | | --- | --- | | **Unidad 13.** Animaciones de Movimiento (I) |  |  |  |  | | --- | --- | | La animación en Flash |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Flash 8 es un programa básicamente orientado a la animación, de ahí la gran importancia de este tema. No obstante, para ir creando animaciones cada vez más complicadas se necesita, sobretodo, mucha práctica.      En este tema y los sucesivos mostraremos las técnicas básicas de animación en Flash. La unión de estas técnicas será la que nos permita crear las más variadas y vistosas animaciones.      En el tema de la animación, Flash ofrece unas facilidades muy grandes, consiguiendo efectos que normalmente requieren muchos conocimientos y espacio de almacenamiento para ser creados, como es el caso por ejemplo de los GIF animados o lenguajes de programación como Java, de una manera muy sencilla, sin necesidad de excesivos conocimientos y ocupando muy poco espacio en disco.      Aunque Flash ofrece técnicas mejores, también se pueden crear animaciones a modo de GIF animado. Se les denomina animaciones Fotograma a Fotograma. Estas son las animaciones más básicas y conviene conocerlas. Para ello conviene visitar nuestro tema básico: [[http://www.aulaclic.es/flash8/comunes/ico_menos.gif](http://www.aulaclic.es/flash8/b_13_1_1.htm#b1)Animaciones Fotograma a Fotograma.](http://www.aulaclic.es/flash8/b_13_1_1.htm#b1)      A continuación veremos distintos tipos de animaciones que el creador de películas Flash deberá tomar como guía inicial. Cabe destacar que podremos aplicar varias de estas animaciones colocándolas en distintas capas. Con esto es con lo que conseguiremos los efectos más espectaculares a la par que útiles.      Es muy importante tener claro un concepto: **no se debe crear animaciones en páginas que no lo necesitan,** ni crear animaciones que distraigan al que visualice nuestros documentos de lo realmente importante, el mensaje. |  |  |  | | --- | --- | | Interpolación de Movimiento (I) |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Es la acción básica de las animaciones en Flash. Permite desplazar un símbolo Flash de un lugar a otro del escenario, siendo necesarios únicamente dos fotogramas, lo que optimiza mucho el rendimiento de la película.       Es importante destacar que **para que una Interpolación de movimiento se ejecute correctamente aquellos objetos que intervengan deberán haber sido previamente convertidos a símbolo.**  También se debe tener cuidado al realizar una Interpolación con dos símbolos que se encuentren en la misma capa, ya que el motor de animación los agrupará como uno sólo y el resultado no será el esperado. Por esto es conveniente asegurarse de dos cosas:  **1)** Separar en distintas capas los objetos fijos y los que estarán animados.  **2)** Poner también en distintas capas objetos que vayan a ser animados con direcciones o formas distintas.       Pasemos pues al tema en cuestión. Una interpolación de movimiento, como hemos dicho, es el desplazamiento de un símbolo de uno a otro punto del escenario. El hecho de que sólo se necesiten dos fotogramas es debido a que Flash, únicamente con la posición inicial y final, "intuye" una trayectoria en línea recta y la representa (veremos que también se pueden realizar movimientos no rectilíneos).       Cuando realicemos la interpolación correctamente observaremos un aspecto como este en la línea de tiempo.  http://www.aulaclic.es/flash8/graficos/interpolac1mx.JPG        Esto indica que la animación cambiará la posición del símbolo del fotograma 1 hasta la posición del mismo símbolo en el fotograma 20, utilizando precisamente 20 fotogramas. El número de fotogramas que se usen en la interpolación de movimiento indicará las subetapas de que constará la animación. Cuantas más subetapas más sensación de "continuidad" (menos saltos bruscos) pero a la vez menos velocidad en el movimiento.        La velocidad en el movimiento de las películas la podemos cambiar también modificando su parámetro en la barra de tiempos, http://www.aulaclic.es/flash8/graficos/interp2.gif pero esto no cambiará lo que hemos comentado anteriormente.        La velocidad está expresada en Fotogramas Por Segundo (fps) y se puede modificar haciendo doble clic en el lugar que hemos indicado de la barra de tiempos. A mayor valor más velocidad, pero se deben poner siempre suficientes fotogramas para que se desarrolle la animación como queremos.    En la animación del tema 1 podrás ver como realizar una interpolación de movimiento.  Mantén en mente que cuando realices una interpolación de movimiento el fotograma inicial y final deberán ser diferentes, en caso contrario no se creará ningún tipo de animación.  Observa en las siguientes imágenes dónde está situada la *pelota* en el primer fotograma de la animación y su posición final en el último fotograma:    http://www.aulaclic.es/flash8/graficos/interpolacion_pago_1.gif          http://www.aulaclic.es/flash8/graficos/interpolacion_pago_2.gif          Aquí puedes ver como realizar una interpolación de movimiento [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/interpolacion.htm',660,500);)        Si el objeto con el que queremos hacer la interpolación de movimiento no está convertido a símbolo nos encontraremos con algo así...  http://www.aulaclic.es/flash8/graficos/interpolac2mx.JPG        ... y la animación no funcionará.        También podemos realizar la interpolación de otra forma, sin convertir previamente el objeto a símbolo, ya que Flash lo convierte a símbolo automáticamente si no lo hacemos, dándole el nombre "Animar" más un número. Basta con hacer clic con el botón derecho sobre el fotograma que contiene nuestro objeto (en la línea de tiempos) y seleccionar **Crear Interpolación de Movimiento.** Después, al crear un fotograma clave en otro lugar de la línea de tiempos se creará automáticamente la interpolación, y sólo deberemos modificar este último fotograma para producir la animación.        Esto quizá no sea lo más conveniente en películas grandes, debido a lo que ya comentamos de la gran cantidad de símbolos que pueden aparecer y la confusión que crean muchos símbolos con nombres parecidos.        Flash también nos permite crear animaciones con trayectorias rectilíneas en varias fases, con direcciones distintas. Para ello, una vez creada la interpolación, basta con pinchar en uno de los fotogramas intermedios y crear un nuevo fotograma clave. Si movemos el símbolo en ese fotograma a otro lugar del escenario y reproducimos, la animación irá primero a esa posición y después a la posición final.        Si hacemos esto varias veces sobre varios fotogramas obtendremos varias trayectorias consecutivas más.        Para entender mejor este concepto recomendamos hacer el [Ejercicio Encadenar Interpolaciones.](http://www.aulaclic.es/flash8/epp_13_1_1.htm) |  |  |  | | --- | --- | | Interpolación de Movimiento (II) |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Acabamos de ver las interpolaciones de movimiento como un mecanismo para desplazar un símbolo Flash de un lado a otro del escenario. Sin embargo, podemos aprovechar este comando del programa para realizar animaciones en las que **nuestro objeto aumente o disminuya de manera progresiva su tamaño.**       Esto es muy sencillo con Flash 8, basta con modificar la instancia del símbolo en el último fotograma de la interpolación de movimiento, pero esta vez cambiándole el tamaño.       Por supuesto, podemos aplicar ambos efectos a la vez, de manera que el cambio de tamaño se producirá mientras el objeto se desplaza. También podemos hacer el cambio de tamaño en varias fases o secuencias encadenadas como en las interpolaciones de movimiento comunes.       El ejemplo siguiente incorpora estas tres características de la Interpolación de Movimiento de Flash 8.         Y la línea de tiempos que nos queda es algo tan sencillo como esto:  http://www.aulaclic.es/flash8/graficos/inter_txtmx2004.jpg | |  |

|  |  |
| --- | --- |
| **Unidad 14.** Animación por Forma (I) |  |

|  |  |
| --- | --- |
| Interpolación por Forma |  |

|  |
| --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif Cuando lo que queramos no sea cambiar la posición de un objeto en el escenario, sino su forma de manera progresiva (o ambas cosas a la vez), Flash 8 nos ofrece la técnica de la **Interpolación por Forma**, que consiste simplemente en ir transformando el contorno de un objeto creado en su interfaz hasta que sea igual que el contorno de otro objeto distinto.      Realizar una interpolación por forma, es muy semejante a crear una interpolación de movimiento. Flash genera fotogramas intermedios en los que va variando ligeramente la forma del fotograma anterior. Así como hacíamos en el tema anterior, sólo necesitamos dos fotogramas clave. Colocaremos en el primer fotograma el objeto con su aspecto original, y en el último la apariencia final que queremos que tenga.      Esta vez, es importante destacar que **para que una Interpolación por Forma funcione como es debido aquellos objetos que intervengan deberán ser objetos vectoriales (no símbolos Flash).**  Debemos tener también dos aspectos en cuenta:  **1)** Separar en distintas capas los objetos fijos y los que estarán animados.  **2)** Poner en distintas capas objetos que vayan a ser transformados con formas distintas, ya que Flash transformará todos los objetos vectoriales del primer fotograma en aquello que haya en el último fotograma de la interpolación.      Si realizamos la interpolación por forma correctamente la línea de tiempos tendrá este aspecto:  http://www.aulaclic.es/flash8/graficos/anim_forma2004_2.jpg      Las consideraciones sobre la velocidad y el número de fotogramas que vimos en las interpolaciones de movimiento se pueden aplicar también a la interpolación por forma y a cualquier animación hecha con Flash.      Si hay algún fallo en los parámetros necesarios para que la interpolación se ejecute bien, como por ejemplo, que uno de los objetos en capa sea un símbolo, se mostrará algo así en la línea de tiempos:  http://www.aulaclic.es/flash8/graficos/anim_forma2004.jpg    Para crear una interpolación de forma deberás actuar de forma similar que para crear una de movimiento. Una vez tengas los fotogramas de inicio y fin de la animación deberás seleccionar todos los fotogramas que participen en ella y abrir el panel **Propiedades**.  http://www.aulaclic.es/flash8/graficos/interpolacion_forma_pago.gif  Allí deberás seleccionar la opción **Forma** en el desplegable **Animar**. Y ya habrás terminado.  Fácil, ¿verdad?      Podemos observar como realizar una interpolación por forma pulsando aquí. [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/interpolacion_forma.htm');) |

|  |  |
| --- | --- |
| A la derecha podemos ver el resultado del proceso anterior. |  |

|  |
| --- |
| También podemos realizar interpolaciones de forma en varias fases de manera análoga a como lo hacíamos en el tema anterior. De esta forma podemos hacer que un determinado objeto de convierta en otro antes de adoptar su forma definitiva. |

|  |  |
| --- | --- |
| Ahora, a la derecha podemos ver el ejemplo anterior pero haciendo que se transforme previamente en un triángulo antes de pasar a ser cuadrado. |  |

|  |
| --- |
| Lo hemos conseguido simplemente haciendo **clic con el botón derecho** sobre el fotograma 10 y creando un fotograma clave (**Insertar Fotograma Clave**). Luego sólo tenemos que borrar el objeto que aparece en el escenario y dibujar el triángulo. |

|  |  |
| --- | --- |
| Transformar Textos |  |

|  |
| --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif  Debido a la importancia de los textos, es importante comentar las aplicaciones que tienen los cambios de forma sobre ellos. Añadimos así una posibilidad más para realizar logotipos o presentaciones vistosas y transmitir información de manera espectacular.        Recordemos que para realizar una interpolación de forma es necesario que el objeto sea de tipo vectorial. Este no es el caso de los textos, que se basan en fuentes y son creados en función de los valores de la tabla ASCII y un tipo de letra asociado. Para solucionar este problema, deberemos transformar el texto previamente en un objeto vectorial, es decir, como si hubiéramos repasado con la herramienta lápiz el contorno de cada letra. |

|  |  |
| --- | --- |
|  | ¿Posibilidades de esta técnica en textos? Muchísimas, aunque puede llevar a confusiones si no se hace adecuadamente. Por ejemplo debemos asegurarnos de dar suficientes fotogramas de margen para que se note la transición. También es conveniente dar un tiempo para que cada letra sea mostrada y pueda ser visualizada y "entendida" por aquel que vea la película, porque si no lo hacemos corremos el riesgo de que la animación parezca sólo un garabato revolviéndose.     Una de las posibilidades es "deletrear" una palabra de manera tan vistosa como vemos a la izquierda. Podemos comprobar como simplemente cambiando el color del objeto final, se produce una transición también en el color. Observemos a su vez como cuando se forma cada letra, mantenemos su imagen el tiempo suficiente como habíamos indicado que era necesario. |

|  |  |  |
| --- | --- | --- |
| En este otro ejemplo se observa la transición de colores más claramente y, además, observamos transiciones de forma a texto.       |  |  | | --- | --- | | Cuando creemos interpolaciones de forma y queramos incluir textos deberemos actuar de un modo especial dado que un objeto de texto no se considera como una forma. Para ello, y una vez tengamos la animación creada con una forma en el fotograma inicial y un texto en el final veremos que la interpolación aparece como incorrecta. Deberemos, entonces seleccionar el fotograma donde se encuentre el texto y seleccionarlo.  Haremos clic en el menú **Modificar**→ **Separar** para convertir el texto en un conjunto de puntos que creen una forma. Y la interpolación ya estará lista.  Consejo: Cuando vayas a aplicar el comando **Separar** sobre una palabra completa observa que primero lo que hace es separar la palabra en letras inidividuales. Deberás ejecutar de nuevo el comando para que estas letras se separen y creen una forma. | http://www.aulaclic.es/flash8/graficos/interpolacion_texto_pago.gif | |

|  |  |
| --- | --- |
| trans_770 | trans_160 |
| |  |  | | --- | --- | | **Unidad 15.** Efectos sobre Animaciones (I) |  |  |  |  | | --- | --- | | Introducción |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif  Si no te parecen suficientes las interpolaciones de forma y de movimiento, y quieres realizar animaciones más complejas, puedes combinar las interpolaciones de movimiento con los efectos y transformaciones que se pueden realizar sobre los símbolos que las componen y aplicar efectos sobre las mismas interpolaciones.         Una vez comprendidas todas las técnicas de este tema, estaremos en disposición de realizar todo tipo de animaciones en Flash. Sólo tendremos que poner un poco de imaginación porque, como hemos visto y veremos, el programa nos facilitará enormemente el trabajo. |  |  |  | | --- | --- | | Efectos sobre la Interpolación de Movimiento |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif  Para describir todos los efectos que podemos aplicar sobre una interpolación de movimiento previamente creada, sin necesidad de tocar ningún símbolo, nos centraremos en el Panel Propiedades, cuando seleccionemos un Fotograma que presente animaciones de Movimiento, los atributos de la animación a modificar aparecerán en este Panel. | | http://www.aulaclic.es/flash8/graficos/prop_fotograma2004mov.jpg |  |  | | --- | | **http://www.aulaclic.es/flash8/comunes/greenball.gif** En el campo **<Etiqueta de fotograma>** podemos darle un nombre a la interpolación que nos ayudará a identificarla en la barra de tiempos. Aparecerá a lo largo de todos los fotogramas que la componen.  **http://www.aulaclic.es/flash8/comunes/greenball.gif Escalar:**Esta casilla, al activarla, permite un incremento/decremento progresivo del tamaño del objeto inicial cuando sus dimensiones son distintas a las del objeto que está en la posición final de la animación. |  |  | | --- | | Todas las interpolaciones de movimiento que hemos hecho hasta ahora tenían esta opción activada aunque no lo hayamos indicado expresamente. Flash la activa por defecto, ya que es lo más normal. De no ser así podríamos encontrarnos con una animación como la siguiente.            Como vemos Flash hace el desplazamiento de posición, pero el cambio de  tamaño se produce bruscamente en el último fotograma.    **http://www.aulaclic.es/flash8/comunes/greenball.gif Aceleración:**Esta barra deslizante nos permite determinar la aceleración del primer tramo de la animación. Podemos elegir valores entre -100 y 100. Si el valor de aceleración es negativo el símbolo se moverá despacio primero e irá acelerando de manera progresiva. Si dicho valor es positivo provocará un efecto opuesto.             Podemos por ejemplo dar la impresión de que un objeto, por ejemplo un coche, arranca, va cogiendo velocidad poco a poco (valor -100) y después va frenando y disminuyendo su velocidad hasta pararse.             Como esta, existen muchas otras aplicaciones. Para comprenderlo mejor recomendamos hacer el [Ejercicio Acelerar Movimiento.](http://www.aulaclic.es/flash8/epp_15_1_1.htm)    **http://www.aulaclic.es/flash8/comunes/greenball.gif Girar:**Esta opción es muy interesante ya que nos da la posibilidad de aplicar una rotación al objeto mientras se produce el movimiento. Para que surta efecto debemos seleccionar el primer fotograma de la interpolación.             Se nos presenta un submenú con cuatro opciones: |  |  |  | | --- | --- | |  | **Ninguna.** Con esto le indicamos a Flash que no aplique rotación alguna sobre el símbolo en movimiento.  **Automática.**Marcando esta opción hacemos que se produzca la rotación en aquella dirección que necesite menos movimientos. Si cuesta lo mismo hacerlo por un lado o y por el otro, es decir, cuando la imagen inicial y final está en la misma posición (en cuanto a rotación se refiere), el hecho de activar esta opción no tendrá ningún efecto. |  |  |  | | --- | --- | | Esta opción es la que está marcada por defecto y por eso no hemos visto hasta ahora un objeto rotar en nuestras interpolaciones. Para que hubiera tenido efecto deberíamos haber rotado la última imagen de la interpolación.         Aquí podemos ver cómo aplicar esta técnica. [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/rotacion_automatica.htm',660,500);)         Este es el resultado que hemos obtenido, como podemos comprobar, ha recorrido el camino más corto, ya que el 6 ha pasado progresivamente de su posición anterior a la que ocupaba el 1 de la otra forma el 6 habría pasado por las posiciones anteriores del 5, 4, 3, 2 y 1, siendo la longitud del  movimiento mucho mayor.    **Antihorario (antes CCMR).**Permite realizar el número de rotaciones completas en la dirección contraria a la de las agujas del reloj, que le indiquemos en la casilla de la derecha. Tanto si la última imagen tiene rotación como si no. | | | En caso de que tuviera rotación la imagen final, esta se le añadiría.  **Horario (antes CMR).**Realiza lo mismo que la opción anterior pero en la dirección de las agujas del reloj (hacia la derecha).  Veamos un ejemplo de rotación a derechas en una interpolación de movimiento. [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/rotacion_derecha.htm',660,500);)  A la derecha vemos el resultado. |  |  |  | | --- | |  |  |  | | --- | | **http://www.aulaclic.es/flash8/comunes/greenball.gif Orientar según trazado:**Si activamos esta casilla, cuando la interpolación sobre la que actuamos es una interpolación por guía de movimiento, Flash hará que el símbolo tome la dirección de la guía, rotando para orientarse en la misma posición que adopta la línea.             Para entender mejor este concepto, recordemos la última animación del tema 13, en la que el avión hacía una especie de "looping". Resultaba un poco extraño ver la animación porque los aviones no van marcha atrás y, en esta ocasión y en muchas otras, no es conveniente que los objetos se muevan solo en el escenario quedando el mismo símbolo inmóvil, como si levitara.             Con este comando solucionaremos esto, y el resultado sería este.    **http://www.aulaclic.es/flash8/comunes/greenball.gif Sincronizar (Sinc):**Con esto evitamos que no se reproduzca el último bucle de un símbolo gráfico incluido en nuestra película con una animación en su línea de tiempos interna, cuando el número de fotogramas que ocupa en la línea principal no es múltiplo de los fotogramas que contiene la instancia.  **http://www.aulaclic.es/flash8/comunes/greenball.gif Ajustar:**Si activamos esta opción, el centro de la instancia (identificado con una cruz) se ajustará forzosamente a la guía que hemos marcado en la correspondiente capa.             De hecho si intentamos mover la instancia en un fotograma de la animación, Flash la volverá a colocar sobre la guía de manera automática. |  |  | | --- | |  | |  |

|  |  |
| --- | --- |
| trans_770 | trans_160 |
| |  |  | | --- | --- | | **Unidad 16.** Generar y Publicar Películas (I) |  |  |  |  | | --- | --- | | Consideraciones sobre el tamaño de las películas |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Antes de aventurarnos a publicar nuestra película para que otros la vean, sobretodo si la vamos a publicar en una página web, donde el tamaño de descarga es de vital importancia, debemos tener en cuenta los siguientes aspectos:    **CONSIDERACIONES EN EL DIBUJO:**  http://www.aulaclic.es/flash8/comunes/orangeball.gif Aunque los **degradados** queden muy vistosos, también requieren más memoria, por lo que debemos evitar su uso excesivo, en la medida de lo posible.  http://www.aulaclic.es/flash8/comunes/orangeball.gif La herramienta**Pincel** gasta más memoria que el resto de herramientas de dibujo, por lo que deberíamos elegir estas últimas en la medida de lo posible.  http://www.aulaclic.es/flash8/comunes/orangeball.gif Hemos visto que la animación de líneas es bastante útil. Sin embargo el uso de **líneas** que no sean las definidas por defecto y que usamos en el capítulo que hemos comentado, hará que el tamaño de la descarga aumente. Por tanto evitemos las líneas discontinuas, de puntos ...  http://www.aulaclic.es/flash8/comunes/orangeball.gif Dibujar las **curvas** con el menor número de nodos posible.    **CONSIDERACIONES EN LA ORGANIZACIÓN:**  http://www.aulaclic.es/flash8/comunes/orangeball.gif **Agrupar** los objetos que estén relacionados, con el comando **Modificar** → **Agrupar**.  http://www.aulaclic.es/flash8/comunes/orangeball.gif Si hemos creado un objeto que va a aparecer varias veces, deberíamos **convertirlo a símbolo**, ya que como hemos visto, Flash lo colocará en la biblioteca y cada vez que quiera mostrarlo, hará referencia a una única posición de memoria.  http://www.aulaclic.es/flash8/comunes/orangeball.gif Ya hemos comentado el mayor tamaño de los **mapas de bits**, lo que hace que debamos minimizar el número de apariciones de éstos en nuestra película.    **CONSIDERACIONES EN LOS TEXTOS:**  http://www.aulaclic.es/flash8/comunes/orangeball.gif Hemos podido observar, cuando manejábamos textos, que cuando abrimos el menú de **tipos de letras**, las tres primeras son siempre "\_sans", "\_serif"  y "\_typewriter". Esto no es una casualidad. Están colocadas ahí para resaltar que estas fuentes ocupan un mínimo de memoria, por lo que se recomienda su uso.    **CONSIDERACIONES EN LA ANIMACIÓN:**  http://www.aulaclic.es/flash8/comunes/orangeball.gif Utilizar lo más que podamos las **interpolaciones de movimiento y las guías** para reducir el número de fotogramas clave y el tamaño de la película.  http://www.aulaclic.es/flash8/comunes/orangeball.gif Evitar el uso de la**interpolación por forma** para animaciones de cambio de color, cuando sea posible.      Independientemente de la optimización que hagamos, a veces no se puede evitar que el tamaño de la película aumente. Es recomendable entonces **hacer un preloader (precarga) cuando la película que queramos publicar sea de tamaño superior a unos 80KB.** |  |  |  | | --- | --- | | Preloader. Cargar la Película entera antes de reproducirla |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Un preloader se usa principalmente para evitar la carga parcial de la película, mientras esta se está reproduciendo, lo que, en ocasiones en las que la película es de un tamaño considerable, hace que la película se vea entrecortada.       Normalmente, los preloaders se hacen vistosos para que el observador no se aburra y deje de lado la opción de visitar nuestra web. Suelen llevar alguna animación sencilla que se va reproduciendo mientras se está cargando simultáneamente la película principal, mucho más grande.       Se pueden complicar mucho más, pero nosotros haremos uno sencillo que nos sirva para entender bien el concepto, y la manera de hacerlo.       Partimos de que ya tenemos nuestra película terminada. Si queremos saber su tamaño podemos ir a **Archivo** → **Configuración de publicación**seleccionar la pestaña **Flash**, y marcar la casilla **Generar Informe de Tamaño**. Si pulsamos el botón **Publicar**, aparecerá en nuestro directorio un archivo de texto donde se explica con detalle el tamaño de nuestra película. |  |  |  | | --- | --- | | http://www.aulaclic.es/flash8/graficos/panel_esc_preloader2004.jpg | Ahora insertaremos una nueva escena (**Insertar** → **Escena**). Deberá ser la primera que se ejecute. Para asegurarnos de ello accedemos a **Ventana → Otros Paneles → Escena**, y en la ventana que aparece arrastramos la escena que acabamos de crear hasta que esté la primera. En nuestro ejemplo le hemos llamado "Preloader" y hemos supuesto que la Escena con la película se llama "Pelicula" (lógicamente). Deberá quedar algo similar a lo que muestra la imagen.             En la escena recién creada insertaremos otra capa, de manera que nos queden dos capas a las que llamaremos, "Acción" y "Cargando".             En la capa "Cargando" crearemos una animación sencilla. Por ejemplo, hagámosle honor  al título y escribamos "Cargando ..."; puedes aplicarle la animación que prefieras, siempre que no sea muy compleja. |  |  | | --- | | En la capa "Acción" diseñaremos el "corazón" del preloader. Vamos a hacer que la animación de nuestra escena de carga se ejecute repetidas veces, hasta que se haya cargado la escena que contiene la película principal, mediante las acciones ActionScript 2 de Flash 8. Para ello abrimos el panel de **Acciones**.               Vamos a seleccionar el último fotograma e introducimos la acción gotoAndPlay, que por defecto nos escribirá "gotoAndPlay()" y nos situará el cursor en el interior del paréntesis para que introduzcamos la escena y el fotograma al que se deberá ir al ejecutar dicha acción. En nuestro caso queremos ir al fotograma 1 de la escena "Pelicula", con lo que escribiremos lo siguiente: *gotoAndPlay("Pelicula", 1);*               Ahora seleccionamos el penúltimo fotograma, que es el que impedirá que se llegue al último (o sea, que repetirá el bucle que muestra la animación "cargando") cuando aún no se haya cargado la película por completo. Para hacer esto, elegimos la acción *\_framesloaded* (fotogramas cargados) y la acción *\_totalframes* (fotogramas totales) e indicamos en el panel acciones lo que queremos hacer. Sería algo como "Cuando el número de fotogramas cargados sea mayor o igual al número de fotogramas totales, entonces comenzamos a reproducir la película" (aunque sea teóricamente imposible que el número de fotogramas cargados sea mayor que el de fotogramas totales, nunca está de más incluir esta condición por si acaso Flash contabiliza internamente los fotogramas de un modo diferente). Podríamos escribir algo como lo que muestra la imagen: | | http://www.aulaclic.es/flash8/graficos/panel_acc_preloader2004.jpg |  |  | | --- | | Con esto, hacemos que aparezca en la pantalla el texto "cargando" y mientras se reproduce su animación, iremos comprobando periódicamente si la película principal está cargada por completo, si no lo está seguiremos mostrando el texto "cargando" hasta que la comprobación que hacemos en el último fotograma nos indique que nuestra película se ha cargado. En ese momento, dejaremos que la animación del preloader llegue a su último fotograma, que contiene la acción que nos llevará a reproducir la película.             Y esto es todo lo que hay que hacer para crear un preloader (aunque por supuesto se puede hacer más sofisticado), con lo que los visitantes ya no huirán de nuestra web. |  |  |  | | --- | --- | | Distribución como archivo swf en un reproductor autónomo |  |  |  |  | | --- | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif  Para poder distribuir películas creadas en Flash que la gente pueda ver, son necesarias dos cosas: crear un archivo SWF y que el que la quiera visualizar tenga instalado el Reproductor de Flash.        Flash nos ofrece varias opciones y funcionalidades para la creación de un archivo SWF. Estas opciones se pueden ver en el panel de **Configuración de Publicación**, al que podemos acceder mediante el menú **Archivo** → **Configuración de Publicación** (Pestaña **Flash**).  Veamos cuáles son estas opciones:  http://www.aulaclic.es/flash8/comunes/orangeball.gif **Versión:**Si queremos publicar nuestra película para que sea vista con versiones anteriores de Flash, debemos seleccionar aquí la versión deseada.  http://www.aulaclic.es/flash8/comunes/orangeball.gif **Orden de Carga:**Aquí indicamos el orden en que queremos que se cargue el documento.    Si seleccionamos **De abajo a arriba** se cargará primero el contenido de la capa inferior (de la capa 1 hasta la última capa que tengamos en el documento).      Por el contrario, si seleccionamos **De arriba a abajo** se cargará primero el contenido de la capa superior (de la última capa que tengamos en el documento hasta la capa 1).      Este comando es importante debido a la propiedad de Flash de ir reproduciendo la película conforme los objetos son cargados. Así, en caso de que nos interese que aparezcan antes unos objetos que otros seleccionaremos uno o el otro.  **http://www.aulaclic.es/flash8/comunes/orangeball.gif Versión de ActionScript:**El uso de ActionScript 2 nos permitirá usar las novedades relativas a objetos, clases etc... Si nuestra película sólo usa acciones sencillas podemos dejar en esta pestaña la opción ActionScript 1.0  http://www.aulaclic.es/flash8/comunes/orangeball.gif **Generar Informe de tamaño:** Esta opción la hemos usado en el apartado anterior. Si la activamos, se creará un archivo de texto con una relación detallada del tamaño del documento.  http://www.aulaclic.es/flash8/comunes/orangeball.gif **Proteger Frente a Importación:** Activando está casilla conseguiremos que cuando otro usuario (o nosotros mismos) queramos importarla no podamos o tengamos que introducir una contraseña si se ha escogido alguna. | **Configuración de Publicación** |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif**Omitir acciones de Trazado:** Las acciones de traza se emplean para comprobar el correcto funcionamiento de la película durante la creación de esta (durante las pruebas). También se consideran trazas los comentarios que insertemos en el código Action Script. Si activamos esta señal, la película creada no los incluirá, ocupará menos tamaño y ahorraremos tiempo innecesario. Es recomendable cuando se publique la película de un modo definitivo.  http://www.aulaclic.es/flash8/comunes/orangeball.gif **Depuración Permitida:**Permite que se pueda depurar el archivo SWF. También exige la introducción de una contraseña ya que se debe tener permiso del creador para Importar el archivo y depurarlo.  http://www.aulaclic.es/flash8/comunes/orangeball.gif**Comprimir película:**Comprime la película al máximo posible.  http://www.aulaclic.es/flash8/comunes/orangeball.gif **Calidad JPEG:** Si en el panel de propiedades del mapa de bits no hemos indicado una compresión concreta, aquí podremos determinar su grado de compresión, que determinará a su vez el espacio ocupado en memoria por este tipo de imágenes. A mayor compresión, menos espacio en memoria ocupará la imagen, pero también su calidad será menor.  **http://www.aulaclic.es/flash8/comunes/orangeball.gif Establecer Flujo de Audio o Evento de Audio:**Esta opción nos permite acceder al Panel "Configuración de Sonido" desde donde podemos configurar, para cada tipo de sonidos, sus características.  http://www.aulaclic.es/flash8/comunes/orangeball.gif **Suplantar configuración de sonido:**Con esto se suplantarán los niveles de compresión seleccionados para cada archivo de sonido de nuestro documento. | |  |

|  |  |
| --- | --- |
| **Unidad 17.** Introducción a ActionScript 2 (I) |  |

|  |  |
| --- | --- |
| ¿Qué es el ActionScript? |  |

|  |
| --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif El **ActionScript**es el lenguaje de programación que ha utilizado Macromedia Flash desde sus comienzos, y que por supuesto, emplea Flash 8. A grandes rasgos, podemos decir que el ActionScript nos permitirá realizar con Flash 8 todo lo que nos propongamos, ya que nos da el control absoluto de todo lo que rodea a una película Flash. Absolutamente de todo.  Sin embargo, en estos dos temas sólo vamos a ver una pequeña introducción a ActionScript que servirá para sentar las bases que permitirán empezar a trabajar con ActionScript. Enseñar a programar con ActionScript requeriría otro curso completo. Profundizar en el conocimiento de este lenguaje queda por cuenta del lector. Recomendamos seguir la estupenda *Ayuda* incluida en Flash 8.  Todo lo referente a este capítulo hace referencia a la versión 2 de ActionScript, última versión de este lenguaje de programación lanzada por Macromedia e incorporada en Flash MX 2004.    **Características generales del ActionScript**  http://www.aulaclic.es/flash8/comunes/orangeball.gif Como ya hemos comentado, el ActionScript es el lenguaje de programación propio de Flash, tal y como el Lingo lo es de Macromedia Director, por ejemplo. El ActionScript está basado en la especificación [ECMA-262](http://www.ecma-international.org/), al igual que otros lenguajes como Javascript.  http://www.aulaclic.es/flash8/comunes/orangeball.gif El ActionScript es, como su nombre indica, un lenguaje de script, esto quiere decir que no hará falta crear un programa completo para conseguir resultados, normalmente la aplicación de fragmentos de código ActionScript a los objetos existentes en nuestras películas nos permiten alcanzar nuestros objetivos.  http://www.aulaclic.es/flash8/comunes/orangeball.gif El ActionScript es un lenguaje de programación orientado a objetos, tiene similitudes, por tanto, con lenguajes tales como los usados en el Microsoft Visual Basic, en el Borland Delphi etc... y aunque, evidentemente, no tiene la potencia de un lenguaje puramente orientado a objetos derivado del C o del Pascal como los anteriores, cada versión se acerca más a un lenguaje de este tipo. Así, la versión 2.0 estrenada en Flash MX 2004 es mucho más potente y mucho más "orientada a objetos" que su anterior versión 1.0  http://www.aulaclic.es/flash8/comunes/orangeball.gif El ActionScript presenta muchísimos parecidos con el Javascript; si conoce Javascript, la sintaxis y el estilo de ActionScript le resultarán muy familiares. Las diferencias entre ambos lenguajes las puede encontrar en la ayuda que acompaña al Flash 8.  http://www.aulaclic.es/flash8/comunes/orangeball.gif En la mayor parte de las ocasiones, no será necesario "programar" realmente, Flash 8 pone a nuestra disposición una impresionante colección de "funciones" (de momento entenderemos "**funciones**" como "*código ActionScript que realiza una función determinada"*) ya implementadas que realizan lo que buscamos, bastará con colocarlas en el lugar adecuado.    Vamos a ver muchas de estas funciones en este curso, pero antes recomendamos tener claros ciertos conceptos relacionados con la programación. Para ello, échadle un vistazo a nuestro tema básico.[[http://www.aulaclic.es/flash8/comunes/ico_menos.gif](http://www.aulaclic.es/flash8/b_17_1_1.htm#b1).](http://www.aulaclic.es/flash8/b_17_1_1.htm#b1) |

|  |  |
| --- | --- |
| El Panel Acciones |  |

|  |
| --- |
| http://www.aulaclic.es/flash8/comunes/orangeball.gif En Flash 8, el Panel **Acciones** sirve para programar scripts con ActionScript. Esto es, que todo lo que introzcamos en dicho Panel se verá reflejado después en nuestra película. Debemos tener claro desde un principio que el Panel **Acciones** puede hacer referencia a Fotogramas u objetos, de modo que el código ActionScript introducido afectará tan sólo a aquello a lo que referencia el Panel. Por ejemplo, en la imagen inferior, se puede distinguir que el Panel**Acciones** hace referencia al **Fotograma 1 de la Capa 1.**  El Panel **Acciones** se divide en 2 partes, a la izquierda tenemos una ayuda facilitada por Flash que nos da acceso de un modo rápido y muy cómodo a todas las acciones, objetos, propiedades etc... que Flash tiene predefinidos. Estos elementos están divididos en carpetas, que contienen a su vez más carpetas clasificando de un modo eficaz todo lo que Flash pone a nuestra disposición. Para insertarlos en nuestro script bastará con un doble clic sobre el elemento elegido.  Posteriormente veremos con detalle los distintos elementos de este Panel.  http://www.aulaclic.es/flash8/graficos/panel_acc_fotograma2004.jpg  A la parte derecha tenemos el espacio para colocar nuestro script, aquí aparecerá lo que vayamos insertando. También incluye herramientas de utilidad, como la búsqueda de palabras, la posibilidad de insertar puntos de corte, la herramienta Revisar Sintaxis y la ayuda de Flash para ActionScript. |

|  |
| --- |
| El Panel **Acciones** de Flash 8, al contrario que el de Flash MX 2004, **no tiene únicamente un modo de edición**, por lo que tiene mucho más en común con el mod en el que se trabajaba con Flash MX que con la anterior versión Flash MX 2004.  La libertad del modo "Experto" es total y por tanto, también lo es la posibilidad de comenter fallos, para asegurarnos de que nuestro script es correcto, al entrar en Modo Experto nos aparecerá un icono con el siguiente aspecto: http://www.aulaclic.es/flash8/graficos/vistomx.jpg Al pulsarlo Flash revisa nuestro código en busca de posibles errores, indicándonos, en su caso, la línea que presente el error y en qué consiste éste.  Es un fallo común pasarse horas buscando porqué nuestra película no funciona correctamente y que el motivo sea que un error de sintaxis ha invalidado todo el código existente en un fotograma, que actua como si no hubiera NADA DE CÓDIGO en él. Pongamos pues, mucha atención en esto y revisemos el código concienzudamente. |

|  |
| --- |
| http://www.aulaclic.es/flash8/graficos/panel_acc_preloader2004.jpg |
| El uso del modo en asistente es mucho más sencillo para ello pulsa el botón **Asistente de Script**y selecciona los comandos que quieras incluir, apareceran las opciones que deberas rellenar para que estos comandos funcionen correctamente y el código se generará automaticamente. Esta es una muy buena utilidad recuperada que permitira a los usuarios que acaban de empezar el uso de este lenguaje.  Aun así es recomendable siempre repasar el código y así irnos familiarizando con él y de este modo poder poco a poco crear nuestro propio código en modo Experto, lo que nos dará myor flexibilidad a la hora de crear nuestras películas. |

|  |  |
| --- | --- |
| trans_770 | trans_160 |
| |  |  | | --- | --- | | **Unidad 18.** Ejemplos ActionScript (I) |  |  |  |  | | --- | --- | | Ejemplos de uso del código ActionScript |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Vamos a mostrar los usos más característicos de código **ActionScript** en los diferentes objetos existentes en Flash 8. De este modo lo comprenderemos mejor y porqué no, nos ahorraremos trabajo al tener ya hechos aquí muchos de los códigos que usaremos en nuestras películas (y con garantía de que funcionan). Mostraremos el código y a continuación, una explicación del mismo. |  |  |  | | --- | --- | | Código ActionScript en BOTONES |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Los Botones (objeto Button para Flash 8) tienen mucha utilidad siempre que queramos que nuestra película interactúe con el usuario. Dado que esto va a ser prácticamente siempre, es conveniente estudiar y entender bien algunos códigos típicos que tendremos que usar para conseguir nuestros propósitos.  Todos los códigos que mostramos en este apartado DEBEN insertarse dentro del objeto de tipo Botón (Button) con el que estemos trabajando. Para ello, seleccionaremos el botón en cuestión y después abriremos y editaremos el Panel **Acciones**, asegurándonos de que éste hace referencia a nuestro botón (lo indicará la cabecera de este Panel).  Veamos algunos de ellos: |  |  |  | | --- | --- | | ***on (release) {          gotoAndPlay(15);       }***  *Esta acción provoca que al pulsar un botón vayamos directamente al Fotograma número 15 de nuestra película.*  La**primera línea** indica a Flash 8 que "al pulsar el botón" haga lo que está entre las llaves " { ... } ". Este código es característico de los botones y nos permite controlar el pulsado del mismo. Dependiendo de lo que coloquemos entre los parentésis, la acción asociada al botón se producirá en un momento o en otro. En este caso, hemos escrito "release" que significa que la acción se producirá cuando, tras pulsar nuestro botón, el usuario deje de hacer clic con el ratón.  La **segunda línea** es la acción que se producirá. En este caso, movemos la cabeza lectora de Flash al fotograma 15. | Ejemplo del código situado a la izquierda. Partimos de una película con 15 fotogramas y una acción stop() colocada en cada uno de ellos. Hemos colocado el código en el botón **"Ir al Fotograma 15".** |  |  |  | | --- | --- | |  | ***on (release) { getURL("http://www.aulaclic.com", "\_blank"); }***  *Esta acción provoca que al pulsar un botón se abra una nueva página web en nuestro navegador por defecto y nos muestre la página www.aulaclic.com*  La **primera línea** tiene la misma función que en el caso anterior  La **segunda línea** es la llamada al navegador web propiamente dicha. Los parámetros indican la dirección URL que queremos mostrar y el modo de ventana que contendrá a la página web. En este caso, se abrirá una nueva ventana independiente de la actual. |  |  |  | | --- | --- | | ***on (release) {           r1.\_width=350;       }***  *Esta acción provoca que al pulsar un botón se modifiquen las propiedades del objeto cuyo nombre de instancia aparece delante de la propiedad.*  La **primera línea** tiene la misma función que en el caso anterior  En la **segunda línea** podemos ver la propiedad **.\_width** (anchura), luego esta llamada va a acceder a esta propiedad. Vemos que hay un operador asignación ( = ), luego deducimos que vamos a asignar una anchura determinada a un objeto. ¿Qué anchura? Pues 350, que es la cantidad que aparece en la parte derecha de la expresión. ¿Y a qué objeto? Al que va delante del "**.**", o lo que es lo mismo, al afectado por la propiedad.  Por tanto, al pulsar el botón vamos a modificar la anchura del objeto r1, que pasará a ser de 350 px. |  |  |  |  | | --- | --- | | Código ActionScript en CLIPS DE PELÍCULA |  |  |  | | --- | | http://www.aulaclic.es/flash8/comunes/orangeball.gif Los Clips de Película (objeto MovieClip) son películas dentro de películas. Pueden tener código AS dentro de sí mismos (al igual que los botones), aunque suele ser más común que algún otro código externo (situado en fotogramas) les haga referencia. Para que el código que contengan los clips de película sea válido, éstos deben tener algun evento de clip asociado (Load, KeyDown etc) que determine cuándo se ejecutará este código (al cargarse el clip, al pulsar alguna tecla ...)  Veamos algunos de estos códigos:    **loadMovieNum("aulaClic.swf", 0);**  Este código AS situado dentro de un fotograma, provoca que nuestra película Flash busque una película llamada "aulaClic.swf" EN LA CARPETA RAÍZ y la cargue en el nivel 0, es decir, en el mismo nivel que la película actual.  La carpeta raíz es la carpeta en la que está la película principal (la que tiene este código).    **loadMovieNum("mifoto.jpeg", 0);**  Este código nos permite cargar en un fotograma una imagen existente en nuestra carpeta raíz con extensión .JPEG de un modo dinámico.  Por dinámico entendemos en tiempo de ejecución, lo que significa que el tamaño de la película no lo notará, ni tampoco la velocidad de descarga de nuestra película. Sólo cuando haga falta ver la imagen y se acceda al fotograma que realice esta llamada, se cargará la imagen y se podrá ver en la película. La cargamos en el nivel 0.  Si tuviéramos la imagen en una carpeta llamada "Imágenes" que se encuentra dentro de nuestro directorio raíz, podríamos acceder a ella del siguiente modo:  **loadMovieNum("Imagenes/mifoto.jpeg", 0);**  Es conveniente remarcar también, que la imagen debe estar en formato .JPEG, no sirve el formato .JPG. |  |  |  | | --- | --- | | La siguiente acción debe incluirse dentro de un botón, pues sucederá al hacer clic sobre él (de este modo el ejemplo es más práctico)  ***on (release) { r1.gotoAndPlay(2); }***  *Esta acción provoca que al pulsar un botón vayamos al fotograma 2 de un Clip de Película determinado. El fotograma de la película principal NO VARIARÁ.*  La**primera línea** es la comentada anteriormente  La **segunda línea** es la acción que se producirá. Flash busca un objeto llamado r1 de tipo clip de película (movieclip) y mueve su cabeza lectora hasta el fotograma 2. La cabeza lectora de la película principal es completamente INDEPENDIENTE de la cabeza lectora de los clips de película y no se verá alterada. | Partimos de una película con 2 fotogramas distintos.  En el primer fotograma están el botón y el clip de película (r1) correspondientes. El clip de película tiene en su línea de tiempos una animación, inicialmente detenida. Veamos que pasa si se accede a su fotograma 2.  En el segundo fotograma no nos importa lo que haya porque la acción situada a la izquierda no nos llevará hasta allí. La cabeza lectora de la película principal es **INDEPENDIENTE.** | |  |

|  |  |
| --- | --- |
| **Unidad 19.** Flash. Navegación (I). |  |

En este tema veremos los puntos más importantes en los que te podrás apoyar para realizar tus animaciones en Flash usando **ActionScript**.

Esta unidad, la de Navegación, está especialmente **orientada a la web**, pues veremos cómo crear elementos que te ayudarán más tarde a crear tus propias presentaciones, secciones, etc. Todo ello apoyado con animaciones y vinculadas entre sí.

A lo largo del tema utilizaremos este ejemplo para ilustrar la teoría de forma guiada.



|  |  |
| --- | --- |
| Los Botones | [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/navegacion.htm',699,500);) |

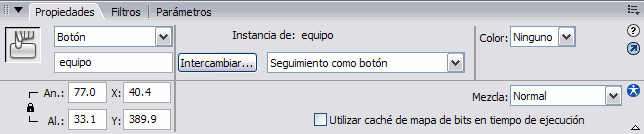
Uno de los elementos que más nos van a ayudar a la hora de añadir interactividad en el diseño son los botones.

En el ejemplo nos hemos ayudado de los botones para crear la navegación entre las diferentes secciones.



Así que el primer paso, después de haber creado la interfaz de la película en una capa, será crear e insertar los botones en una nueva capa para trabajar con mayor facilidad.

**Para asignarle una acción a un botón es necesario darle un nombre de instancia**. Para ello (y como hemos visto en unidades anteriores) escribimos el nombre que queramos (al cual nos referiremos más tarde para llamar al botón) en el **Inspector de Propiedades**, en este caso lo hemos llamado ***equipo***.



Luego, con el botón seleccionado, abrimos el **Panel Acciones** y añadiremos el código que deberá realizar el botón.

Ésta es la parte más importante pues deberemos decidir **a qué estímulo (o evento) responderá el botón**. Existen varios eventos que son capturados en Flash, nombraremos los más importantes:

* **press**: ejecuta la acción al presionarse el botón.
* **release**: ejecuta la acción al soltarse el botón (después de haberlo presionado).
* **rollOver**: ejecuta la acción al desplazar el ratón dentro del botón.
* **rollOut**: ejecuta la acción al desplazar el ratón fuera del botón.

Por ello, escribiremos en el panel acciones el siguiente código para nuestros botones:

on (release) {

}

Nota: **ActionScript diferencia entre mayúsculas y minúsculas**, por lo que si escribes, por ejemplo, Release no será reconocido.

Entre las llaves {} introduciremos el código que queremos que se ejecute al producirse el evento seleccionado sobre el botón.

Existen más manejadores de eventos que puedes revisar en este tema avanzado. [http://www.aulaclic.es/flash8/comunes/ico_mas.gif](http://www.aulaclic.es/flash8/a_19_1_1.htm)

|  |  |
| --- | --- |
| **Unidad 20.** ActionScript - Formularios (I). |  |

|  |  |
| --- | --- |
| Los Elementos de Formulario | [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/formulario.htm',699,500);) |

|  |  |
| --- | --- |
| En el uso de formularios podremos utilizar muchos elementos. Pero los principales serán siempre los mismos: **cajas de texto y botones**.  De vez en cuando utilizaremos otros elementos como los **radioButtons**, **checkBoxes**, **comboBoxes** o **listBoxes**. Veremos como utilizarlos y a sacarles el mayor partido.  Flash ofrece estos objetos como componentes. Para acceder a ellos sólo tendrás que abrir el **Panel Componentes** desde **Ventana** → **Componentes**.  Una vez abierto el panel haz clic en **User Interface** para desplegar sus elementos y verás todos los componentes que podrás utilizar.    Incluso para la introducción de texto en formularios es aconsejable el uso de componentes, pues veremos que poseen propiedades que las simples cajas de texto no tienen.  Para utilizar alguno de estos componentes basta con arrastrarlo del panel al escenario, o puedes arrastrarlo a la biblioteca para utilizarlo más tarde.  En cualquier caso, cuando hayas añadido el componente a tu película deberás darle un nombre de instancia para poder acceder a él en el código y configurar sus opciones en el **Panel Parámetros**:    Panel Parámetros | Panel Componentes |

Veamos cuales son las opciones para los diferentes componentes:

http://www.aulaclic.es/flash8/comunes/orangeball.gif **TextInput** (Introducción de texto):

* **editable**: **true** o **false**. Permite que el texto se pueda editar o no.
* **password**: **true** o **false**. Oculta el contenido del texto mostrándose un asterisco por carácter.
* **text**: Indica el texto inicial de la caja.

http://www.aulaclic.es/flash8/comunes/orangeball.gif **TextArea** (Área de texto):

* **editable**: **true** o **false**. Permite que el texto se pueda editar o no.
* **html**: **true** o **false**. Permite que se acepte contenido HTML dentro de la caja de texto. Propiedad muy útil para incluir enlaces en el texto.
* **text**: Idica el texto inicial de la caja.
* **wordWrap**: **true** o **false**. Permite que se pueda realizar un desplazamiento del texto de arriba abajo. En caso de que no se permita (false) cuando el texto sobre pase el área del componente aparecerá una barra de desplazamiento que permitirá mover el texto de izquierda a derecha.

http://www.aulaclic.es/flash8/comunes/orangeball.gif **Button** (Botón):

* **icon**: Añade un icono al botón. Para insertar un icono deberás crear un gráfico o clip de película y guardarlo en la **Biblioteca**. Una vez allí selecciónalo y haz clic derecho sobre él y selecciona **Vinculación**. Marca la casilla **Exportar para ActionScript** en el cuadro de diálogo que aparecerá y dale un nombre en **Identificador**. Este nombre es el que deberás escribir en el campo **icon** del componente botón. Ni el botón ni el símbolo se ajustará al tamaño del otro, así que deberás modificar sus tamaños para que el icono no sobresalga del botón.
* **label**: Texto que se leerá en el botón.
* **labelPlacement**: **left**, **right**, **top** o **bottom**. Indica la posición de la etiqueta de texto en caso de que se utilice junto a un icono. Respectivamente, izquierda, derecha, arriba y abajo.
* **selected**: **true** o **false**. Indica si el botón se encuentra seleccionado.
* **toggle**: **true** o **false**. Cuando se encuentra a true hace que el botón pueda tomar dos posiciones, presionado y no presionado.

http://www.aulaclic.es/flash8/comunes/orangeball.gif **RadioButton** (Botón de opción):

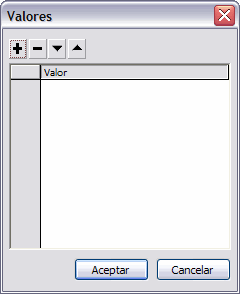
* **data**: Especifica los datos que se asociarán al RadioButton. La propiedad data puede ser cualquier tipo de datos. Puedes acceder a esta propiedad a través de código para ver que contiene.
* **groupName**: Nombre del grupo. En un grupo de botones de opción sólo uno de ello puede estar seleccionado. Definiremos este grupo mediante esta propiedad. Todos los botones que tengan el mismo nombre en **groupName** pertenecerán al mismo grupo.
* **label**: Texto que se leerá al lado del botón.
* **labelPlacement**: **left**, **right**, **top** o **bottom**. Indica la posición de la etiqueta de texto respecto al botón. Respectivamente, izquierda, derecha, arriba y abajo.
* **selected**: **true** o **false**. Indica si el botón se haya seleccionado o no. De nuevo, en un mismo grupo sólo un botón de opción puede estar seleccionado.

http://www.aulaclic.es/flash8/comunes/orangeball.gif **CheckBox** (Casilla de verificación):

* **label**: Texto que se leerá al lado de la casilla.
* **labelPlacement**: **left**, **right**, **top** o **bottom**. Indica la posición de la etiqueta de texto respecto a la casilla. Respectivamente, izquierda, derecha, arriba y abajo.
* **selected**: **true** o **false**. Indica si la casilla de verificación se haya seleccionada.

http://www.aulaclic.es/flash8/comunes/orangeball.gif **ComoBox** (Lista desplegable):

* **data**: Matriz donde determinaremos el valor qué devolverá el componente al seleccionar determinada posición.
* **editable**: **true** o **false**. Permite la edición del campo. Mediante ActionScript puedes hacer que se añadan nuevos elementos a la lista.
* **labels**: Matriz donde determinaremos el nombre de los elementos de la lista. Estos elementos se corresponderán uno a uno a los valores de la matriz introducida en data. Para ambas propiedades se abrirá el siguiente cuadro de diálogo:



Desde aquí podrás añadir o quitar elementos utilizando los botones Añadir elemento y Eliminar elemento. O alterar el orden de éstos subiéndolos o bajándolos en la lista con los botones Lista arriba y Lista abajo.

* **rowCount**: Número máximo de elementos visibles en la lista. Si este número es superado por los elementos se añadirá una barra de desplazamiento.

http://www.aulaclic.es/flash8/comunes/orangeball.gif **List** (Lista):

* **data**: Matriz donde determinaremos el valor qué devolverá el componente al seleccionar determinada posición.
* **labels**: Matriz donde determinaremos el nombre de los elementos de la lista. Estos elementos se corresponderán uno a uno a los valores de la matriz introducida en data. Para ambas propiedades se abrirá el mismo cuadro de diálogo visto para el **ComboBox**.
* **multipleSelection**: **true** o **false**. Permite la selección múltiple de elementos manteniendo la tecla **Ctrl** presionada. También es posible seleccionar un rango de elementos seleccionando uno de ellos y pulsando la tecla **Shift** mientras seleccionamos otro diferente, todos los elementos intermedios resultarán también seleccionados.
* **rowHeight**: Número máximo de filas visibles en la lista. Si este número es superado por los elementos se añadirá una barra de desplazamiento.

http://www.aulaclic.es/flash8/comunes/orangeball.gif **NumericStepper** (Contador numérico):

* **maximum**: Establece el número máximo del contador.
* **minimum**: Establece el número mínimo del contador.
* **stepSize**: Establece el intervalo en el que se aumenta o disminuye el número del campo al pulsar los botones del componente.
* **value**: Valor inicial del campo.

Nota: Todas las propiedades mencionadas para cada uno de los componentes son accesibles a través de ActionScript escribiendo el nombre de instancia del componente seguido de un punto y el nombre de la propiedad:

miBoton.label = "Haz clic aquí";

|  |  |
| --- | --- |
| **Unidad 21.** Los Filtros (I). |  |

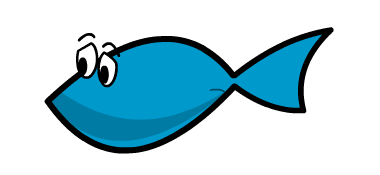
|  |  |
| --- | --- |
| Introducción | [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/filtros.htm',699,500);) |

Como comentamos al principio del curso podemos ver que Flash 8 incorpora una nueva característica: **los Filtros**.

Su uso aumentará la capacidad de mejorar el aspecto y funcionamiento de nuestros proyectos. El único problema es que, en esta versión, **sólo están disponibles para los clips de película, los botones y el texto**.

En este tema veremos los filtros disponibles, su uso y como manejarlos mediante ActionScript.

Para ilustrar su funcionamiento mostraremos ejemplos de cómo afectan los filtros a esta imagen:

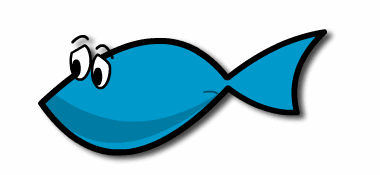
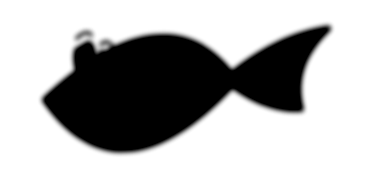


Antes de empezar resaltaremos una cosa. En ActionScript **los colores se indican mediante una cadena de 8 caracteres** con un componente hexadecimal: **0xRRVVAA**. Las dos primeras serán siempre las mismas 0x, las dos siguientes corresponderán a la cantidad de color rojo, las dos siguientes de verde y las dos últimas de azul.

De este modo ***0x000000*** sería el color negro (la ausencia de color), y ***0xFFFFFF*** el color blanco (con el rojo, azul y verde al máximo).

|  |  |
| --- | --- |
| El Filtro Sombra |  |

Utilizando este filtro podremos añadirle una sombra a la imagen:

Sus opciones son las siguientes:

* .**alpha**: determina la transparencia de la sombra (de 0 a 1).
* .**angle**: determina el ángulo de la sombra.
* .**blurX**: determina la cantidad de desenfoque horizontal.
* .**blurY**: determina la cantidad de desenfoque vertical.
* .**color**: determina el color de la sombra (escrito ***0xRRVVAA***).
* .**distance**: determina la distancia de la sombra al objeto.
* .**hideObject**: booleano, esconde el objeto dejando sólo la sombra (segunda imagen).
* .**inner**: booleano, hace que la sombra sea interior.
* .**knockout**: booleano, activa el filtro extractor, hace que el objeto se vuelva del color del fondo.
* .**quality**: número de veces que se aplica el filtro. Puede tomar los valores ***1***, ***2*** ó ***3***. Calidad **baja**, **media** y **alta**, respectivamente.
* .**strength**: determina la intensidad de la sombra.

Podrás utilizar el filtro **Sombra** escribiendo el siguiente código:

import flash.filters.DropShadowFilter;

var sombra:DropShadowFilter = new DropShadowFilter();

sombra.strength = 5;

sombra.alpha = 0.50;

sombra.color = 0x000000;

...

miClip.filters = [sombra];

También podrías crear la sombra directamente con todas sus propiedades en el constructor de este modo:

import flash.filters.DropShadowFilter;

var sombra:DropShadowFilter = new DropShadowFilter(distance, angle, color, alpha, blurX, blurY, strength, quality, inner, knockout, hideObject);

miClip.filters = [sombra];

Así en el constructor deberemos sustituir cada parámetro por el valor que queremos darle.

Puedes omitir los parámetros que quieras, pero ten en cuenta que siempre se interpretarán los valores que des en este orden hasta que no encuentre más.

http://www.aulaclic.es/flash8/comunes/greenball.gifPara practicar la carga y modificación de filtros te aconsejamos realizar el [Ejercicio Filtro Sombra](http://www.aulaclic.es/flash8/epp_21_1_1.htm)

|  |  |  |
| --- | --- | --- |
| trans_770 | | trans_160 |
| |  |  | | --- | --- | | **Unidad 22.** Vídeo (I). |  |  |  |  | | --- | --- | | Introducción | [http://www.aulaclic.es/flash8/comunes/camera.gif](javascript:mostrar('http://www.aulaclic.com.es/flash8/secuencias/videos.htm',699,500);) |   Con Flash cada vez es más fácil añadir vídeos a nuestras presentaciones o webs. En particular Flash 8 incorpora una serie de características que facilitan la tarea al máximo y permiten el uso de vídeos como si se tratase de cualquier otro objeto en pantalla.  A continuación explicaremos cómo hacerlo y cuales son las mejores alternativas. Utilizaremos el siguiente ejemplo para documentar algunos de los pasos que deberemos realizar.       |  |  | | --- | --- | | Importando Vídeos |  |   Lo primero que deberemos hacer es transformar nuestro vídeo de formato **avi**, **mpeg**, **mov**, **wmv** o **dv** al formato que utiliza Flash: **flv**.  Este formato, además de crear archivos de vídeo **de buena calidad muy comprimidos**, te permitirá introducir **puntos de referencia** desde donde controlaremos la interacción del entorno de la película con el vídeo.  Haz clic en **Archivo** → **Importar** → **Importar vídeo** para empezar a configurar el archivo .flv que crearemos.  Se abrirá una pantalla como esta:  Importando vídeo 1  Aquí deberás seleccionar el archivo que quieres importar para utilizarlo en tu película. Haz clic en el botón **Examinar** y búscalo en tus carpetas. Cuando lo hayas encontrado pulsa el botón **Aceptar** y estarás listo para seguir pulsando el botón **Siguiente**.  También es posible marcar la opción **Ya se ha implementado en un servidor Web, Flash Vídeo Streaming Service o Flash Communication Server**. En este caso deberás introducir la **URL** del archivo, que previamente habrá sido preparado para poder utilizarlo en Flash.  La siguiente pantalla te permitirá seleccionar el modo en el que se implementará el vídeo:  Importando vídeo 2  Puedes elegir entre varias opciones, si seleccionas cualquiera de ellas verás su descripción en la derecha de la pantalla. Te adelantamos que para la implementación de vídeo para **Streaming** deberás disponer de un servidor (de pago) que transmita tu vídeo al usuario de una forma óptima.  La opción **Descarga progresiva**, a pesar de no ser la más aconsejada para conexiones de baja velocidad (sobre todo para archivos muy grandes) es la más utilizada por aquellos que no disponen de los recursos necesarios para contratar los servicios de un servidor de vídeo en formato **Stream**.  Así que seleccionaremos la primera opción y pulsaremos **Siguiente**.  Importando vídeo 3  Ahora es cuando realmente configuraremos el archivo de salida. Antes que nada pulsa el botón **Mostrar configuración avanzada**.  Verás 3 pestañas. La primera te servirá para **ajustar la compresión y calidad del vídeo**. Puedes seleccionarla directamente en el desplegable de arriba del todo o cambiar alguna de las opciones que te aparecen en la pestaña **Codificación**.  Nosotros haremos hincapié en la pestaña **Puntos de referencia**.  Importando vídeo 4  Desde aquí podremos configurar puntos en nuestra película. | |  |
| trans_770 | | trans_160 |
| |  |  | | --- | --- | | **Unidad 23.** Juegos (I). |  |  |  |  | | --- | --- | | Introducción |  |   En este tema veremos los elementos básicos que necesitaras a la hora de crear juegos con Flash: el movimiento, los choques, la rotación, creación aleatoria de elementos, etc.  A medida que avancemos en el tema iremos haciendo referencia a [este juego](javascript:mostrarjuego('flash_ejemplos/arma/arma.htm');) que hemos creado como ejemplo. Pero, primero empezaremos por lo básico.   |  |  | | --- | --- | | Manejando Elementos |  |   Cuando trabajemos con objetos necesitaremos acceder a algunas características básicas, como por ejemplo su anchura y altura o su posición en el Escenario.  Podemos modificar el tamaño del objeto utilizando las propiedades \_width, \_height, \_xscale e \_yscale.  Las dos primeras (\_width y \_height) modificarán o devolverán la anchura y altura, respectivamente, en píxeles del objeto referenciado, por tanto, podemos cambiar el tamaño de un clip escribiendo lo siguiente:  miClip.\_width = 100;  miClip.\_height = 200;  Las propiedades \_xscale e \_yscale cambian el tamaño, pero de forma porcentual. Podemos utilizar estas propiedades para hacer un clip el doble de grande que era, por ejemplo:  miClip.\_width = 100;  miClip.\_height = 100;  miClip.\_xscale = 200;  miClip.\_yscale = 200;  Hará que el clip tome el tamaño de 100px de alto y 100px de ancho, aunque su visualización sea del doble (200%).  Podríamos devolver sus dimensiones a su estado real escribiendo:  miClip.\_xscale = 100;  miClip.\_yscale = 100;  También podemos averiguar o modificar la posición de un objeto, basta con trabajar con sus propiedades \_x e \_y:  miClip.\_x = 50;  miClip.\_y = 150;  posicionDeClipHorizontal = miClip.\_x;  posicionDeClipVertical = miClip.\_y;  Recuerda que estas propiedades trabajan con píxeles.  De esta forma podemos mover el objeto de una forma muy fácil, por ejemplo para hacer que se desplace hacia la derecha del Escenario deberás escribir:  miClip.\_x = miClip.\_x + 1;    http://www.aulaclic.es/flash8/comunes/orangeball.gif Utilizando estas propiedades y combinándola con las características de los Movieclips podemos recrear de forma más natural el movimiento.  Veamos este ejemplo:    Aquí hemos creado un clip de película con tres posiciones, parado, izquierda y derecha:  Posiciones del clip  Así, podemos llamar mediante la instrucción gotoAndStop() a uno de los fotogramas del clip para que se muestre dicha posición y dé más realismo al movimiento.  El código que hemos incluido en la película es el siguiente:  var izquierda:Boolean = false;  var derecha:Boolean = false;  miClip.gotoAndStop("parado");  miClip.\_x = 10;  miClip.onPress = function() {  if (miClip.\_x <= 10) {  miClip.gotoAndStop("derecha");  derecha = true;  }  if (miClip.\_x >= 495) {  miClip.gotoAndStop("izquierda");  izquierda = true;  }  }  \_root.onEnterFrame = function() {  if (derecha) {  if (miClip.\_x < 495) {  miClip.\_x = miClip.\_x + 6;  }  else {  miClip.gotoAndStop("parado");  derecha = false;  }  }  if (izquierda) {  if (miClip.\_x > 10) {  miClip.\_x = miClip.\_x - 6;  }  else {  miClip.gotoAndStop("parado");  izquierda = false;  }  }  }  Observa cómo hemos utilizado aquí la función onEnterFrame. Nos valemos de ella para crear el movimiento, de esta forma hacemos que la coordenada x del objeto se incremente cada vez que entre en el fotograma de nuevo. Veremos esta función con más detalle en el tema 24.  Hemos utilizado las variables derecha e izquierda (de tipo booleano) para evaluar si el objeto debe desplazarse. En el evento onEnterFrame chequeamos estas variables y si están a true comenzamos el movimiento hasta la posición que queramos.  A medida que vayamos avanzando en el tema veremos más funciones y métodos que nos permitirán acciones más versátiles y ajustadas a nuestras necesidades.  http://www.aulaclic.es/flash8/comunes/orangeball.gif Otra forma de modificar la apariencia del objeto con el que trabajamos es utilizando la propiedad \_rotation.  Esta propiedad actúa sobre el objeto rotándolo tantos grados como especifiques respecto a su posición original. Por ejemplo:  miClip\_rotation = 90;  Esta línea hará que el objeto rote 90 grados en el sentido de las agujas del reloj. Para rotar un objeto hacia el lado contrario basta con añadirle un signo menos (-) antes del ángulo:  miClip.\_rotation = -90;  Para devolver al objeto a su posición original sólo tendrás que escribir la siguiente línea:  miClip.\_rotation = 0;  Así la rotación que actúa sobre el objeto es nula y se muestra en su posición original.  Recuerda que para que las rotaciones parezcan reales deberás posicionar el objeto dentro del clip en el centro absoluto, para que así la referencia del clip se encuentre en la posición adecuada.  Observa la diferencia entre estos dos clips:  Diferencias de centro de referencia  En ambos la rotación se aplicará respecto al punto de referencia, por lo que en la edición de éste deberás utilizar los botones Alinear centro vertical y Alinear centro horizontal del Panel Alinear (Ventana → Alinear) para centrar el objeto respecto al punto de referencia del clip.  http://www.aulaclic.es/flash8/comunes/orangeball.gif Realiza el [paso a paso de rotación](http://www.aulaclic.es/flash8/epp_23_1_1.htm) para ver mejor como funciona. | |  |
| **Unidad 24.** Animaciones avanzadas (I) |  | |

|  |  |
| --- | --- |
| Movimiento de objetos mediante ActionScript |  |

Para poder mover objetos con ActionScript debemos conocer el **sistema de coordenadas** utilizado por Flash.

La **esquina superior izquierda** es la **posición x=0, y=0**. (ver la imagen)

Para mover un objeto debemos modificar su posición definida por las propiedades ***objeto***.**\_x** y objeto.**\_y**

Por ejemplo, si hacemos ***objeto***.**\_x** ***+=*** ***10*** estamos desplazando el objeto **10 píxeles hacia la derecha**.

Vamos a ver como mover un objeto a partir de este ejemplo de una bola botando.



|  |
| --- |
| El algoritmo es muy sencillo, se trata de ir **modificando la posición de la bola desde una posición inicial para que vaya bajando hasta que llegue al suelo**, en ese momento**invertimos el movimiento** para que suba y calculamos hasta donde tiene que subir. Así sucesivamente hasta que deje de botar.  Para que la bola baje debemos sumar una cantidad a la coordenada **y**, y para que suba debemos restarle una cantidad. Si queremos que la bola se vaya desplazando hacia la derecha debemos sumar una cantidad a la coordenada **x**.  Cuando la bola **baja**, según el esquema de la derecha, pasa de la posición (x1, y1) a la posición (x2, y2), para ello se incrementa la**x** con el valor ***incre\_x*** (que estaria representado por la línea azul) y se incrementa la**y** con el valor***incre\_y*** (que estaria representado por la línea roja).  Cuando la bola **sube**, la única diferencia es que la coordenada **y** va disminuyendo en lugar de aumentando, según el esquema de la derecha, al subir se pasa de la posición (x'1, y'1) a la posición (x'2, y'2), para ello se incrementa la **x** con el valor ***incre\_x*** (que estaria representado por la línea azul) y se **decrementa** la **y** con el valor ***incre\_y*** (que estaria representado por la línea roja) |

|  |
| --- |
| esquema bola |

|  |
| --- |
| Vamos a ver lo más importante del código ActionScript que tenemos un poco más abajo.  Hemos definido creado una bola azul y la hemos convertido en un movieclip, la instancia se llama ***bola\_mc***.  Inicialmente la bola esta en la posición (**bola\_mc.\_x**= 0,**bola\_mc.\_y**= 0)  Mediante la variable ***baja***, sabremos si la bola esta bajando o subiendo. Cuando la coordenada **y** llega al suelo, la bola pasará a subir, cuando la coordenada **y** llegue al tope la bola comenzará a bajar.  En cada bote la bola ira perdiendo fuerza y el bote sera más bajo, la variable ***elasticidad*** determinará esta perdida de potencia.  Cada vez que la bola llega al tope calculamos el nuevo tope, para ello calculamos la distancia de la bola al suelo, la dividimos por la ***elasticidad*** y esa cantidad se la sumamos al tope.  En el esquema, **la linea naranja representa esa cantidad** que le sumamos al ***tope 1*** para pasar al ***tope 2***.  Ya tenemos nuestro código casi preparado, ahora sólo falta hacer que el proceso se repita un número determinado de veces. Esta película Flash sólo tiene un fotograma. ¿Cómo hacemos esto? Lo primero que se nos ocurre es incluir una instrucción ***while ...*** o una instrucción ***for ...*** , si lo hacemos veremos que no funcionará porque la repetición es tan rápida que no da tiempo a ver nada.  La solución está en introducir una instrucción que haga que el proceso **se repita cada 35 milisegundos**. Esto lo vamos a ver en el punto siguiente.  Hemos colocado un botón en la película de forma que cada vez que se pulsa, se detine la ejecución (si todavía no había acabado) y se lanza una nueva ejecución.  Cuando el tope llega al suelo se detine la ejecución. |

|  |
| --- |
| on (release) {  var baja:Boolean = true;  var incre\_x:Number = 3.5;  var incre\_y:Number = 8;  var suelo:Number = 280;  var tope:Number = 0;  var elasticidad:Number = 3;  bola\_mc.\_y = 0;  bola\_mc.\_x = 0;  tope += (suelo - **bola\_mc.\_y**) / elasticidad;  var repeticion:Number;  function botar() {  if (baja) { ***// baja***  **bola\_mc.\_y** += incre\_y;  **bola\_mc.\_x** += incre\_x;  if (**bola\_mc.\_y**>=suelo) {  baja = false;  }  } else { ***// sube***  **bola\_mc.\_y** -= incre\_y;  **bola\_mc.\_x** += incre\_x;  if (**bola\_mc.\_y**<=tope) {  baja = true;  if (tope>=suelo) {  incre\_y = 0;  clearInterval(repeticion);  }  tope += (suelo - **bola\_mc.\_y**) / elasticidad;  }  }  }  clearInterval(repeticion);  repeticion = setInterval(botar, 35);  } |

|  |
| --- |
| En este sencillo ejemplo la trayectoria de la bola es rectilínea, para que fuese un movimiento más real debería seguir una trayectoría curva. Dibujar curvas es un poco más complicado que dibujar rectas. Flash nos ofrece una opción más sencilla para hacer que un objeto siga una trayectoria curva: **las guías de movimiento** que veremos en este mismo tema aplicadas al bote de una pelota. |