****

**ADOBE FLASH • Nivel 2**

**Contenido**

Herramienta Huesos (Bones) ……………………………………………………………………………………………………... 3

Herramienta 3d (Rotación y Traslación) ………………………………………………………………………………………. 6

Menú Desplegable por “Release” ……………………………………………………………………………………………..... 8

Menú Desplegable por “RollOver” …………………………………………………………………………………………….. 11

Precargador con ActionScript ……………………………………………………………………………………………………. 14

Precargador en Línea de Tiempo ………………………………………………………………………………………………. 16

Galería de fotos …………………………………………………………………………………………………………………………. 18

Reproductor MP3 …………………………………………………………………………………………………………………….. 23

Formulario de Contacto ……………………………………………………………………………………………………………. 27

**Herramienta Huesos (Bones)**

La herramienta de huesos  nos permite crear animaciones a diferentes movieclips manteniendo una estructura de esqueleto que los une entre ellos.

Para crear animaciones con huesos inicialmente debemos crear la “armadura” o esqueleto. La creación es un paso fundamental ya que los puntos de articulación que creemos determinan los diferentes movimientos que podremos ejecutar sobre los movieclips.

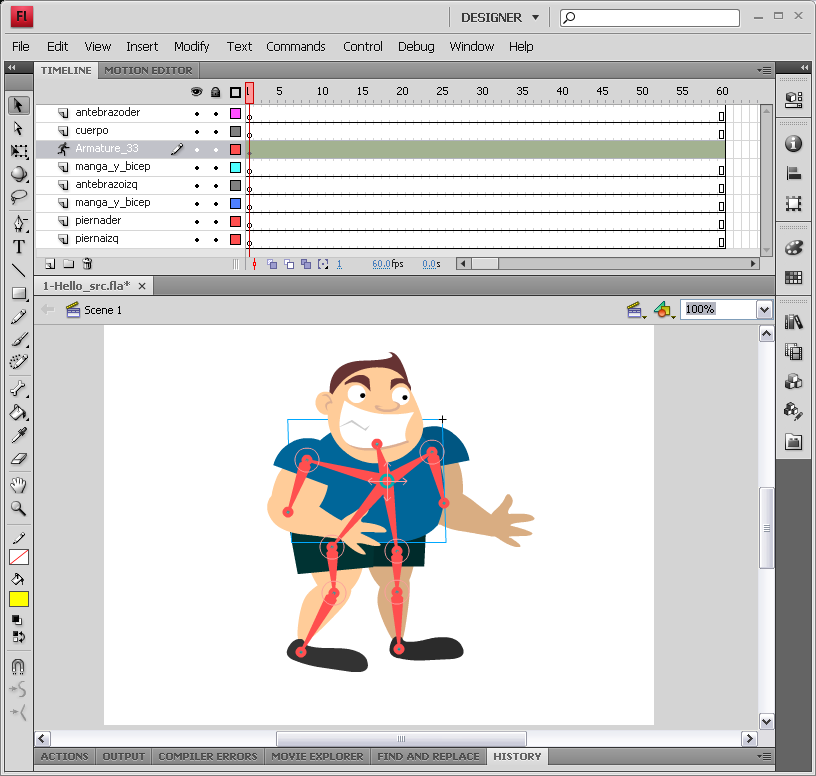
Una vez creada la armadura, el segundo paso fundamental es la configuración de las propiedades de cada movimiento que se permitirá efectuar sobre los movieclips, para los cuales básicamente se permiten dos tipos: rotación y traslación(ésta última en el eje X y en el eje Y).

Para construir la armadura simplemente seleccionamos la herramienta huesos y hacemos click sobre el punto de un movieclip y arrastramos el mouse hasta un segundo punto de otro movieclip creando un brazo que une a ambos. Es importante la localización de este primer click pues determina el eje principal de la armadura.

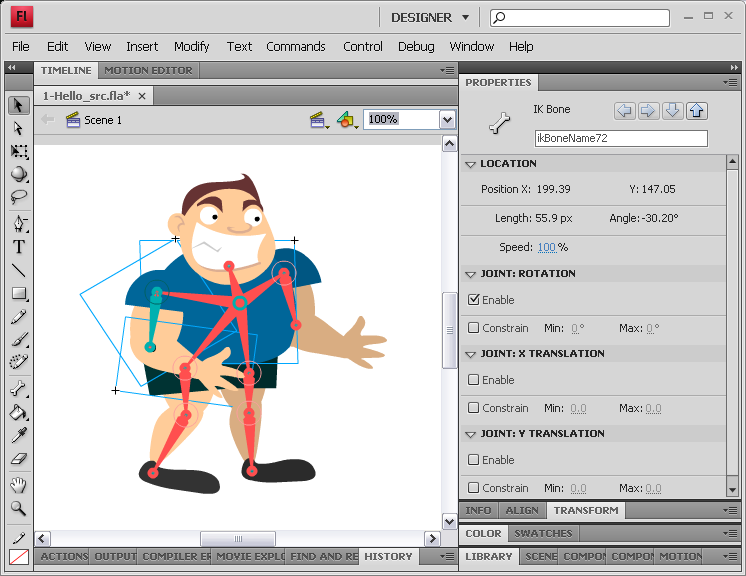
Para este ejercicio utilizaremos los movieclips que muestra la figura, que corresponden a una figura humana donde los miembros del cuerpo los tenemos distribuidos en capas para poderlos animar de manera independiente.



Iremos armando el esqueleto uniendo los movieclips como los muestra la siguiente figura. Al momento de hacer las uniones es posible que se desordenen, para corregir esto usamos la herramienta de “arreglo” presionando el botón derecho sobre los movieclips y colocándolos hacia atrás o hacia adelante dependiendo donde queramos ubicarlos.

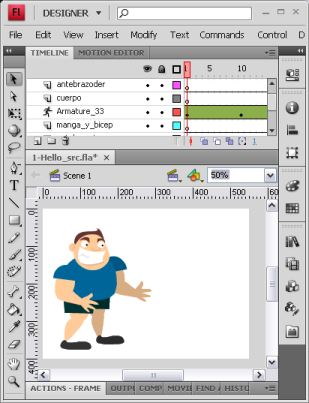
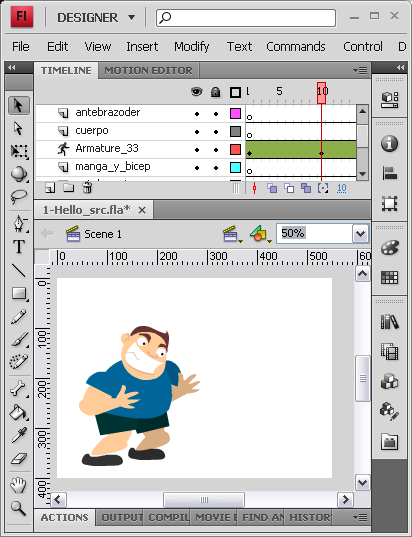
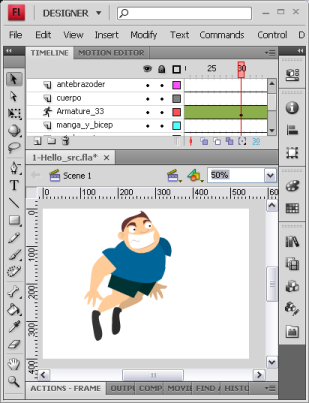
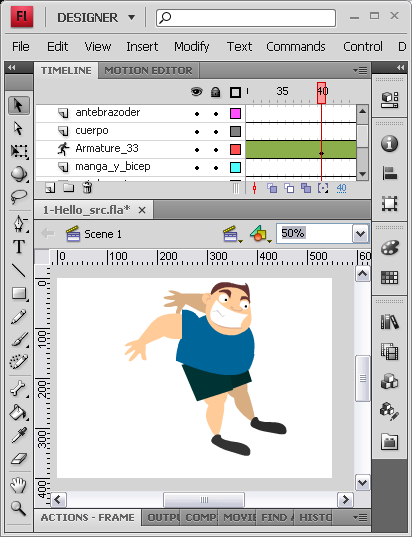


Una vez creada la armadura ya podemos animar el objeto en diferentes frames de la línea de tiempo para crear animaciones articuladas. Para esto es importante definir el tipo de animación seleccionando cada brazo de la articulación y configurando los movimientos de rotación y traslación que queramos en cada instante de tiempo.

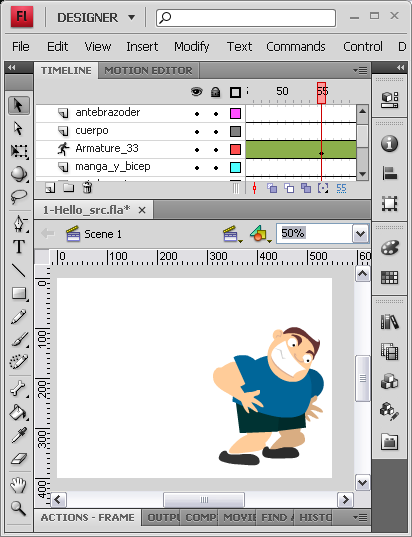
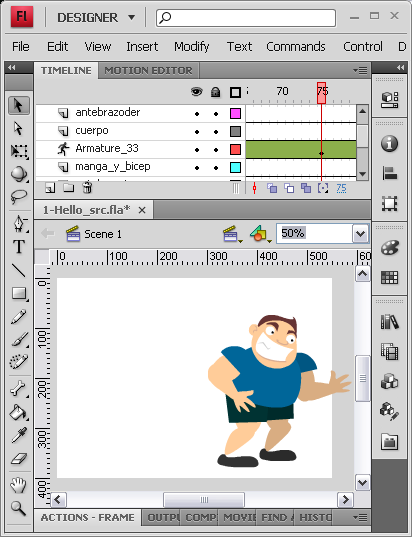
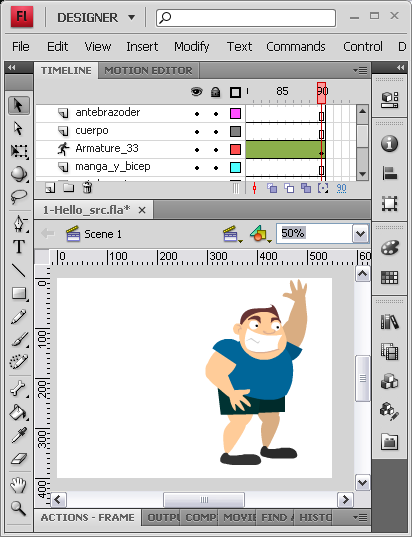


Si queremos hacer ajustes podemos eliminar brazos de la armadura, sólo debemos seleccionar el brazo y presionar “delete”.

Un ejemplo de movimiento se muestra en la siguiente secuencia de animaciones creadas en los frames: 1-10-30-40-55-75-90

Frame 1 Frame 10 Frame 30 Frame 40

Frame 55 Frame 75 Frame 90

**Animación**:



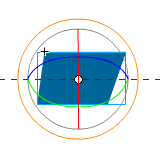
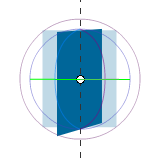
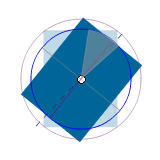
**Herramienta 3D (Rotación y Traslación)**

La herramienta 3D  incorporada en las últimas versiones de Flash nos permite hacer simulaciones de tercera dimensión sobre objetos planos.

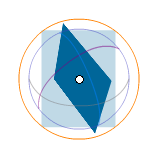
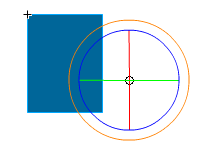
Para trabajar simulando animaciones en 3D debemos intentar imaginar un espacio tridimensional con tres ejes X, Y y Z.

**Rotación**

Cuando seleccionamos con la herramienta de rotación 3D un movieclip en el escenario, visualizamos sobre él cinco parámetros de control sobre su rotación en el espacio:

eje X (línea roja vertical) eje Y (línea verde horizontal) eje Z (círculo azul)

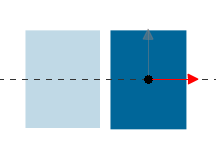
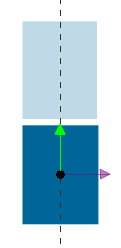
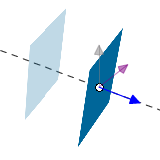
 

todos los ejes (línea naranja) orígen de coordenadas de los ejes (círculo blanco).

|  |  |
| --- | --- |
|  |  |

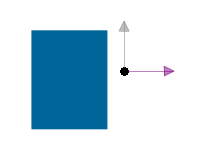
**Traslación**

Cuando seleccionamos con la herramienta de traslación 3D un movieclip en el escenario, visualizamos sobre él cuatro parámetros de control sobre su traslación en el espacio:

eje X (flecha roja horizontal) eje Y (flecha verde vertical) eje Z (flecha azul visible

cuando se ha rotado el objeto)

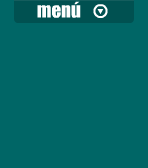
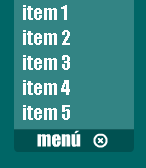


Origen de los ejes de coordenadas (centro negro cuando no muestra la Z).

|  |  |
| --- | --- |
| **Parámetros de rotación y transformación.**  También podemos configurar la rotación y traslación de objetos mediante valores específicos en los paneles de *propiedades* y *transformación*, cuando queremos trabajar con absoluta precisión.  En el panel de propiedades podemos ajustar:  • la posición exacta en cada eje X, Y y Z  • las dimensiones del objeto (W y H)  • El ángulo de perspectiva (por defecto 55)  • Los puntos de fuga para los ejes X y Y.  En el panel de transformación podemos ajustar:  • Los ángulos de rotación para cada eje  • El origen de coordenadas. |  |

**Menú Desplegable por “Release”**

El objetivo del ejercicio es la creación de un menú desplegable reutilizable que se active al hacer click sobre el mismo y se repliegue al escoger una de sus opciones.

Necesitaremos construir la siguiente estructura de movieclips con sus correspondientes nombres de instancia:

menu\_desplegable (mc: menud)

boton\_menu (botón: menub)

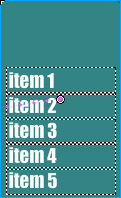
opciones (mc: options)

boton5 (b5)

boton2 (botón: b2)

boton1 (botón: b1)

**Movieclips y botones:**

menud menub options b1

|  |  |
| --- | --- |
| Una vez creada la estructura se ubicando únicamente visible dentro del escenario al botón “menúb”. |  |

Teniendo los movieclips y los botones ubicados en el escenario con los nombres de instancia arriba indicados, editamos el movieclip menud haciendo doble click sobre él, y en su línea de tiempo crearemos dos interpolaciones de movimiento sobre el movieclip de opciones y sobre el boton menub:



En los frames 1 y 24 colocamos los objetos opciones y menub de manera que solo sea visible dentro del escenario el botón menub.

En el frame 12 hacemos descender juntos ambos objetos hasta que todos los botones se encuentren dentro del escenario.

Una vez que se han creado ambas interpolaciones procedemos a colocar el código de actionscript que permitirá el despliegue del menú, el mismo se colocará en los frames 1 y 12 como se indica a continuación:

**Actionscript para el frame 1:**

stop();

menub.onRelease = function() {

play();

}

options.b1.onRelease = function() {

play();

// otros comandos para la opción 1

}

options.b2.onRelease = function() {

play();

// otros comandos para la opción 2

}

options.b3.onRelease = function() {

play();

// otros comandos para la opción 3

}

options.b4.onRelease = function() {

play();

// otros comandos para la opción 4

}

options.b5.onRelease = function() {

play();

// otros comandos para la opción 5

}

**Actionscript para el frame 12:**

stop();

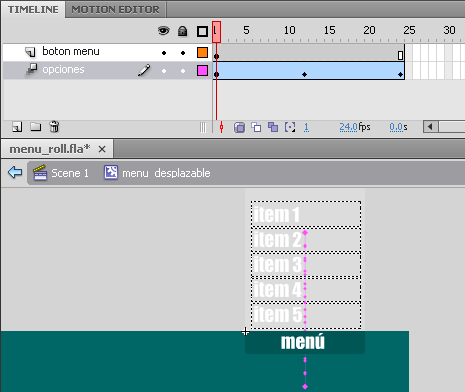
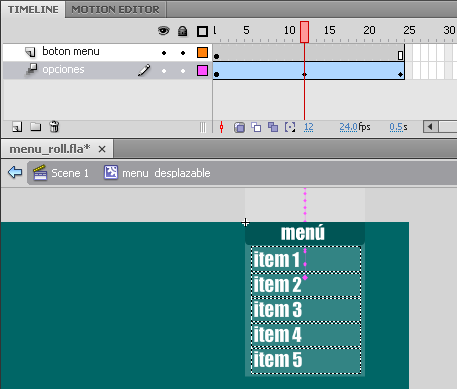
Finalmente se publica la película.

Para reutilizar este menú basta con copiar el objeto principal ubicado en la escena 1 dentro de cualquier otro documento de flash.

**Menú Desplegable por “RollOver”**

En este ejercicio utilizaremos los mismos componentes del ejercicio anterior, sólo haremos algunas modificaciones a las interpolaciones y al código actionscript.

Teniendo la misma estructura de movieclips anteriormente creada, editamos el movieclip *menud* y sólo creamos una interpolación de movimiento para el movieclip *options* como indican las siguientes figuras:

Posición de options en los frames 1 y 24 Posición de options en el frame 12

Como se observa en las figuras, el botón *menub* permanece siempre en la misma posición.

Adicionalmente deberemos crear una zona externa al área de botones dentro del movieclip *opciones*, para capturar el evento de salida del menú.

|  |  |
| --- | --- |
| Una vez editado el movieclip *opciones*, cremos una capa nueva que llamamos “zona fuera” y dibujamos una area alrededor de los botones como muestra la figura a la derecha y la convertimos en movieclip con el nombre “zona\_fuera”. Posteriormente le damos el nombre de instancia “zout” y le damos un alfa=0, para hacerla invisible. |  |

Una vez creada la interpolación colocamos los siguientes códigos actionscript en los frames 1, 2, 12 y 13:

**Actionscript para el frame 1:**

stop();

menub.onRollOver = function() {

play();

}

options.zout.onRollOver = function() {

play();

}

options.b1.onRelease = function() {

play();

// otros comandos para la opción 1

}

options.b2.onRelease = function() {

play();

// otros comandos para la opción 2

}

options.b3.onRelease = function() {

play();

// otros comandos para la opción 3

}

options.b4.onRelease = function() {

play();

// otros comandos para la opción 4

}

options.b5.onRelease = function() {

play();

// otros comandos para la opción 5

}

**Actionscript para el frame 2:**

menub.enabled = false;

options.zout.enabled = true;

**Actionscript para el frame 12:**

stop();

**Actionscript para el frame 13:**

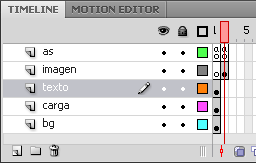
menub.enabled = true;

options.zout.enabled = false;

Finalmente publicamos la película.

**Precargador con Actionscript**

Para construir este precargador necesitaremos 5 capas como muestra la siguiente figura



En la capa inferior *bg* colocaremos una barra de fondo como muestra la siguiente figura, ésta nos dará el tamaño total de la carga.



En la capa *carga* colocamos una copia de la barra de fondo, la convertimos en movieclip y le damos nombre de instancia “barra\_mc”. A su vez, le damos un efecto de color y le aplicamos un filtro para diferenciarla de la barra de fondo como muestra la siguiente figura.



En la capa texto colocaremos un campo de texto dinámico, le damos nombre de instancia “porcentaje\_txt”, y tipeamos el texto 100% para darle las características de texto que sean de nuestro agrado.

En la capa imagen, en el frame 2, colocamos una imagen importada a nuestro flash como muestra la siguiente figura.



En la capa “as” colocamos los siguientes códigos en los frames 1 y 2.

**Actionscript para el frame 1**:

stop();

var porcentaje = 0;

onEnterFrame = function () {

porcentaje = Math.round(getBytesLoaded()/getBytesTotal()\*100);

porcentaje\_txt.text = porcentaje + " %";

barra\_mc.\_xscale = porcentaje;

if (porcentaje == 100) {

delete (onEnterFrame);

play();

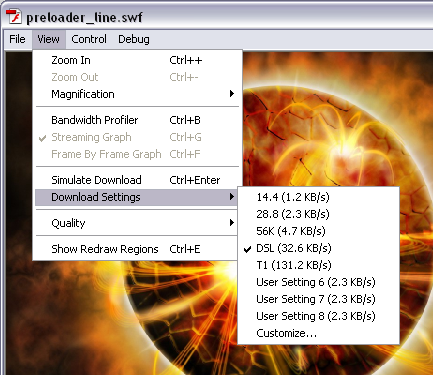
}

}

**Actionscript para el frame 2**:

stop();

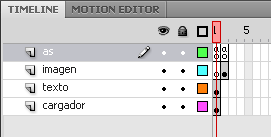
Para hacer pruebas de nuestro cargador presionamos CTRL+ENTER lo cual abrirá el visor de flash. Como el tiempo de descarga en un ambiente local es ínfimo debemos realizar una simulación de descarga en el visor. Para esto al cargar el visor de flash ubicamos en el menú VER la opción CONFIGURACIÓN DE DESCARGA y seleccionamos una dependiendo de la velocidad de descarga que queramos simular.



Luego presionamos de nuevo CTRL+ENTER y podremos ver el funcionamiento del precargador.

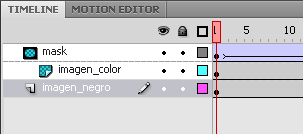
**Precargador en Línea de Tiempo.**

Para construir este precargador necesitaremos 4 capas como muestra la siguiente figura

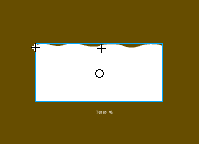


En la capa *texto* colocaremos un campo de texto dinámico, le damos nombre de instancia “porcentaje\_txt”, y tipeamos el texto 100% para darle las características de texto que sean de nuestro agrado.

En la capa *cargador* colocaremos un movieclip que contendrá la animación que mostrará nuestro precargador, esta animación será una interpolación de exactamente 100 frames que iremos recorriendo con actionscript a medida que se lleve a cabo la carga. El movieclip que tendrá nombre de instancia “cargador\_mc”, tendrá internamente las siguientes tres capas:

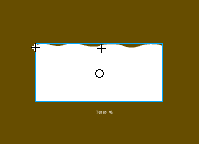


En las capa *imagen\_color* importaremos una imagen de nuestro interés. La misma imagen la usamos en la capa *imagen\_negro* pero le eliminamos el color (para esto podemos usar el efecto tinta); y en la capa *mask* usaremos un objeto parea enmascarar toda la imagen de color.

Capa *imagen\_ne*gro capa *imagen\_color* capa *mask*

La animación interna del movieclip será una interpolación de movimiento desde el frame 1 hasta el 100 al movieclip que usamos como máscara como indican las siguientes figuras

Frame 1 Frame 100

En la capa imagen de la línea de tiempo principal, en el frame 2, colocamos una imagen importada a nuestro flash como muestra la siguiente figura.



En la capa “as” de la línea de tiempo principal colocamos los siguientes códigos en los frames 1 y 2.

**Actionscript para el frame 1**:

stop();

var porcentaje = 0;

onEnterFrame = function(){

porcentaje = Math.round(getBytesLoaded() / getBytesTotal() \* 100);

porcentaje\_txt.text = porcentaje + " %";

cargador\_mc.gotoAndStop(porcentaje);

if(porcentaje == 100){

play();

delete(onEnterFrame);

}

}

**Actionscript para el frame 2**:

stop();

Finalmente probamos el cargador siguiendo los pasos de prueba del ejercicio anterior.

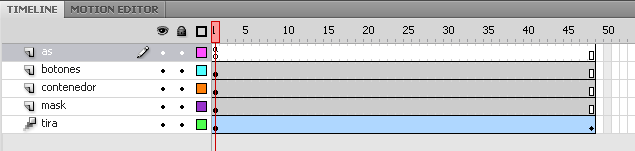
**Galería de fotos**

En este ejercicio crearemos una galería de fotos con un scroll de botones para llamar dinámicamente cada imagen de la galería, esto nos permitirá tener una galería más liviana y lás imágenes sólo son cargadas de un servidor si son invocadas por su correspondiente botón.

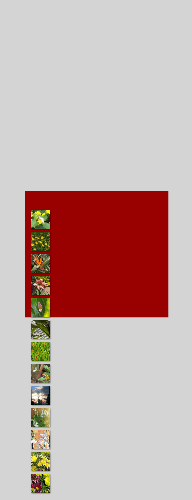
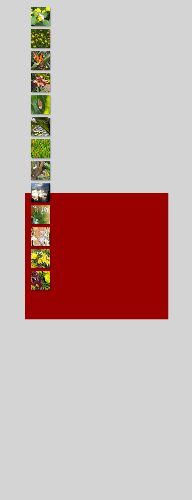
Para esto necesitamos las imágenes en tamaño original y a su vez una copia en pequeño de cada imagen para la creación de los botones.



En la línea de tiempo principal tendremos 4 capas para los objetos y una capa para el actionscript como se muestra en la siguiente figura.



En la capa inferior que llamaremos “tira”, vamos a construir un movieclip al que dremos nombre de instancia “tira” el cual contendrá una columna de botones creados a partir de las imágenes reducidas que importaremos a nuestra película, colocadas es orden de manera vertical y nombrándolos “boton1”, “boton2”, etc. Una vez creada la tira de botones, le aplicamos una interpolación de movimiento que le haga desplazarse verticalmente desde el frame 1 hasta el 48 como muestran las siguientes figuras.

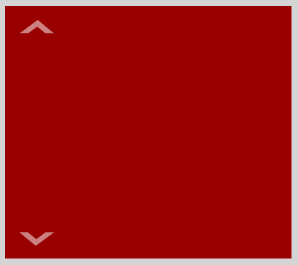
Frame 1 Frame48

En la capa *mask* colocamos un movieclip que contenga un recuadro con bordes superior e inferior degradados a alfa 0 como muestra la siguiente figura.



En la capa *contenedor* colocaremos un movieclip vacio con el nombre de instancia “contenedor” poniendo atención a su ubicación ya que dentro del mismo se cargarán dinámicamente las imágenes de la galería, por esto el movieclip vacío debe ser localizado donde queramos ubicar la esquina superior izquierda de las fotos.

Para terminar la preparación de nuestra galería colocamos en la capa *botones* un par de flechas como botones para subir y bajar la tira de botones como muestra la siguiente figura, al boton superior le damos nombre de instancia “subir” y al inferior “bajar”.



Botones

Una vez colocados todos los objetos con sus correspondientes nombres de instancia colocamos el siguiente código actionscript en la capa as de la línea principal.

**Actionscript para el frame 1**

stop();

tira.setMask(mascara);

bajar.onRollOver = function() {

onEnterFrame = function() {

nextFrame();

if (\_currentframe == 48) {

delete onEnterFrame;

}

}

}

bajar.onRollOut = function() {

delete onEnterFrame;

}

subir.onRollOver = function() {

onEnterFrame = function() {

prevFrame();

if (\_currentframe == 1) {

delete onEnterFrame;

}

}

}

subir.onRollOut = function() {

delete onEnterFrame;

}

// BOTONES PARA LAS IMAGENES

tira.boton1.onRelease = function() {

contenedor.loadMovie("fotos/foto1.jpg");

}

tira.boton2.onRelease = function() {

contenedor.loadMovie("fotos/foto2.jpg");

}

tira.boton3.onRelease = function() {

contenedor.loadMovie("fotos/foto3.jpg");

}

tira.boton4.onRelease = function() {

contenedor.loadMovie("fotos/foto4.jpg");

}

tira.boton5.onRelease = function() {

contenedor.loadMovie("fotos/foto5.jpg");

}

tira.boton6.onRelease = function() {

contenedor.loadMovie("fotos/foto6.jpg");

}

tira.boton7.onRelease = function() {

contenedor.loadMovie("fotos/foto7.jpg");

}

tira.boton8.onRelease = function() {

contenedor.loadMovie("fotos/foto8.jpg");

}

tira.boton9.onRelease = function() {

contenedor.loadMovie("fotos/foto9.jpg");

}

tira.boton10.onRelease = function() {

contenedor.loadMovie("fotos/foto10.jpg");

}

tira.boton11.onRelease = function() {

contenedor.loadMovie("fotos/foto11.jpg");

}

tira.boton12.onRelease = function() {

contenedor.loadMovie("fotos/foto12.jpg");

}

tira.boton13.onRelease = function() {

contenedor.loadMovie("fotos/foto13.jpg");

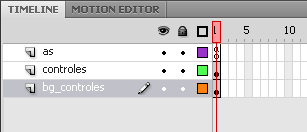
}

Con esto ya tendremos una galería funcional, pero para el correcto funcionamiento debemos tener las fotos en una carpeta esterna situada en el mismo directorio donde coloquemos nuestro archivo publicado. Los llamados a las imágenes del código actionscript invocan la ruta “fotos/”, por lo que la carpeta de fotos debe tener ese nombre para que puedan ser encontradas y cargadas en el SWF.

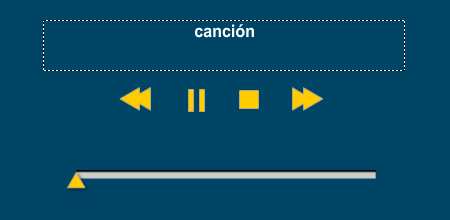
**Reproductor MP3**

En este ejercicio crearemos un reproductor mp3 con cinco canciones preestablecidas, nuestro reproductor nos dará datos de las canciones y nos permitirá controlar el sonido y el volumen.

En la línea de tiempo principal tendremos tres capas como indica la siguiente figura.



En la capa controles colocaremos los siguientes objetos:



En primer lugar tendremos un campo de texto dinámico donde cargaremos el nombre de la canción y de su intérprete, a este campo le damos nombre de instancia “info”, lo configuramos como multilinea en el panel de propiedades, con el tamaño y características de nuestro agrado.



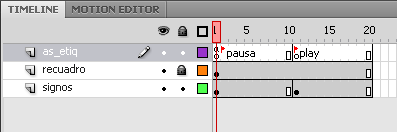
Ahora colocamos los movieclips de control de sonido, como se muestran a continuación con los nombres de instancia indicados.

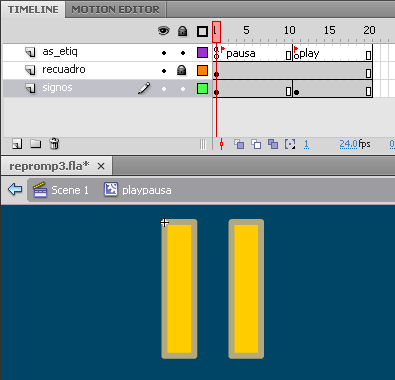
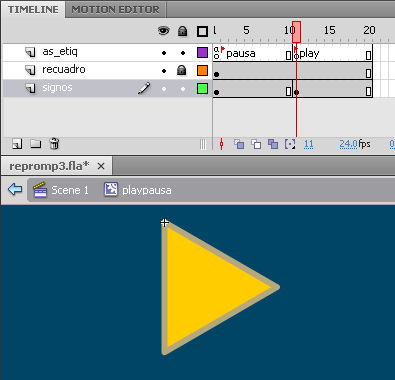
movieclip: “posterior” movieclip: “playpause” movieclip: “parar” movieclip: “proximo”

El movieclip playpause sera particular, pues el mismo movieclip nos permitirá mostrar el botón de “play” y el de “pausa” dependiendo del tiempo de ejecución del sonido, cuando el sonido este activo mostrará el signo de pausa, cuando el sonido este detenido mostrará el signo de play.

Para ésto creamos el movieclip armando el signo de pausa, lo convertimos en movieclip, y lo llamamos “playpausa”. Luego lo editamos y en su línea de tiempo le damos a la capa donde esta el signo de pausa el nombre “signos”. Adicionalmente creamos dos capas superiores como muestra la figura.



En la capa signo en el frame ya se encuentra el signo, luego en esta misma capa en el frame 11 presionamos F7 para crear un keyframe vacío, donde colocaremos el signo de play como se muestra en las siguientes figuras.

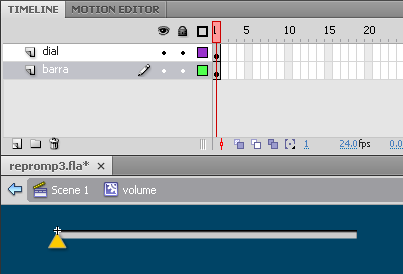
 

Frame 1 Frame 11

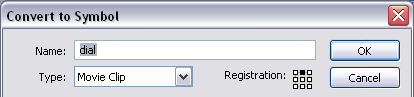
Como se ve en la figura anterior le damos permanencia al signo de pausa hasta el frame 10 y al signo de play hasta el frame 20.

En una capa nueva llamada recuadro colocamos un recuadro de alfa=0 que recubra ambas figuras y le damos permanencia hasta el frame 20. Adicionalmente colocamos las etiquetas que se ven en la línea de tiempo, esto se aplica seleccionando el frame 1 en la capa *as\_etiq* y en el panel de propiedades rellenamos el campo ETIQUETA con el nombre “pausa”, igualmente nos posicionamos en la misma capa en el frame 11 y le damos el nombre de etiqueta “play”. Finalmente colocamos el actionscript, únicamente en el frame 1 de la misma capa abrimos el panel de acciones y le asignamos un *stop();*.

Una vez listo el botón de play/pausa continuamos con los controles. Para contruir la barra de volumen iniciamos creando un rectángulo de 300 px de ancho y lo convertimos en el movieclip “volume”. Luego lo editamos y a la capa donde está el rectángulo la llamamos “barra”. Agregamos una nueva capa y en ella dibujamos un triángulo como muestra la figura.



Luego convertimos el triángulo en movieclip con el nombre “dial” y asegurándonos de colocar el punto de registro en la parte central superior como muestra la siguiente figura. La posición en el eje X se ajustará con actionscript pero la posición en el eje Y debemos fijarla dentro del movieclip.



Una vez listo nuestro movieclip de sonido salimos a la escena 1 y le damos el nombre de instancia “barra”.

En la capa bg\_controles colocaremos rectángulos de fondo para cada sección, uno para el campo de texto, uno para los controles, y un último para la barra de volumen.

Teniéndo ya los objetos listos procedemos con el actionscript en la capa *as*.

**Actionscript para el frame 1**

//creamos las variables

var sonido:Sound = new Sound();

var cantidad:Number = 5;

var pos:Number = 0;

//arranca de una vez la primera canción

i = 1;

sonido.loadSound("musica/cancion"+i+".mp3",true);

sonido.onID3 = function(){

nombre = sonido.id3.songname;

artista = sonido.id3.artist;

info.text = nombre+"\r"+artista;

}

sonido.start();

//boton proximo

proximo.onRelease = function(){

if (i<cantidad) {

i = i+1;

sonido.loadSound("musica/cancion"+i+".mp3",true);

sonido.start();

}

}

//boton anterior

anterior.onRelease = function(){

if (i>1) {

i = i-1;

sonido.loadSound("musica/cancion"+i+".mp3",true);

sonido.start();

}

}

//boton stop

parar.onPress = function(){

pos = 0;

sonido.stop();

playpause.gotoAndStop("play");

}

//boton play pausa

playpause.onRelease = function(){

if (this.\_currentframe==1) {

this.gotoAndStop("play");

pos = sonido.position;

sonido.stop();

} else {

this.gotoAndStop("pausa");

sonido.start(pos/1000);

}

}

//dial de volumen

barra.dial.\_x = 150;

barra.dial.onEnterFrame = function(){

sonido.setVolume(0-(this.\_x/3));

}

barra.dial.onPress = function(){

startDrag(this, false, 0, this.\_y, 300, this.\_y);

}

barra.dial.onRelease = barra.dial.onReleaseOutside = function(){

stopDrag();

}

**Formulario de Contacto**

En este ejercicio crearemos un formulario de contacto, con un conjunto campos de entrada de texto para agregar datos que posteriormente serán enviados mediante un email a una dirección preestablecida y finalmente arroje un mensaje de respuesta.

|  |  |
| --- | --- |
| Preparamos el escenario como muestra la figura a la derecha, donde fundamentalmente necesitamos tres campos de entrada de texto a los cuales damos los siguientes nombres de instancia: “nombre”, “correo” y “respuesta”.  Los dos primeros los configuramos de línea única, y el tercero será multilínea (esto se realiza en el panel de propiedades).  Adicionalmente un campo de texto dinámico donde escribiremos la respuesta obtenida del envío del email. Y finalmente un botón para ejecutar el envío de los datos.  *NOTA*: Flash por sí mismo no realizará el envío del e-mail, tan sólo pasará los datos a una página PHP (o ASP) la cual si estará en comunicación con un servidor para realizar dicho envío, recibirá una respuesta y la transmitirá de regreso a nuestro SWF.  Las animaciones de entrada de los campos son opcionales en la creación del formulario. |  |

Una vez que tenemos todos los objetos necesarios arriba indicados procedemos a colocar el código actionscript.

**Actionscript del frame 44**

stop();

//activamos el botón

enviar.onRelease = function() {

//con esto impedimos que dejen algún cuadro vacío

if (length(nombre.text)<5) {

respuesta.text = "Escriba su Nombre";

}

else if (length(correo.text)<5) {

respuesta.text = "Escriba su Correo";

}

else if (length(mensaje.text)<10) {

respuesta.text = "Escriba un Comentario";

}

else {

//si se cumplen todas las condiciones creamos

//la variable que envía los datos con LoadVars

var datos = new LoadVars();

datos.nombre = nombre.text;

datos.email = correo.text;

datos.mensaje = mensaje.text;

//conectamos con el archivo de php que envía el correo

datos.sendAndLoad("contacto.php",datos,"POST");

//si no hubo inconvenientes nos da las gracias

datos.onLoad = function() {

delete datos;

respuesta.text = "Gracias por su comentario";

}

}

}

Finalmente publicamos nuestra película y generamos el SWF. Este deberá estar localizado en el mismo directorio donde se encuentre la página ***contacto.php*** que efectúa el envio del correo.