



JavaScript

Sesión 8 - jQuery UI



Índice

- Descarga y Configuración
- Componentes
 - Acordeón
 - Pestañas
 - Botón
 - *Tooltip*
 - Diálogo
 - Menú
 - Selector de Fecha
- Temas
- Efectos
- Interacciones



8.1 jQuery UI

- Plugin de *jQuery*
- <http://jqueryui.com/>
- Funcionalidades de interfaz de usuario que ya están construidas
- Se dividen en:
 - interacciones
 - widgets (componentes)
 - efectos
 - utilidades.



Descarga

- El plugin contiene
 - La librería
 - La hoja de estilo
 - Un conjunto de imágenes
- Ofrece la posibilidad de descargar la parte del plugin en el que estemos interesado.

Download Builder

Quick downloads: [Stable \(Themes\) \(1.11.1: for jQuery1.6+\)](#) | [Legacy \(Themes\) \(1.10.4: for jQuery1.6+\)](#) | [Legacy \(Themes\) \(1.9.2: for jQuery1.6+\)](#)
[All jQuery UI Downloads](#)

Version

1.11.1 (Stable, for jQuery1.6+)

1.10.4 (Legacy, for jQuery1.6+)

1.9.2 (Legacy, for jQuery1.6+)

Components

Toggle All

UI Core <input checked="" type="checkbox"/> Toggle All A required dependency, contains basic functions and initializers.	<input checked="" type="checkbox"/> Core <input checked="" type="checkbox"/> Widget <input checked="" type="checkbox"/> Mouse <input checked="" type="checkbox"/> Position	The core of jQuery UI, required for all interactions and widgets. Provides a factory for creating stateful widgets with a common API. Abstracts mouse-based interactions to assist in creating certain widgets. Positions elements relative to other elements.
Interactions <input checked="" type="checkbox"/> Toggle All These add basic behaviors to any element and are used by many components below.	<input checked="" type="checkbox"/> Draggable <input checked="" type="checkbox"/> Droppable <input checked="" type="checkbox"/> Resizable <input checked="" type="checkbox"/> Selectable	Enables dragging functionality for any element. Enables drop targets for draggable elements. Enables resize functionality for any element. Allows groups of elements to be selected with the mouse.



Ejemplos

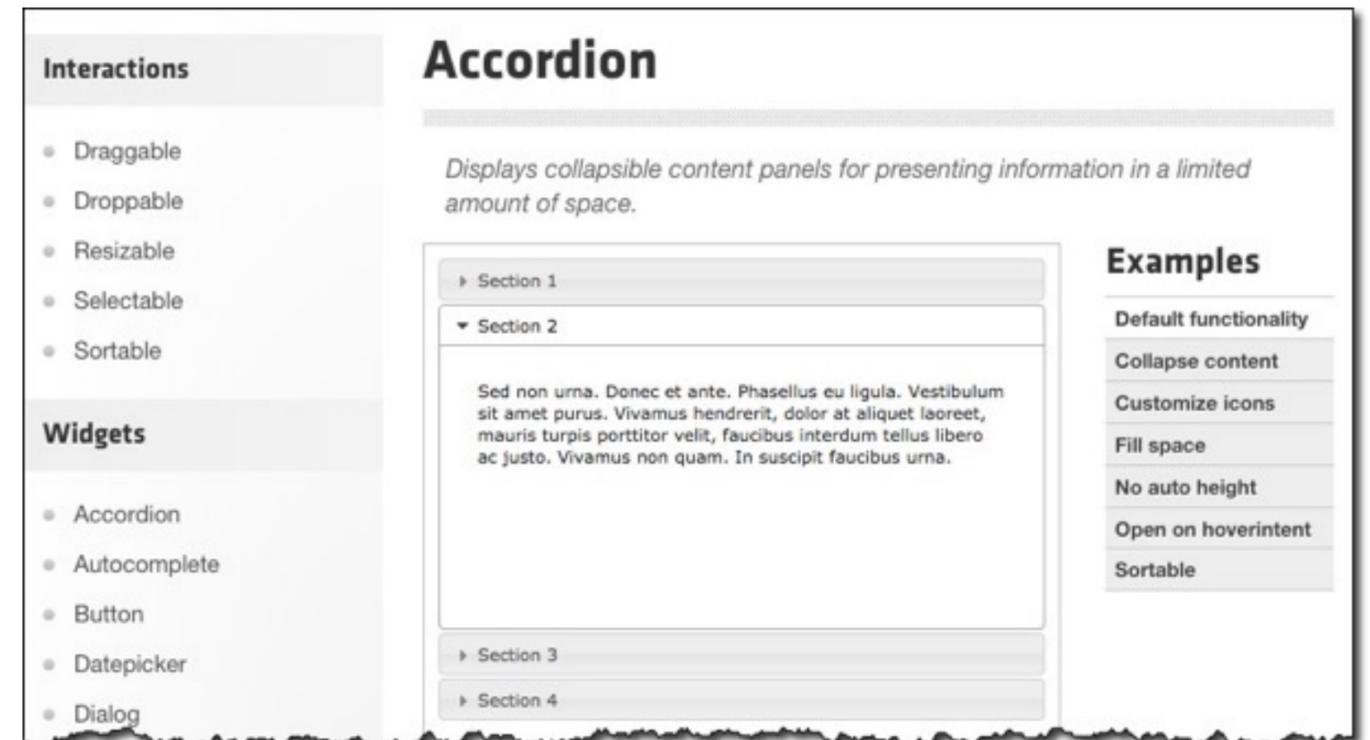
- Al descargar la librería, además de los fuentes y las versiones minificadas, hay un archivo de ejemplo de los controles.

The screenshot shows a collection of jQuery UI widgets. On the left, there are three sections: Section 1 (with a warning icon and text), Section 2, and Section 3. Below these are tabs labeled 'First', 'Second', and 'Third'. A dialog box is shown with an 'Open Dialog' button. At the bottom, there's an 'Overlay and Shadow Classes' section with a semi-transparent box over text. On the right side, there are several widgets: a button labeled 'A button element', three choice buttons ('Choice 1', 'Choice 2', 'Choice 3'), an 'Autocomplete' text input, a 'Spinner' input, a 'Slider' control, a 'Datepicker' showing 'September 2014' with a calendar grid, and a 'Progressbar'.



8.2 Componentes (*Widgets*)

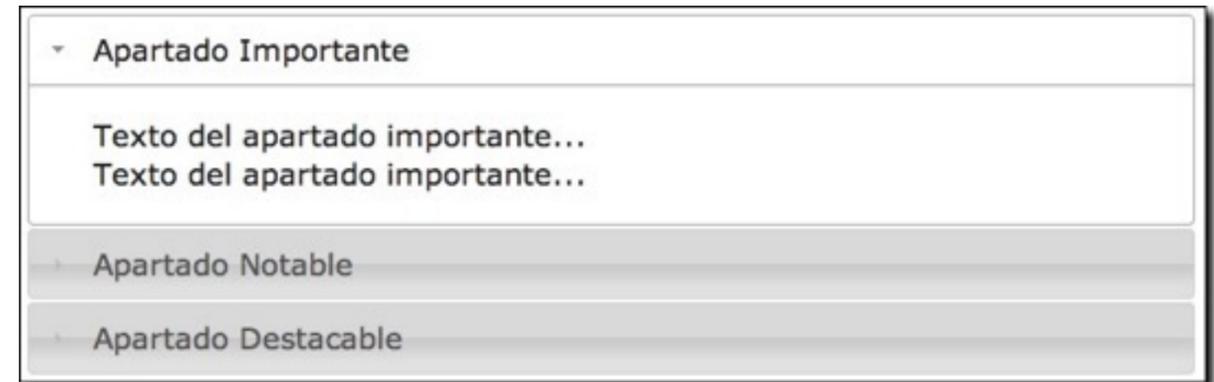
- jqueryui.com → panel lateral izquierdo → seleccionar control
 - Permite visualizar ejemplos con código
- Todos los controles comparten algunas características
 - propiedades de configuración
 - métodos
 - eventos.
- Al aprender un control, estamos aprendiendo un poco de todos.





8.2.1 Acordeón

- `accordion ()`
- <http://api.jqueryui.com/accordion> - <http://jqueryui.com/accordion/>
- Muestra información de forma colapsada
 - Reduce el tamaño de espacio necesario
- Al seleccionar otro apartado, el contenido visible se oculta mediante una animación.
- La información debe estar agrupada en cabeceras (`h1 ... h6`) y capas, dentro de una capa padre que englobe todo el acordeón.



<http://jsbin.com/fuwobu/1/edit?html,js,output>

```
$( "#contenido" ).accordion( );
```

```
<div id="contenido">
  <h3>Apartado Importante</h3>
  <div>Texto del apartado importante...<br />Texto del apartado importante...</div>
  <h3>Apartado Notable</h3>
  <div>Texto del apartado notable...<br />Texto del apartado notable...</div>
  <h3>Apartado Destacable</h3>
  <div>Texto del apartado destacable...<br />Texto del apartado destacable...</div>
</div>
```



Configuración Acordeón I

- Para configurar un control, **cualquier que sea**, el control puede recibir hasta tres parámetros:
 - un objeto con las opciones de configuración
 - eventos que disparan el control
 - *callbacks* tras su finalización

Propiedad	Propósito
active	Indica el panel del acordeón que aparece desplegado, por defecto es el primero. Como valor se puede usar una cadena a modo de selector sobre las cabeceras, un objeto <i>jQuery</i> , un objeto <i>HTMLElement</i> o un número (0... N-1).
header	Selector para el elemento cabecera, al cual se le aplica el método <code>find()</code> dentro del objeto <i>jQuery</i> . Los paneles de contenido serán los hermanos inmediatos (el que esté a su derecha).
animate	<ul style="list-style-type: none">• Booleano que indica si se usa una animación• número entero con los milisegundos de la animación• o una cadena con diferentes posibilidades de animación (<code>slide</code>, <code>easeOutBounce</code>, <code>linear</code>, etc..). Para indicar varias de estas posibilidades, crear un objeto como valor de la propiedad



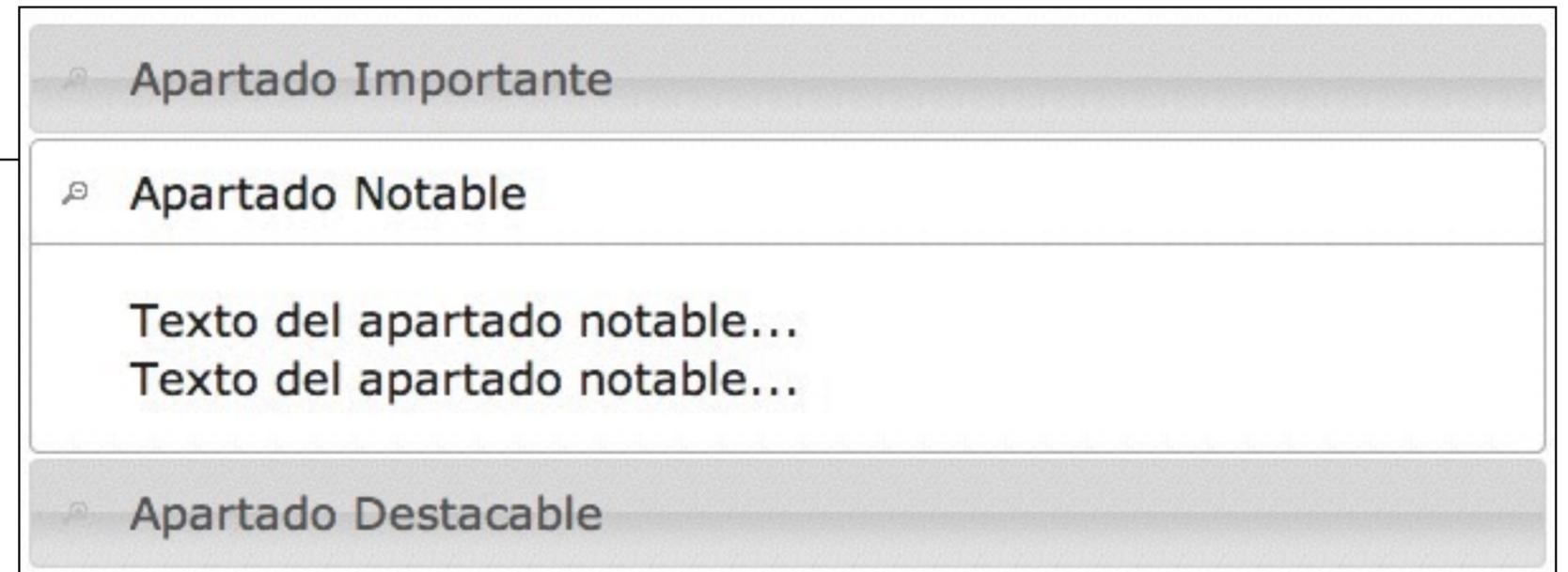
Configuración Acordeón II

Propiedad	Propósito
event	Indica el evento sobre el encabezado que hace desplegar y recoger el acordeón. Por defecto es <code>click</code>
collapsible	Booleano para indicar si se pueden cerrar todos los paneles. Para que al cargar el acordeón aparezca completamente cerrado, necesitamos esta propiedad a <code>true</code> y la propiedad <code>active</code> a <code>false</code> .
autoHeight	Booleano para indicar si la altura de los elementos serán los del elemento más alto (todos igual de altos). Cuando cambia mucho la altura entre unos elementos y otros es mejor dejarlo a <code>false</code> y ajustar la propiedad <code>fillSpace</code> . Por defecto es <code>true</code>
fillSpace	Booleano para indicar si rellena la altura del elemento padre, y en el caso de que el contenido sea mayor que la altura, mostrará una barra de scroll. Por defecto es <code>false</code> , lo que significa que la altura del acordeón se basa en el contenido de los elementos
icons	Permite configurar los iconos a usar, mediante un objeto con las propiedades <code>header</code> y <code>activeHeader</code>



Ejemplo Acordeón

```
$( "#contenido" ).accordion({
  active: 1,
  animate: {
    easing: "linear",
    duration: 100,
    down: {
      easing: "easeOutBounce",
      duration: 300
    }
  },
  collapsible: true,
  event: "click mouseover",
  icons: {
    header: "ui-icon-zoomin",
    activeHeader: "ui-icon-zoomout" // en version anteriores selectedHeader
  }
});
```



<http://jsbin.com/xfaha/1/edit?html,js,output>



Métodos / Operaciones

Método	Propósito
enable	Habilita el componente acordeón
disable	Deshabilita el componente acordeón
destroy	Elimina la funcionalidad del acordeón
option	Asigna una o más opciones. Mediante <code>option</code> podemos configurar todas las propiedades de configuración del apartado anterior.
resize	Modifica el tamaño del acordeón cuando se usa la propiedad <code>fillSpace</code> y el tamaño del elemento padre cambia. Esta característica es mejor usarla con la interacción de redimensionar.

- Las 4 primeros comunes a todos los controles jQuery UI
- Se pasan como una cadena al método del control.
- Si tienen parámetros, se pasan como parámetros posteriores al del método

```
$( "#contenido" ).accordion( "disable" );  
$( "#contenido" ).accordion( "option", "active", 1 );
```



Eventos

- Se añaden dentro del objeto de configuración como una propiedad
- O capturando el evento mediante el método `on` de *jQuery*

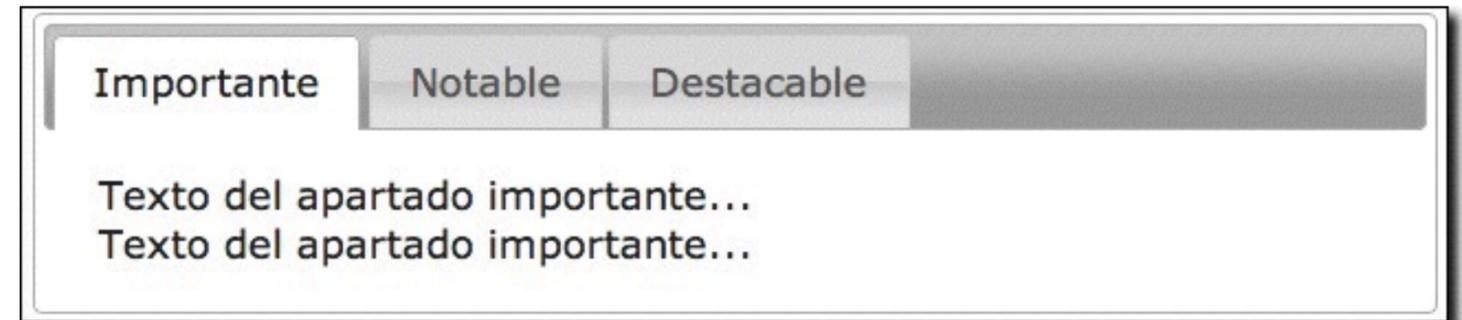
Evento	Tipo	Propósito
<code>create(evento, ui)</code>	<code>accordioncreate</code>	Se lanza al crear el acordeón
<code>activate(evento, ui)</code>	<code>accordionactivate</code>	Se lanza tras cambiar el elemento activo
<code>beforeActivate(evento, ui)</code>	<code>accordionbeforeactivate</code>	Se lanza antes de cambiar el elemento activo

```
$( "#contenido" ).accordion({
  create: function(evt, ui) {
    console.log( "Acordeón creado con la cabecera activa " + ui.header );
  }
});

$( "#contenido" ).on( "accordioncreate", function(evento, ui) {
  console.log( "Acordeón creado con la cabecera activa " + ui.header );
});
```



8.2.2 Pestañas



- `tabs()`
- <http://api.jqueryui.com/tabs> - <http://jqueryui.com/tabs>
- Añade pestañas para organizar el contenido.
- La estructura HTML debe estar formada por:
 - una lista desordenada con un enlace que referencie al `id` de cada capa e indique su nombre
 - una capa por cada pestaña con un `id`.

```
$( "#contenido" ).tabs();
```

```
<div id="contenido">
  <ul>
    <li><a href="#importante">Importante</a></li>
    <li><a href="#notable">Notable</a></li>
    <li><a href="#destacable">Destacable</a></li>
  </ul>
  <div id="importante">Texto del apartado importante...<br />Texto del apartado importante...</div>
  <div id="notable">Texto del apartado notable...<br />Texto del apartado notable...</div>
  <div id="destacable">Texto del apartado destacable...<br />Texto del apartado destacable...</div>
</div>
```

<http://jsbin.com/larece/1/edit?html,js,console,output>



Configuración Pestañas

Propiedad	Propósito
active	Indica la pestaña que aparece abierta, por defecto la primera. Como valor se puede usar una cadena a modo de selector sobre las cabeceras, un objeto <i>jQuery</i> , un objeto <i>HTML Element</i> o un número (0... N-1).
show	Booleano que indica si se usa una animación al mostrar el contenido, número entero con los milisegundos de la animación, o una cadena con diferentes posibilidades de animación (<code>slide</code> , <code>easeOutBounce</code> , <code>linear</code> , etc..). Si queremos indicar varias de estas posibilidades podemos crear un objeto como valor de la propiedad
hide	Similar a <code>show</code> pero para ocultar el contenido
event	Indica el evento sobre los tabuladores que hace abrir la pestaña activa. Por defecto es <code>click</code>
collapsible	Booleano para indicar si se pueden cerrar todas las pestañas.
disabled	Permite activar o desactivar todas las pestañas a la vez mediante un booleano o mediante un array para indicar cuales deshabilitar, por ejemplo, <code>[0, 2]</code> deshabilitará la primera y la tercera

```
$( "#contenido" ).tabs( {  
  active: 3,  
  hide: "explode"  
} );
```



Métodos Pestañas

- `enable`, `disable`, `destroy`, `option`

Método	Propósito
<code>load</code>	Carga el contenido de una determinada pestaña al pasarle un valor numérico o de manera remota mediante una cadena que representa una <i>URL</i>
<code>refresh</code>	Procesa cualquier pestaña que ha sido añadida o eliminada vía DOM y recalcula la altura de los paneles.

```
function anyadirPestanya(titulo, texto) {  
    var numTabs = $("#contenido ul li").length + 1;  
  
    $("#contenido ul").append("<li><a href='#tab' +numTabs+ ''>" +titulo+ "</a></li>");  
    $("#contenido").append("<div id='tab' +numTabs+ ''>" +texto+ "</div>");  
    $("#contenido").tabs("refresh");  
}
```

<http://jsbin.com/muxuno/1/edit?html,js,output>



Pestañas vía AJAX

- Mediante método `load`
- O el enlace de definición de las pestañas referenciar a una URL
- El control automáticamente cargará dicho contenido al seleccionar la pestaña

```
<div id="contenido">  
  <ul>  
    <li><a href="contenidoHTML.html">Contenido HTML</a></li>  
    <li><a href="#importante">Importante</a></li>  
    ...  
  </ul>  
</div>
```





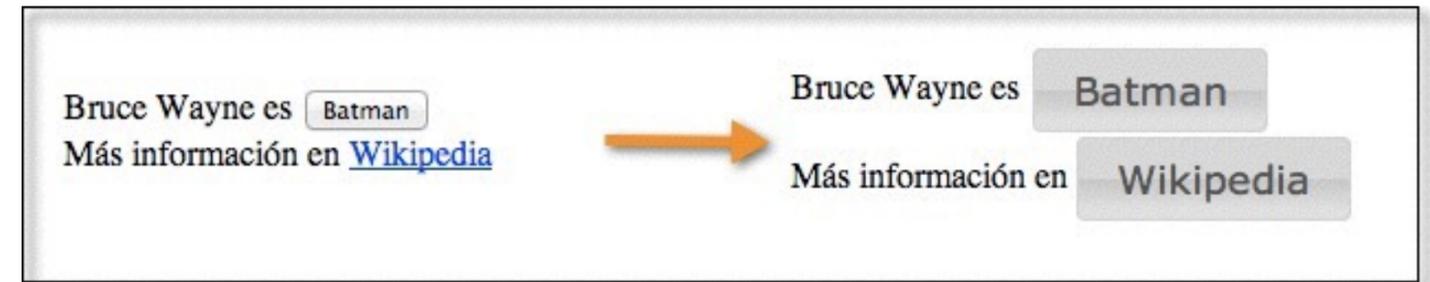
Eventos Pestañas

- `create` (tipo `tabscreate`), `activate` (tipo `tabsactivate`) y `beforeActivate` (tipo `tabsbeforeactivate`).

Evento	Tipo	Propósito
<code>beforeLoad(evento, ui)</code>	<code>tabsbeforeload</code>	Se lanza antes de cargar una pestaña remota
<code>load(evento, ui)</code>	<code>tabsload</code>	Se lanza tras cargar una pestaña remota



8.2.3 Botón



- `button()`
- <http://api.jqueryui.com/button>
- Control sencillo que repercute en el aspecto y funcionalidad de los documentos.
- Se aplica tanto a los botones (de envío, *radio*, *checkboxes*, etc...) como a los enlaces
- Les aplica el interfaz del tema de *jQuery UI*
 - tamaño, forma, fuente y color que define la hoja de estilos.

```
<label for="batman">Bruce Wayne es</label>  
<button id="batman">Batman</button> <br />  
Más información en <a href="http://es.wikipedia.org/wiki/Batman">Wikipedia</a>
```

```
$( "button, a" ).button();
```



Configuración Botón

Propiedad	Propósito
disabled	Booleano que deshabilita el botón
icons	Objeto con los iconos a mostrar, con o sin texto (opción <code>text</code>). Permite añadir un icono a la izquierda del texto (propiedad <code>primary</code>) o a la derecha (propiedad <code>secondary</code>). Podemos ver los iconos posibles en http://api.jqueryui.com/theming/icons
label	Cadena que muestra el botón. Cuando no se especifica, se usa el contenido HTML o el atributo <code>value</code> .
text	Booleano para mostrar el texto del botón

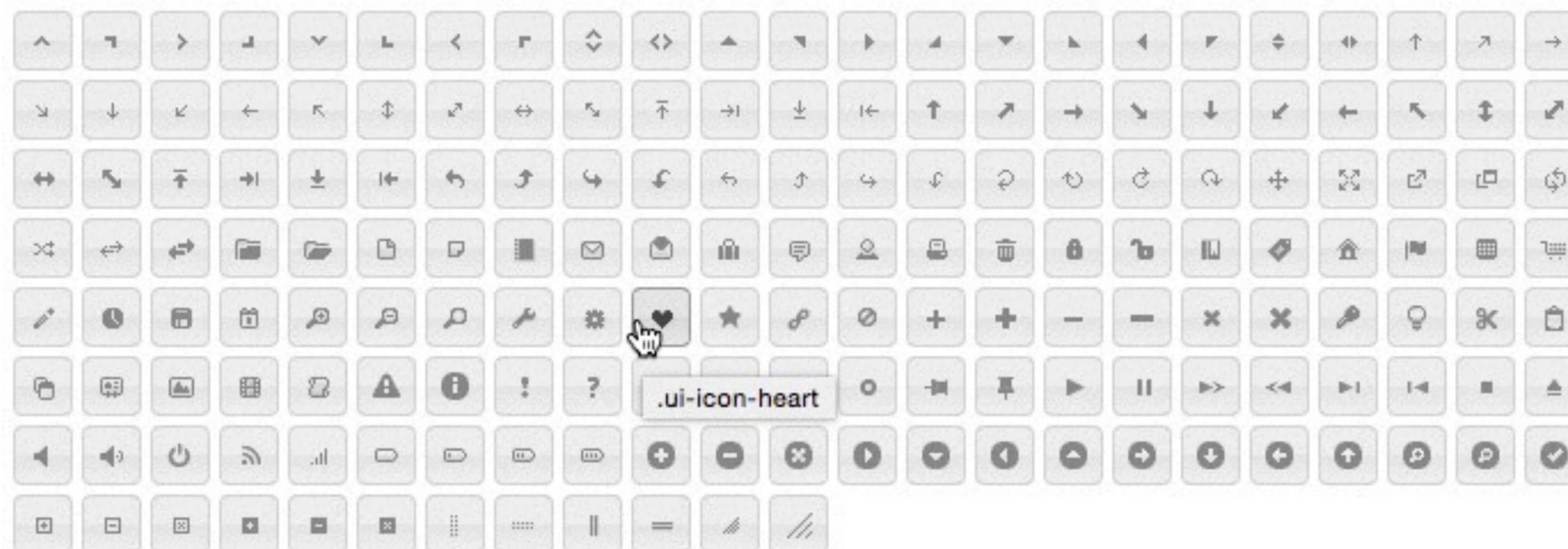
```
$( "a" ).button( {  
  disabled: true  
} );  
$( "button" ).button( {  
  label: "Texto desde configuración",  
  icons: { primary: "ui-icon-circle-plus" }  
} );
```





Iconos

- <http://api.jqueryui.com/theming/icons/>
- *jQuery UI* ofrece un gran conjunto de iconos que podemos incluir en la mayoría de controles.
- Nomenclatura → comienzan por `ui-icon`, seguido de un sufijo relativo al significado del icono.
- Iconos disponibles → <http://jqueryui.com/themeroller/>
 - Al dejar el cursor encima de un icono aparecerá el nombre del mismo.



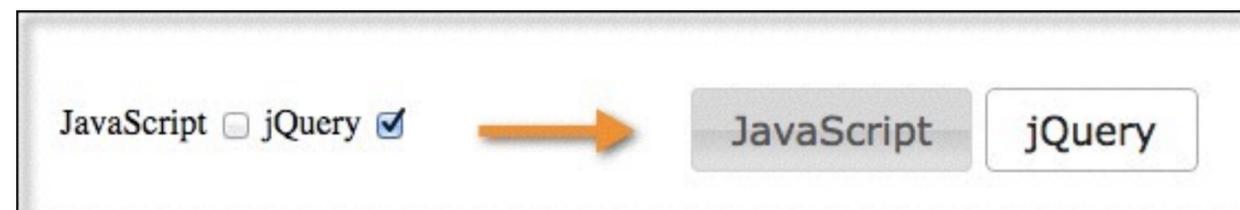


Botón *Toggle*

- 2 estados
- campos de tipo `checkbox` donde su `id` coincida con el atributo `for` de su `label`

```
<label for="check1">JavaScript</label>  
<input id="check1" type="checkbox" name="checks" value="1" />  
<label for="check2">jQuery</label>  
<input id="check2" type="checkbox" name="checks" value="2" />
```

```
$( "[type='checkbox']" ).button( );
```





Conjunto de Botones

- `buttonset()`
- <http://api.jqueryui.com/buttonset>
- Botones `radio`
- Selección excluyente
- Agrupar los botones dentro de una capa

```
<div id="conjunto">  
  <label for="radio1">JavaScript</label>  
  <input id="radio1" type="radio" name="rads" value="1" />  
  <label for="radio2">jQuery</label>  
  <input id="radio2" type="radio" name="rads" value="2" />  
  <label for="radio3">Angular</label>  
  <input id="radio3" type="radio" name="rads" value="3" />  
</div>
```

```
$( "#conjunto" ).buttonset( );
```

JavaScript jQuery Angular



JavaScript

jQuery

Angular



8.2.4 *Tooltip*

- tooltip()
- <http://api.jqueryui.com/tooltip>
- Sustituye los tooltips nativos.
- Configurables por el tema del widget
- Se puede:
 - ajustar el estilo (mensajes de alerta o error)
 - configurar la posición
 - añadir contenido vía AJAX.
- Cualquier elemento con atributo `title`



```
$( "#img" ).tooltip( {track:true} );
```

```

```



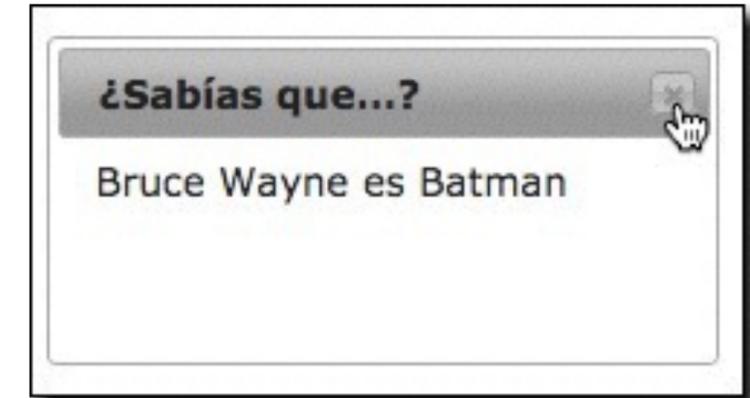
Configuración *Tooltip*

Propiedad	Propósito
content	Cadena o función que devuelve el contenido del <i>tooltip</i>
items	Selector que indica que elementos se mostrarán en el <i>tooltip</i> . Por defecto es <code>[title]</code> . Al modificar esta propiedad, hay que modificar <code>content</code> también para indicarle como obtener el contenido.
disabled	Booleano que deshabilita el botón
show	Booleano que indica si se usa una animación al mostrar el contenido, número entero con los milisegundos de la animación, o una cadena con diferentes posibilidades de animación (<code>slide</code> , <code>easeOutBounce</code> , <code>linear</code> , etc..). Si queremos indicar varias de estas posibilidades podemos crear un objeto como valor de la propiedad
hide	Similar a <code>show</code> pero para ocultar el contenido
position	Objeto que define la posición del <i>tooltip</i> respecto al elemento asociado
tooltipClass	Cadena con la clase CSS que se le añade al control para poder mostrar diferentes tipos de <i>tooltips</i> , como advertencias o errores.
track	Booleano que indica si el control debe seguir al ratón



8.2.5 Diálogo

- `dialog()`
- <http://api.jqueryui.com/dialog>
- Permite crear diálogos flotantes, modales o no, modificables en posición y tamaño con la misma apariencia del tema de *jQuery UI*.
- A partir de una capa contenedora
- Al crear una diálogo sencillo, por defecto, se va a mostrar automáticamente en el centro de la pantalla y vamos a poder cambiar tanto su tamaño como su posición.
 - Si el contenido supera el tamaño del diálogo, aparecerá una barra de *scroll*.



```
$( "#dialogo" ).dialog( );
```

```
<div id="dialogo" title="¿Sabías que...?">Bruce Wayne es Batman</div>
```



Configuración Diálogo I

Propiedad	Propósito
autoOpen	Booleano para indicar si se abre el diálogo tan pronto como se invoque. Por defecto es <code>true</code>
title	Título del diálogo
resizable	Booleano para indicar si el diálogo permite modificar su tamaño. Por defecto es <code>true</code> .
draggable	Booleano para indicar si el diálogo puede moverse arrastrándolo por la pantalla. Por defecto es <code>true</code> .
appendTo	Selector que indica a qué elemento se anexa el diálogo (y se superpondrá si es modal)
buttons	Objeto o array que especifica el conjunto de botones que se añaden al diálogo y las funciones que se invocarán al pulsar dichos botones. Por defecto no se muestran botones. Cada uno de los elementos define las propiedades <code>text</code> para el texto del botón y <code>click</code> para el <i>callback</i>
modal	Booleano que indica si el diálogo es modal y se superpone por encima del resto de elementos que se deshabilitan
closeOnEscape	Booleano que especifica si el diálogo se cierra al pulsar la tecla ESC. Por defecto es <code>true</code>
closeText	Cadena que especifica el texto del botón de cierre. Destacar que este texto aparece oculto al usar el tema estándar.



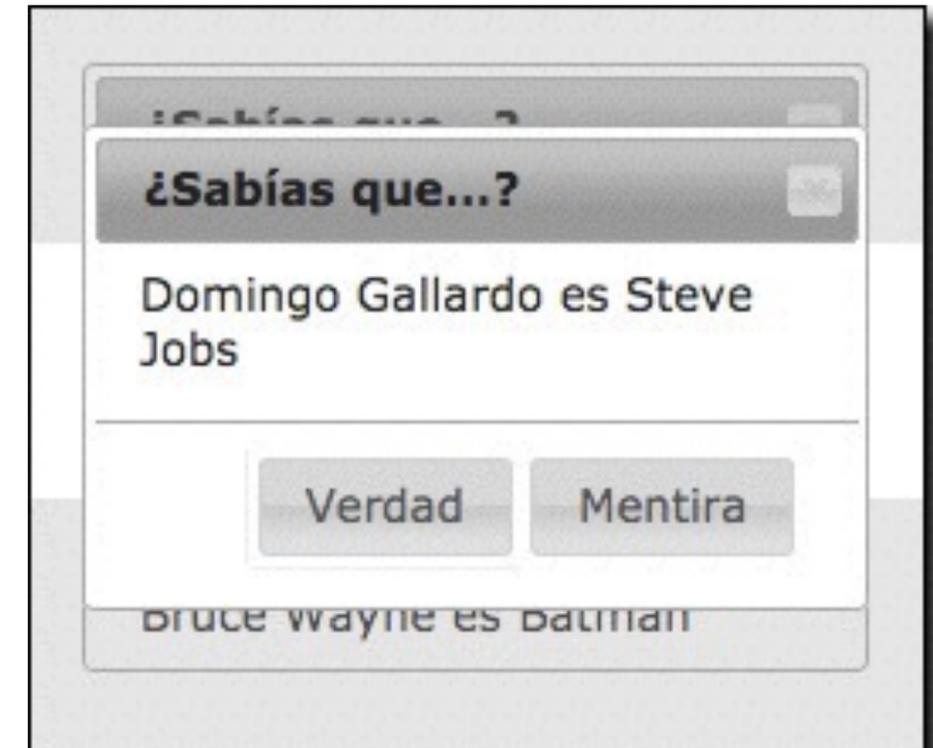
Configuración Diálogo II

Propiedad	Propósito
dialogClass	Cadena que se añadirá como clase CSS al diálogo, para extender el tema.
show	Booleano que indica si se usa una animación al mostrar el contenido, número entero con los milisegundos de la animación, o una cadena con diferentes posibilidades de animación (<code>slide</code> , <code>easeOutBounce</code> , <code>linear</code> , etc..). Si queremos indicar varias de estas posibilidades podemos crear un objeto como valor de la propiedad
hide	Similar a <code>show</code> pero para ocultar el contenido
height	Altura del diálogo, mediante un número que indica la cantidad de píxeles, o mediante <code>auto</code> para que sea ajustable al contenido
width	Anchura del diálogo, mediante un número que indica la cantidad de píxeles, o mediante <code>auto</code> para que sea ajustable al contenido
maxHeight, minHeight	Número que indica la altura máxima o mínima que puede tener el control.
maxWidth, minWidth	Número que indica la anchura máxima o mínima que puede tener el control.
position	Objeto que define la posición del <i>tooltip</i> respecto al elemento asociado



Ejemplos Diálogos

```
$( "#dialogoBoton" ).dialog({
  autoOpen: false
});
$( 'button' ).button().click(function(e) {
  $( '#dialogoBoton' ).dialog( "open" )
});
```

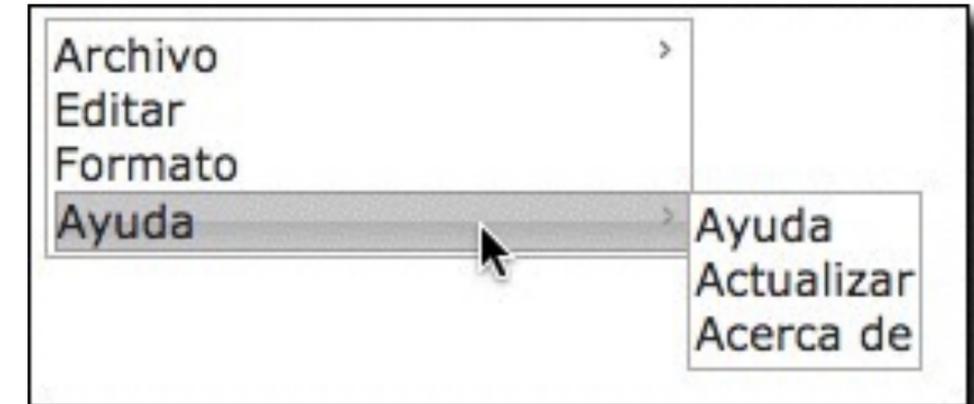


```
$( "#dialogoConf" ).dialog({
  draggable: false,
  modal: true,
  closeOnEscape: false,
  buttons: [
    {text: "Verdad", click: function() {
      console.log("Me alegro que también lo supieses"); }}},
    {text: "Mentira", click: function() {
      $(this).dialog("close"); }}
  ]
});
```



8.2.6 Menús

- `menu()`
- <http://api.jqueryui.com/menu> -
- Permite crear un menú con submenús siguiendo la apariencia del resto de controles.
- Responde a eventos tanto de teclado como de ratón
- Necesita estructura HTML con relación padre-hijo
 - Normalmente se usan listas desordenadas
- Por defecto, el menú coge al anchura de la página
 - Limitar vía CSS



```
#menu {  
  width: 250px;  
}
```

```
$( "#menu" ).menu( );
```

```
<ul id="menu">  
<li>Archivo  
  <ul>  
    <li>Abrir</li>  
    <li>Guardar</li>  
    <li>Salir</li>  
  </ul>  
</li>  
<li>Editar</li>  
<li>Formato</li>  
<li>Ayuda  
  <ul>  
    <li>Ayuda</li>  
    <li>Actualizar</li>  
    <li>Acerca de</li>  
  </ul>  
</li>  
</ul>
```



Configuración Menús

```
<li><span class="ui-icon ui-icon-disk"></span>Guardar</li>
```

- Para mostrar un icono → css `ui-icon`
- Para deshabilitar un elemento → css `ui-state-disabled`
- Para introducir un elemento separador → crear elementos que contengan sólo espacios o guiones

```
<ul id="menu">  
  <li>Elemento 1</li>  
  <li>-</li>  
  <li>Elemento 2</li>  
</ul>
```

Propiedad	Propósito
icons	Objeto con los iconos que se usan en los submenús, con la sintaxis de iconos de <i>jQuery UI</i>
menus	Cadena con el selector para indicar que elementos conforman el contenedor del menú, incluyendo los sub-menús. Por defecto es <code>ul</code>
position	Objeto que define la posición del menú respecto al elemento asociado
role	Personaliza el rol <i>ARIA</i> (accesibilidad) para el menú y sus elementos. El rol por defecto es <code>menu</code> que asigna <code>menuitems</code> a sus elementos. Si el rol toma el valor de <code>listbox</code> , los elementos serán <code>option</code> .
disabled	Booleano que permite desactivar el menú



Ejemplos Menús con Iconos

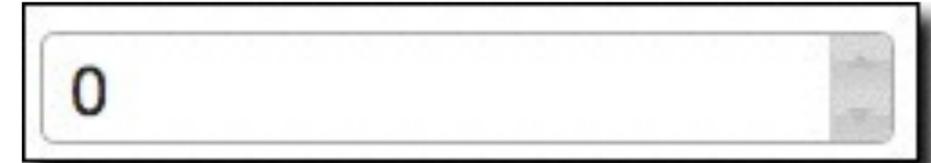
```
<ul>
  ...
  <li><a href="#"><span class="ui-icon ui-icon-folder-open"></span>Abrir</a></li>
  <li><a href="#"><span class="ui-icon ui-icon-disk"></span>Guardar</a></li>
  <li class="ui-state-disabled">Salir</li>
  ...
</ul>
```

```
$( "#menu" ).menu( {
  icons: { submenu: "ui-icon-circle-triangle-e" },
});
```





8.2.7 Spinner



- `spinner()`
- <http://api.jqueryui.com/spinner> - <http://jqueryui.com/spinner/>
- En HTML `input [type="number"]`
 - <http://caniuse.com/#feat=input-number>
- Control numérico con botones para incrementar/decrementar valor
- Tanto el valor de los incrementos como el paso de páginas se puede configurar.
- Soporte de teclado: Cursores verticales y *PageUp/PageDown*

```
<input id="spin" value="0" />
```

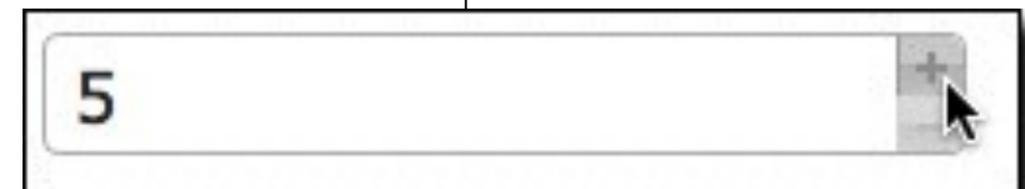
```
$( "#spin" ).spinner( );
```



Configuración Slider

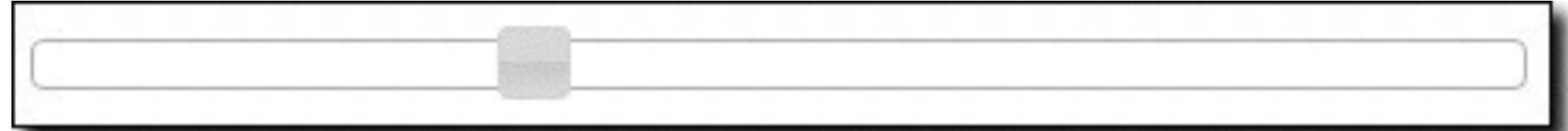
Propiedad	Propósito
min	Valor mínimo que puede alcanzar
máx	Valor máximo que puede alcanzar
step	Incremento/decremento (paso) que se aplicará al valor del campo, lo que permite que en vez de ir de uno en uno, los pasos sean de un valor personalizado
page	Número de pasos de paginación al invocar los métodos <code>pageUp</code> o <code>pageDown</code> .
incremental	Función que permite definir el valor con el que cambiará el control, lo que posibilita definir valores incrementales.
icons	Objeto con los iconos que se usarán en los botones de incrementar y decrementar valor, con la sintáxis de iconos de <i>jQuery UI</i>

```
$( "#spin" ).spinner( {  
  min: -5,  
  max: 10,  
  step: 5,  
  icons: {  
    up: "ui-icon-plus", down: "ui-icon-minus"  
  }  
} );
```





8.2.8 Slider



- `slider()`
- <http://api.jqueryui.com/slider> - <http://jqueryui.com/slider>
- En HTML `input [type="range"]`
 - Ampliamente soportado por todos los navegadores
- Control visual para elegir un valor dentro de un rango.
- A partir de una capa
- Soporte de teclado: Cursores verticales/horizontales y *PageUp/PageDown*

```
<div id="slide"></div>
```

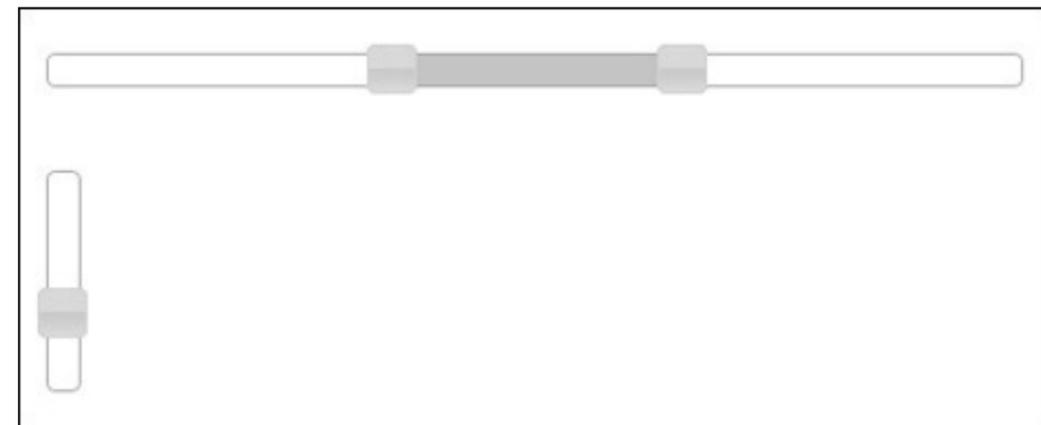
```
$( "#slide" ).slider();
```



Configuración Slider

Propiedad	Propósito
min	Valor mínimo que puede alcanzar (sin incluir)
máx	Valor máximo que puede alcanzar (sin incluir)
value	Valor con el que se inicializa el control (50 es la mitad)
values	Array con los valores de los diferentes elementos que podemos deslizar
range	Booleano que al usarlo con la propiedad <code>values</code> , define si realiza el rango existente entre el valor inferior y el superior
step	Incremento/decremento que se aplicará al valor del campo, lo que permite que en vez de ir de uno en uno, los saltos sean de un valor personalizado
orientation	Cadena que permite situar el control de manera <code>vertical</code> u <code>horizontal</code>

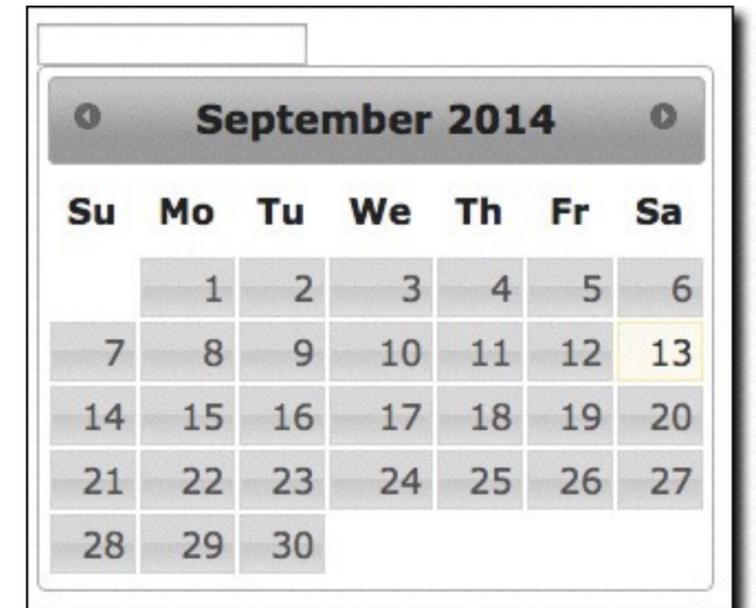
```
$( "#slide" ).slider({  
  values: [35, 65], range: true  
});  
$( "#slideV" ).slider({  
  orientation: "vertical", value: 35  
});
```





8.2.9 Selector de Fecha

- `datepicker()`
- <http://api.jqueryui.com/datepicker> - <http://jqueryui.com/datepicker>
- En HTML `input [type="date"]`
 - Soporte parcial (<http://caniuse.com/#feat=input-datetime>)
- Campo para introducir fechas con ayuda de un calendario y validación de formato.
- API muy completa
- Soporte completo de teclado



```
<input id="fecha" />
```

```
$( "#fecha" ).datepicker();
```



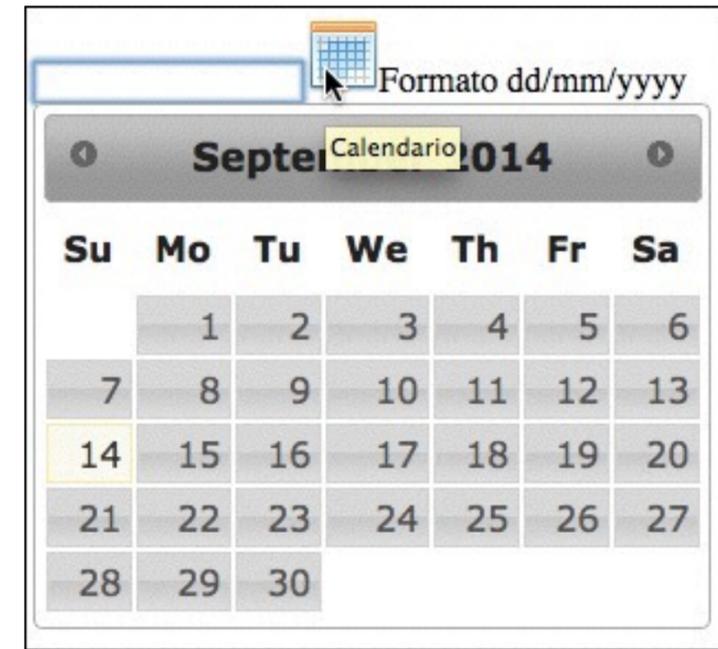
Configuración Básica - Selector de Fecha

Propiedad	Propósito
defaultDate	Fecha con el día que saldrá el calendario seleccionado al cargar el control. Los formatos posibles los estudiaremos en la siguiente sección.
altField	Indica un campo adicional que se actualizará con la fecha seleccionada. Se puede indicar mediante un selector, un objeto <i>jQuery</i> o un objeto DOM <i>HTMLInputElement</i>
appendText	Texto a mostrar después de cada campo de fecha, por ejemplo, para mostrar el formato esperado
showOn	Indica el evento que muestra el control. Por defecto es <code>focus</code> . Otros valores son <code>button</code> cuando se pulsa un botón o <code>both</code> , tanto para <code>focus</code> como <code>button</code> .
buttonImage	URL de la imagen que se usa para mostrar el control cuando <code>showOn</code> es <code>button</code> o <code>both</code> .
buttonText	Texto a mostrar como propiedad <code>alt</code> del botón que lanza el control.
buttonImageOnly	Booleano que indica si el imagen se debe renderizar por si misma (como imagen) o mediante un botón. Por defecto es <code>false</code> .
showOtherMonths	Booleano que indica si muestra los días del mes anterior y el posterior. Por defecto es <code>false</code> y además los días no son seleccionables. Para ello hay que usar la propiedad <code>selectOtherMonths</code>
selectOtherMonths	Booleano que indica si los días mostrados del mes anterior y posterior son seleccionables. Por defecto es <code>false</code> y sólo se aplica si <code>showOtherMonths</code> es <code>true</code> .



Ejemplos Configuración Básica - Selector de Fecha

```
$( "#fechaBoton" ).datepicker({  
  appendText: "Formato dd/mm/yyyy",  
  buttonImage: "cal.png",  buttonImageOnly: true,  
  buttonText: "Calendario",  
  showOn: "both"  
});
```



```
$( "#fechaBoton" ).datepicker({  
  buttonText: "Calendario",  
  showOtherMonths: true  
});
```





Formatos y Valores de Fecha

Valor/Formato	Propósito
<code>null</code>	Usa la fecha actual del sistema
<code>objeto Date</code>	Objeto <code>Date</code> con la fecha máxima que se puede seleccionar.
<code>+dias, -dias</code>	Utiliza <i>dias</i> como el número de días a partir de hoy. Por ejemplo, <code>+3</code> significa dentro de tres días, y <code>-2</code> hace 2 días.
<code>+1d +7w -1m +1y</code>	Utiliza una fecha relativa a la fecha actual, expresada como días (<code>d</code>), semanas (<code>w</code>), meses (<code>m</code>) y años (<code>y</code>), en el futuro (+) o en el pasado (-). Se pueden mezclar valores positivos y negativos en la misma fecha. Por ejemplo, <code>-1d +1m</code> sobre el 12 de Noviembre del 2014 quedaría como 11 de Diciembre del 2014.

Propiedad	Propósito
<code>dateFormat</code>	Cadena con el formato que se usa para parsear y formatear fechas. Por ejemplo, en España se suele usar <code>"dd/mm/yy"</code> . El formato por defecto es <code>"mm/dd/yy"</code> .
<code>altFormat</code>	Se utiliza con <code>altField</code> para configurar el formato del campo adicional
<code>autoSize</code>	Booleano que indica si el tamaño se ajusta al formato de <code>dateFormat</code> .
<code>isRTL</code>	Booleano que indica si el texto se lee de derecha a izquierda



Localización de Fechas

- <https://github.com/jquery/jquery-ui/tree/master/ui/i18n>
- Cada idioma va en su archivo correspondiente.
 - Por ejemplo, el castellano es `datepicker-es.js`
- Se incluye tras la librería de jQuery UI

```
<script src="jquery.js"></script>  
<script src="jquery-ui.js"></script>  
<script src="datepicker-es.js"></script>
```

- `$.datepicker.regional(locale)`

```
$( "#fecha" ).datepicker( $.datepicker.regional[ "es" ] );
```



Opciones de Selección de Fecha

Propiedad	Propósito
minDate / maxDate	Objeto Date con la fecha mínima/máxima que se puede seleccionar.
changeMonth	Booleano que indica si el mes aparece como desplegable o como texto. Por defecto, es <code>false</code>
changeYear	Booleano que indica si el año aparece como desplegable o como texto. Por defecto, es <code>false</code> . Para controlar los años disponibles hay que usar la propiedad <code>yearRange</code> .
yearRange	Cadena que especifica el número de años que se muestran cuando <code>changeYear</code> es <code>true</code> . Por defecto van a aparecer los 10 anteriores y posteriores al año actual, mediante la cadena <code>"c-10:c+10"</code> . Otros valores pueden ser relativos a hoy (<code>"-nn:+nn"</code>), relativos al año actual (<code>"c-nn:c+nn"</code>), absolutos (<code>"nnnn:nnnn"</code>) o combinando estos formatos
hideIfNoPrevNext	Booleano para ocultar los botones de anterior y siguiente. Normalmente se ocultan automáticamente si el rango de fechas lo permite. Por defecto es <code>false</code>
constraintInput	Booleano que restringe el campo de entrada al formato especificado por la opción <code>dateFormat</code> . Por defecto es <code>true</code> y espera una fecha con formato de números separados por barras inclinadas <code>99/99/99</code> . Si se usa con <code>minDate</code> y <code>maxDate</code> , validará que la fecha se encuentre entre dicho rango.
numberOfMonths	Número de meses a mostrar de una vez en una sola fila. También se puede usar un array que indique las filas y las columnas de meses a mostrar.
showCurrentAtPos	Número con la posición del mes actual. Se usa con <code>numberOfMonths</code> para mostrar varios meses
stepMonths	Número con la cantidad de meses a mover al pulsar sobre los enlaces de anterior y siguiente mes.

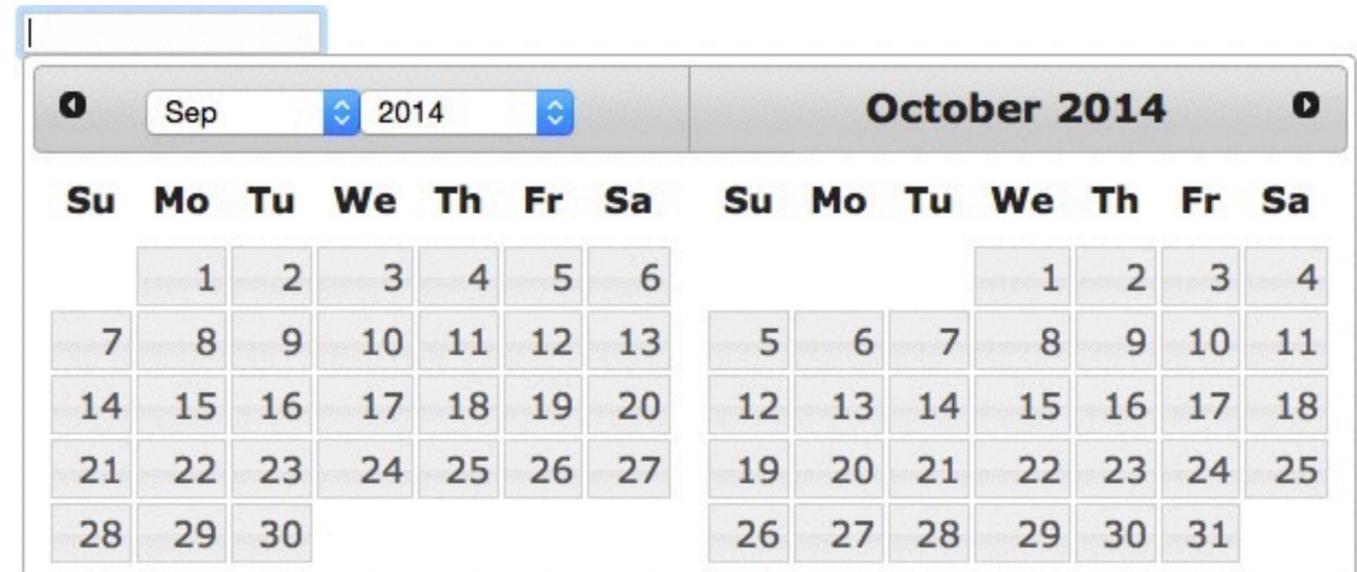


Propiedad	Propósito
showWeek	Booleano que indica si muestra el número de la semana. Por defecto es <code>false</code>
weekHeader	Cadena con el texto a mostrar como encabezado de la columna del número de la semana
firstDay	Número que indica el primer día de la semana: Domingo es 0, Lunes es 1, etc...
showMonthAfterYear	Booleano que indica si en la cabecera se muestra el mes tras el año
yearSuffix	Texto adicional a mostrar en la cabecera detrás del año.
showButtonPanel	Booleano que indica si muestran un barra de botón debajo del calendario, la cual contiene un botón <i>Today</i> que activa la fecha actual, y un botón <i>Done</i> que cierra el control. Los textos se pueden personalizar con las propiedades <code>currentText</code> y <code>closeText</code>
currentText / closeText	Texto del botón que activa/cierra la fecha actual, si <code>showButtonPanel</code> es <code>true</code>
prevText / nextText	Texto a mostrar para el enlace del anterior / siguiente mes. Si el control tiene un tema, el valor se sustituye por un icono
navigationAsDateFormat	Booleano que indica si las opciones <code>prevText</code> y <code>nextText</code> se parsean como fechas mediante la función <code>formatDate</code> , lo que permite mostrar, por ejemplo, el nombre de los meses.
gotoCurrent	Booleano que indica si el botón <i>Today</i> pasa a ser la fecha seleccionada en vez de la actual
dayNames / dayNamesShort / dayNamesMin	Array de cadenas que contiene un listado con el nombre largo / corto / muy corto de los días (todas / 3-4 letras / 1-2 letras)
monthNames / monthNamesShort	Array de cadenas que contiene un listado con el nombre largo / corto de los meses



Ejemplos de Configuración de Selección / Apariencia

```
$( "#fechaDesplegable" ).datepicker({  
  changeMonth: true,  
  changeYear: true,  
  numberOfMonths : 2  
});
```



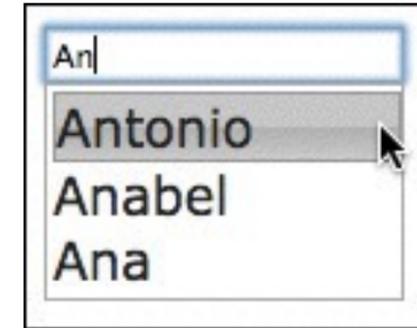
```
$( "#fechaEsp" ).datepicker({  
  showWeek: true,  
  showButtonPanel: true,  
  currentText: "Hoy",  
  dayNamesMin : [ "Lu", "Ma", "Mi", "Ju", "Vi", "Sa", "Do" ]  
});
```





8.2.10 Autocompletar

- `autocomplete()`
- <http://api.jqueryui.com/autocomplete>
- HTML5 → `[datalist]`
 - Soporte parcial (<http://caniuse.com/#feat=datalist>)
- Mecanismo flexible para completar el contenido de un campo de texto
 - Conforme el usuario comience a escribir, sugiere textos que contengan el texto escrito.
- Puede completar mediante contenido estático o vía AJAX.
- El contenido estático se configura mediante la propiedad `source`
- Si queremos etiquetas diferentes de los valores, tenemos que crear objetos que contengan la propiedad `label` y `value`



```
var datos = ["Aitor", "Antonio", "Anabel", "Ana",  
            "Marcos", "María", "Marta"];
```

```
$("#auto").autocomplete({  
  source: datos  
});
```

```
<input id="auto" />
```

```
var datosArray = [ {label:"Aitor (H)", value:"Aitor"},  
                  {label:"Antonio (H)", value:"Antonio"} ... ];
```



Configuración Autocompletar

Propiedad	Propósito
source	Campo obligatorio que indica la fuente de datos. Puede ser un array con los valores, una cadena con la url donde obtener un recurso JSON, o una función que obtiene los datos de una fuente externa.
delay	Milisegundos de retraso tras una pulsación de tecla para actualizar el control. Por defecto es 300
minLength	Número mínimo de caracteres que el usuario tiene que escribir antes de mostrar el menú de autocompletado. Por defecto es 1
autoFocus	Booleano para indicar si el primer elemento de la lista tomará el foco, con lo que el usuario puede seleccionarlo automáticamente pulsando <code>Enter</code> . Por defecto es <code>false</code>
appendTo	Selector que especifica el elemento al cual se adjuntará el desplegable. Por defecto es <code>body</code>
position	Posición del menú respecto respecto al elemento de entrada
disabled	Booleano que deshabilita el control. Esta característica no repercute en el elemento <code>input</code> . Por defecto es <code>false</code>



Ejemplo Autocompletar

```
$( "#cervezas" ).autocomplete({
  source: function( request, response ) {
    $.ajax({
      url: "http://api.openbeerdatabase.com/v1/beers.json?callback=?",
      dataType: "jsonp",
      data: {
        query: request.term
      },
      success: function( data ) {
        // Devuelve un array de cervezas
        var beers = data.beers.map(function( valor, indice, array ) {
          return valor.name; // Atributo con el nombre
        });
        response( beers );
      }
    });
  }
},
minLength: 2
});
```

<http://jsbin.com/yadavo/1/edit?html,js,output>



8.2.11 Barra de Progreso



- `progressbar()`
- <http://api.jqueryui.com/progressbar> - <http://jqueryui.com/progressbar/>
- HTML 5 → `<progress>`
 - Soporte Parcial (<http://caniuse.com/#feat=progressmeter>)
- Progreso de completitud de una tarea, mediante un indicador preciso de la cantidad de tarea que se ha realizado como un porcentaje.
- A partir de una capa (ocupa todo el ancho)
 - Limitar vía CSS

```
<div id="progreso"></div>
```

```
$( '#progreso' ).progressbar( {  
  value: 25  
} );
```



Configuración de Barra de Progreso

Propiedad	Propósito
disabled	Booleano que permite deshabilitar el control
max	Valor máximo. Por defecto es 100
value	Valor actual comprendido entre 0 y max.

```
<button id="decrementar">-</button>  
<button id="incrementar">+</button>
```



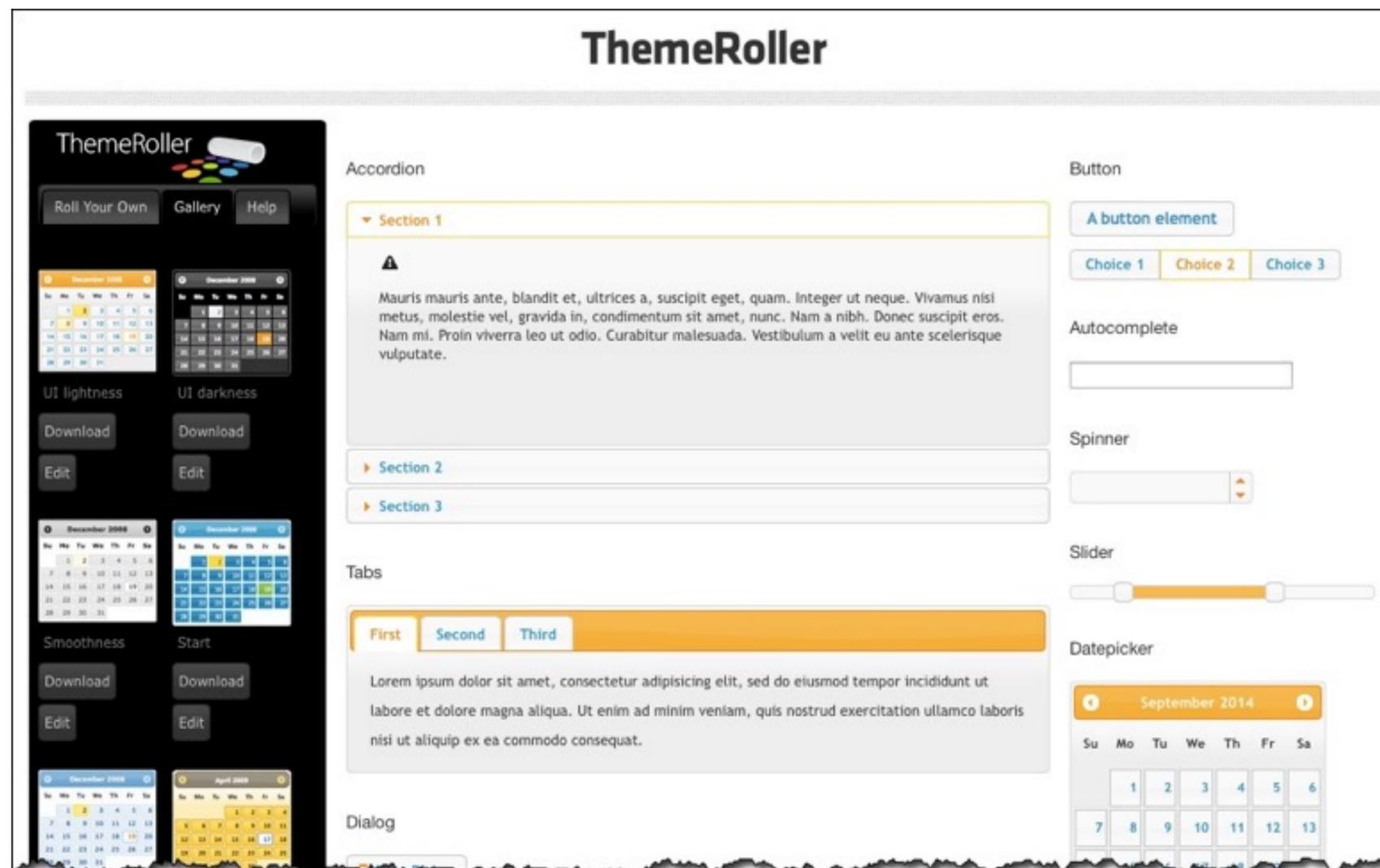
```
$( 'button' ).click( function (e) {  
    var divElem = $( '#progreso' );  
    var valor = divElem.progressbar( "value" );  
    divElem.progressbar( "value", this.id == "decrementar" ? valor - 10 : valor + 10 )  
});
```

<http://jsbin.com/zuguja/1/edit?html,js,output>



8.3 Temas

- jQuery UI ofrece un conjunto de temas predefinidos mediante una Galería
- Podemos personalizar nuestro tema desde **ThemeRoller**
 - <http://jqueryui.com/themeroller/>
- Se descarga
 - `jquery-ui.structure.css`
 - `jquery-ui.theme.css`
 - carpeta `images`





8.4 Efectos - Color

- JQuery UI amplia los efectos y animaciones de jQuery.
- Animaciones en base al **color**
- Propiedades `borderTopColor`, `backgroundColor`, y `color`

```
$(document).ready(function() {  
  $("#listado > li").hover(  
    function() {  
      $(this).animate({  
        backgroundColor: "#000", color: "#fff"  
      }, 500);  
    }, function() {  
      $(this).animate({  
        backgroundColor: "#fff", color: "#000"  
      }, 500);  
    });  
});
```

JavaScript

- jQuery
- jQuery UI
- Backbone
- Angular



Efectos en Clases

- jQuery UI añade un segundo parámetro con la duración de la animación en los métodos que ofrece jQuery para trabajar con las clases CSS
 - `addClass()`
 - `removeClass()`
 - `toggleClass()`
- La página se comporta como si hubiésemos realizado una llamada a `animate()`
 - especifica directamente todos los atributos de estilo que se modifican al aplicar la clase.

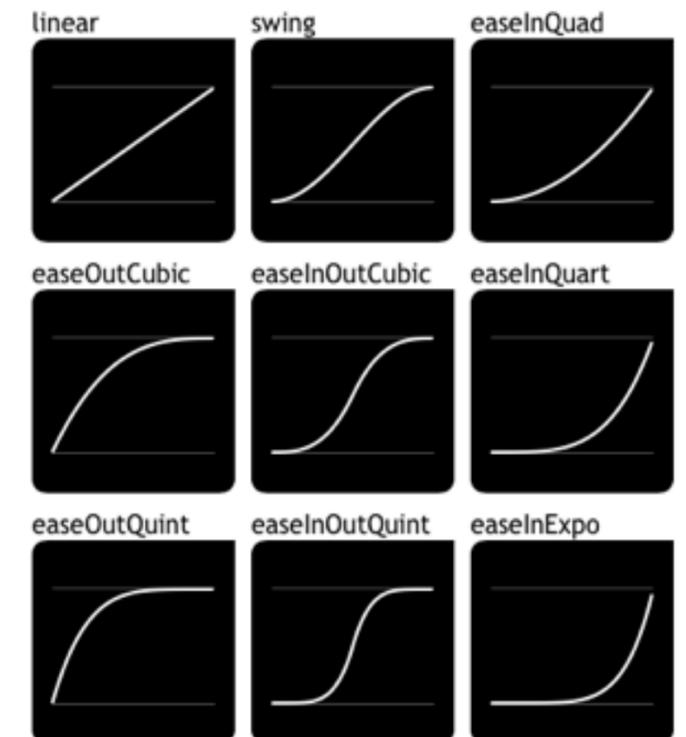
```
$(document).ready(function() {  
    $('h1').click(function() {  
        $(this).toggleClass('resaltado', 'slow');  
    });  
});
```



Easing

- Al aplicar un efecto, la animación no se realiza a un ritmo constante
 - Por ejemplo, en una animación de un segundo, tanto al principio como al final la aceleración es mayor que en el punto medio
- **Easing** (flexibilización) → Alteración del ritmo
- Se indica como un tercer parámetro (o vía objeto de configuración) en los métodos de animación de jQuery o jQuery UI.
- Posibles valores: <http://api.jqueryui.com/easings/>

```
$(document).ready(function() {  
  $('h1').click(function() {  
    $(this).toggleClass('resaltado', 'slow', 'easeInExpo');  
  });  
});
```





Efectos Adicionales

- `effect` (*nombreEfecto*)
- <http://api.jqueryui.com/effect/> - <http://jqueryui.com/effect/>
- Efectos con transiciones extra a las de enrollar, aparecer o mostrar.
 - `blind`, `bounce`, `clip`, `drop`, ...
- También podemos usar los métodos de *jQuery* pasándole como parámetro el efecto.

```
$( "#efectos" ).effect( "bounce" );  
$( "#efectos" ).hide( "bounce" );
```



8.5 Interacciones

- Añaden comportamiento a elementos/controles de la página.
- Se aplican a un objeto jQuery

Interacción	Método	URL
Arrastar	<code>.draggable()</code>	http://jqueryui.com/draggable/
Soltar	<code>.droppable()</code>	http://jqueryui.com/droppable/
Redimensionar	<code>.resizable()</code>	http://jqueryui.com/resizable/
Seleccionar	<code>.selectable()</code>	http://jqueryui.com/selectable/
Ordenar	<code>.sortable()</code>	http://jqueryui.com/sortable/



Ejemplo Arrastrar

```
<div id="capa">
  Capa
  <div id="1">uno</div>
  <div id="2">dos</div>
  <div id="3">tres</div>
</div>
```

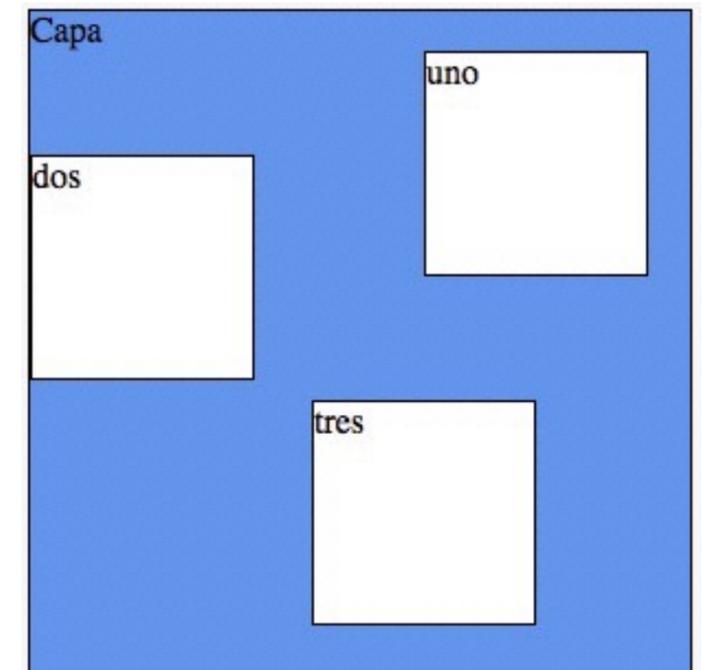
```
div {
  border: 1px solid black;
  height: 100px;
  width: 100px;
  background-color: white;
}

div#capa {
  width: 300px;
  height: 300px;
  background-color: cornflowerblue;
}
```

```
$( "#1" ).draggable( {
  axis: "x"
} );

$( "#2" ).draggable( {
  axis: "y"
} );

$( "#3" ).draggable( {
  containment: "#capa"
} );
```



<http://jsbin.com/joquh/2/edit?html,js,output>



¿Preguntas?