***Forms***

***Blur***: su funcionalidad es la detección de la perdida del foco en los diferentes elementos del DOM de una pagina Web. El evento se produce cuando faltala definición <input>

$(“imput”).blur(function() {

alter(“this imput fideñ has lost its focus.”);

});

***Change***: este método detecta el cambio de valor del elemento, es decir altera un texto cuando se cambia un <input> seleccionado en el DOM de nuestra Web.

$(“input”).change(function() {

alter(“the text has benn change.”);

});

***focus***: su funcionalidad es la de detectar el foco en los diferentes elementos del DOM. El evento se produce cuando el foco <input> campo obtenido del foco.

$(“input”).focus(function() {

$(“span”).css(“display” , “inline”).fodeOut(2000);

});

***focusin***: su funcionalidad es establecer el color de fondo de un elemento <div> cuando el elemento secundario de obtener el foco.

$(“div”).fucsin(function() {

$(this).css(“background-color” , “#FFFFCC”);

});

***focusOut***: su funcionalidad es establecer de un elemento <div> cuando el elemento <div> o cualquier elemento secundario pierde el foco.

$(“div”).fucusout(function() {

$(this).css(“background-color” , “#FFFFFF”);

});

***param***: su funcionalidad es crear una representación en serie de una matriz o un objeto, los valores serializados se pueden utilizar en la cadena de consulta URL al hacer una petición AJAX.

$(“button”).click(function() {

$(div).text($.**param**(personOBJ));

});

***select***: su funcionalidad es enlazar un controlador de eventos “seleccionador” evento JavaScritp o desencadenador ese evento es un elemento.

$(“#target”).**select**(function() {

alter(“Handler for .**select**() called.”);

});

***serialize***: su funcionalidad es crear una cadena de texto URL codificad por la serializacion de los valores del formulario, puede seleccionar uno o mas elementos del formulario o el propio formulario.

$(“button”).click(function() {

$(“div”).text($(“from”.serialize));

});

***serializeArray***: su funcionalidad es codificar un conjunto de elementos de formulario como una serie de un nombre y valores.

$(“from”).submit(function( event ) {

console.log($(this).***serializeArray***());

event.pereventDefult();

});

***submit***: su funcionalidad es mostrar una alerta cuando se esta enviando un formulario.

$(“form”).submit(function() {

alter(“submitted”);

});

***val***: su función es obtener el valor actual del primer elemnto en el conjunto de los elementos coincidentes.

$(“button”).click(function() {

$(“input:text”).val(“Glen Quagmire”);

});

***Efectos***.

***Hide***: su función es ocultar los elementos coincidentes.

$(“.target”).hide();

***show***: su función es mostrar los elementos ocultos seleccionados, no funciona en los elementos ocultos con los métodos de jQuery y diplay.

$(“button”).click(function() {

$(“p”).show();

});

***slide***: su función es deslizar el elemento fuera de la ventana gráfica.

$(“#clickme”).click(function() {

$(“#book”).slide(“slow”, function() {

});

});

***anímate***: su función es realizar una animación personalizada de un conjunto de propiedades de CSS.

$( "#clickme" ).click(function() {

$( "#book" ).animate({

opacity: 0.25,

left: "+=50",

height: "toggle"

}, 5000, function()

});

});

stop: su función es detener la animación en ejecución en los elementos coincidentes.

$( "#hoverme-stop-2" ).hover(function() {

$( this ).find( "img" ).stop( true, true ).fadeOut();

}, function() {

$( this ).find( "img" ).stop( true, true ).fadeIn();

});

***callbacks***: su función es una sentencia de JavaScript que ejecuta línea por línea. sin embargo, con efectos, la línea de código se puede ejecutar a pesar de que el efecto no se ha terminado. esto puede crear errores.

$(“jQuery”).callbacks(flags);

var acciones = $.callabacks(once);

***chaining***: su función es devolver un objeto jQuery, que luego se puede utilizar para llamar a otro método. esto permite hacer el encadenamiento de comandos donde se pueden realizar varios métodos en el mismo conjunto de elementos.

$(“#p1”).css(“color” , “red”)

.slideUp(2000)

.slideDown(2000);

***manejo de html y css***.

***text***: se utiliza para establecer la alineación horizontal de un texto.

h1 {

text-align: center;

}

h2 {

text-align: left;

}

***attr***: su función es obtener el valor de un atributo para el primer elemento en el conjunto de los elementos coincidentes. cuando se utiliza este método para devolver el valor de atributos, devuelve el valor del elemento emparejado primero.

$("button").click(function(){

$("img").attr("width","500");

});

***append***: su función es la inserción de contenido, especificado por el parámetro, al final de cada elemento en el conjunto de los elementos coincidentes.

$("p").append("Some appended text.");

***prepend***: su función es insertar el contenido, especificado por el parámetro, al principio de cada elemento en el conjunto de los elementos coincidentes.

$("button").click(function(){

$("p").prepend("<b>Prepended text</b>");

});

***after***: su función es instertar el contenido, especificado por el parámetro, después de cada elemento en el conjunto de los elementos coincidentes.

$("button").click(function(){

$("p").after("<p>Hello world!</p>");

});

***before***: su función es insertar el contenido, especificado por el parámetro, antes de cada elemento en el conjunto de elementos coincidientes.

$("button").click(function(){

$("p").before("<p>Hello world!</p>");

});

***remove***: su función es quitar los elementos HTML existentes, el método remove elimina elementos seleccionados y sus elementos secundarios.

$("#div1").remove();

***empty***: su función es eliminar todos los nodos hijos secundarios y el contenido de los elementos seleccionados. Este método no elimina el elemento en sí, o sus atributos.

$("button").click(function(){

$("div").empty();

});

***addclass***: su función es añadir uno o más nombres de clase a los elementos seleccionados, este método no elimina los atributos de clase existentes, solo añade uno o más nombres de clase para el atributo de clase.

$("button").click(function(){

$("p:first").addClass("intro");

});

***removeclass***: su función es eliminar uno o varias clases de los elementos seleccionados, si no se especifica ningún parámetro, este método eliminara todos los nombres de las clases de los elementos seleccionados.

$("button").click(function(){

$("p").removeClass("intro");

});

***toggleclass***: su función es añadir o quitar una o mas clases de cada elemento en el conjunto de elementos coincidentes, ya sea en función de la presencia de la clase o el valor del argumento del estado.

$("button").click(function(){

$("p").toggleClass("main");

});

***css***: su función es obtener las propiedades de estos estilos calculados para el primer elemento en el conjunto de los elementos coincidentes.

$(“div.exmaple”).css(“whidth” , function ( index ) ) {

retun index \* 50;

});

***width***: su función es obtener la anchura calculada actual para el primer elemento en el conjunto de los elementos seleccionados.

$("button").click(function(){

alert($("div").width());

});

***height***: su función es obtener la altura calculada actual para el primer elemento en el conjunto de los elementos seleccionados.

$("button").click(function(){

alert($("div").height());

});

***innerWidth***: su función es obtener la anchura interior computarizado actual para el primer elemento en el conjunto de elementos emparejados, incluyendo el relleno, pero no frontera.

$(selector).innerWidth()

***outerWidth***: su función es obterenr la altura calculada actual para el primer element en el conjunto de elementos emparejados, incluyendo el relleno.

$(selector).innerHeight()

***outerHegth***: su función es obtener la altura exterior computarizado actual (incluid el relleno, borde y margen) para el primer elemento en el conjunto de los elementos seleccionados.

$(selector).outerHeight(includeMargin)

***Modificar***

***Parent***: u función es obtener el padre de cada elemento en el conjunto actual de los elementos seleccionados opcionalmente filtradas por un selector.

$(document).ready(function(){

$("span").parent().css({"color": "red", "border": "2px solid red"});

});

***patentsUntil***: su función es obtener los antepasados de cada elemento en el conjunto actual de los elementos seleccionados, y hasta, pero sin incluir el elemento emparejado por el selector, nodo DOMS, o un objeto jQuery.

$(document).ready(function(){

$("span").parentsUntil("div").css({"color": "red", "border": "2px solid red"});

});

***Children***: su función es obtener los hijos d cada elemento en el conjunto de los elementos seleccionados, opcionalmente filtradas por un selector.

$(document).ready(function(){

$("ul").children().css({"color": "red", "border": "2px solid red"});

});

***Find***: su función es obtener los descendientes de cada elemento en el conjunto actual de los elementos coincidentes, filtrada por un selector, objetos jQuery o un elemento.

$(document).ready(function(){

$("ul").find("span").css({"color": "red", "border": "2px solid red"});

});

***Siblings***: su función es obtener los hermanos de cada elemento en el conjunto de elementos seleccionados opcionalmente filtradas por un selector.

$(document).ready(function(){

$("li.start").siblings().css({"color": "red", "border": "2px solid red"});

});

***Next***: su función es obtener el hermano inmediatamente después de cada elemento en el conjunto de elementos seleccionados. Si se proporciona un selector.

$(document).ready(function(){

$("li.start").next().css({"color": "red", "border": "2px solid red"});

});

***Nextall***: su función es obtener todos lo siguientes hermanos de cada elementos en el conjunto de elementos coincidentes, opcionalmente filtradas por un selector.

$(document).ready(function(){

$("li.start").nextAll().css({"color": "red", "border": "2px solid red"});

});

***netUntil***: su función es obtener todos los siguientes hermanos de cada elemento hasta, pero sin incluir el elemento emparejado por el selector, nodo DOM o un objeto jQuery pasado.

$(document).ready(function(){

$("li.start").nextUntil("li.stop").css({"color": "red", "border": "2px solid red"});

});

***prev***: su función es obtener el hermano inmediatamente anterior de cada elemento en el conjunto de elementos seleccionados.

$(document).ready(function(){

$("li.start").prev().css({"color": "red", "border": "2px solid red"});

});

***prevAll***: su función es obtener todos los hermanos anteriores de cada elemento en el conjunto de elementos seleccionados, opcionalmente faltados por un selector.

$(document).ready(function(){

$("li.start").prevAll().css({"color": "red", "border": "2px solid red"});

});

***prevUntil***: su función es obtener todos los hermanos anteriores de cada elemento hasta, pero sin incluir el elemento emparejado por el selector, nodo DOM o un elemento jQuery.

$(document).ready(function(){

$("li.start").prevUntil("li.stop").css({"color": "red", "border": "2px solid red"});

});

***First***: su función es reducir el conjunto de elementos emparejados a la primera en el conjunto.

$("div p").first()

***Last***: su función es reducir el conjunto de elementos que coinciden con el final en el conjunto.

$("p").last()

***Eq***: su función es reducir el conjunto de elementos que coincidan con el que está en el índice especificado.

$("p").eq(1).css("background-color", "yellow");

***Filter***: su función es reducir el conjunto de elementos emparejados a los que coinciden con el selector o pasan la prueba de la función.

$("p").filter(".intro")

***Not***: su función es eliminaran los elementos del conjunto de elementos seleccionados.

$("p").not(".intro")

***Ajax***: su función es realizar una solicitud HTTP asíncrona (AJAX).

$("button").click(function(){

$.ajax({url: "demo\_test.txt", success: function(result){

$("#div1").html(result);

}});

});

***Load***: su función es cargar los datos desde el servidor y colocar el HTML devuelto en el elemento seleccionado.

$("button").click(function(){

$("#div1").load("demo\_test.txt");

});

***Get***: su función es recuperar uno de los elementos que coinciden con el objeto jQuery.

$("button").click(function(){

$.get("demo\_test.asp", function(data, status){

alert("Data: " + data + "\nStatus: " + status);

});

});

***Post***: su función es cargar los datos desde el servidor mediante una petición HTTP POST.

$("button").click(function(){

$.post("demo\_test.asp", function(data, status){

alert("Data: " + data + "\nStatus: " + status);

});

});

***Json sintaxis***: los datos JSON se escribe como pares de nombre / valor. Un par nombre / valor consiste en un nombre de campo (entre comillas), seguido de dos puntos y seguido por un valor.

"firstName":"John"

***Json HTTP***: un uso común de JSON es para leer datos desde un servidor web, y mostrar los datos en una página web.

function myFunction(arr) {

var out = "";

var i;

for(i = 0; i < arr.length; i++) {

out += '<a href="' + arr[i].url + '">' + arr[i].display + '</a><br>';

}

document.getElementById("id01").innerHTML = out;

}

***Json files***: un uso común es para leer los datos desde un servidor web y mostrar los datos en una página web.

function myFunction(arr) {

var out = "";

var i;

for(i = 0; i < arr.length; i++) {

out += '<a href="' + arr[i].url + '">' + arr[i].display + '</a><br>';

}

document.getElementById("id01").innerHTML = out;

}

***Json SQL***: su uso común es para leer datos JSON desde un servidor web con PHP y MySQL.

***JQuery UI Interacciones P1***

***Draggable***: su función es permitir que los elementos puedan moverse con el ratón. Habilitar la función que puede arrastrarse en cualquier elemento DOM, mover el objeto arrastrable haciendo clic sobre ella con el ratón y arrastrándolo en cualquier lugar dentro de la ventana.

$(function() {

$("#draggable").draggable({

revert : function(event, ui)

});

***Droppable***: su función es crear objetos para los elementos que puedan arrastrarse, permite a cualquier elemento DOM ser lanzables, un objetivo para los elementos que pueden arrastrarse.

$("#dock").droppable({

drop: function(event, ui) {

}

});

***Resizable***: su función es cambiar el tamaño de un elemento con el ratón, permitir a cualquier elemento DOM ser de tamaño variable, con el cursor agarrar el borde derecho inferior y arrastre a la anchura o la altura que desea.

$(selector, context).resizable({option1: value1, option2: value2..... });

***Selectable***: su función es usar el ratón para seleccionar los elementos de forma individual o en grupo, habilitar un elemento DOM para ser seleccionable.

$(function() {

$( "#selectable-2" ).selectable({

delay : 1000

});

***Sortable***: su función es reordenar los elementos de una lista o de cuadricula utilizando el ratón, permitiendo a un grupo de elementos DOM para ser clasificable.

$(function () {

$("ol.example").sortable();

});

***jQuery Widgets***: su función es convertir un par de colectores y paneles de contenido en un acordeón, el margen de beneficio de su contenedor necesita pares de cabecera y los paneles de contenido.

$( ".selector" ).accordion({

active: 2

});

***Autocomplete***: su función es permitir al usuario seleccionar de una lista pre-poblada de los valores a medida que escribe, el aprovechamiento de búsqueda y filtrado de forma rápida.

$( ".selector" ).autocomplete({

appendTo: "#someElem"

});

***Button***: su función es realizar los elementos por defecto d forma como los botones, las entradas y los anclajes a los botones temable con vuelo estacionario apropiado y estilos activos.

$( ".selector" ).button({

classes: {

"ui-button": "highlight"

}

});

***Checkboxradio***: su función es convertir entrada de tipo radio a la casilla de verificación a los botones temable.

$( ".selector" ).checkboxradio({

classes: {

"ui-checkboxradio": "highlight"

}

});

***Controlgroup***: su función es proporcionar una agrupación visual de botón y otros widgets de entrada.

$( ".selector" ).controlgroup({

classes: {

"ui-controlgroup": "highlight"

}

});

***Datepicker***: su función es seleccionar una fecha de un calendario emergente o en línea.

$( "#datepicker" ).datepicker({

beforeShowDay: $.datepicker.noWeekends

});

***Dialog***: su función es obtener el contenido abierto en una superposición interactiva.

$( ".selector" ).dialog({

appendTo: "#someElem"

});

***JQuery UI Interacciones P1***

***Menú***: su función es configurar con el ratón y el teclado interacciones para la navegación.

$( ".selector" ).menu({

classes: {

"ui-menu": "highlight"

}

});

***Progressbar***: su función es mostrar el estado de un determinado proceso o indeterminado.

$( ".selector" ).progressbar({

classes: {

"ui-progressbar": "highlight"

}

});

***Selectmenu***: su función es duplicados y se extiende la funcionalidad de un elemento select de HTML nativa para superar las limitaciones del control nativo.

$( ".selector" ).selectmenu({

appendTo: "#someElem"

});

***Slider***: su función es arrastrar un controlador para seleccionar un valor numérico.

$( ".selector" ).slider({

animate: "fast"

});

***Spinner***: su función es mejora de una entrada de texto para introducir valores numéricos, con los botones ARRIBA / ABAJO y flecha entrega de llaves.

$( ".selector" ).spinner({

classes: {

"ui-spinner": "highlight"

}

});

***Tabs***: su función es obtener el área de contenido individual con múltiples paneles, cada uno asociado con un encabezamiento en una lista.

$( ".selector" ).tabs({

classes: {

"ui-tabs": "highlight"

}

});

***Tooltip***: su función es mostrar información sobre herramientas temable personalizables, que sustituye a la información de herramientas nativas.

$( ".selector" ).tooltip({

classes: {

"ui-tooltip": "highlight"

}

});

***JQuery UI Temas y Utilidades***

***Position***: su función es obtener las coordenadas actuales del primer elemento en el conjunto de elementos coincidentes, en relación con el padre de desplazamiento.

$("button").click(function(){

var x = $("p").position();

alert("Top: " + x.top + " Left: " + x.left);

});

**Widget Factory**:su función es crear plugins jQuery con estado utilizando la misma abstracción como todos los widgets jQuery UI.

$( ".selector" ).progressbar({

classes: {

"ui-progressbar": "highlight"

}

});