**JQuery**

JQuery es una librería de JS que nos deja manipular de forma más fácil cierta colección de funciones, tales como: Ajax, CSS, efectos y animaciones, eventos, formularios, manipulación del DOM, etc.

Su selector universal es el **$** o **JQuery**.

\*Para evitar errores con Jquey en cuanto a si no reconoce eventos u otros códigos, se puede utilizar un **if**, dándole como condición que exista un elemento que siempre estará. Por ejemplo:

 if (document.getElementById(calcular)) {

// Codigo

}

Uso Estricto

El uso estricto nos asegura que el documento de haya cargado antes que nuestro JS, así no suceden errores:

**$(document).ready(function() {**

**});**

O también:

**$(function() {**

**‘use strict’;**

**});**

Seleccionar contenido del DOM

Se pueden seleccionar elementos del DOM utilizando el mismo lenguaje que CSS:

**$(‘**Elemento**’); :** Con **.clase** elegimos una clase, con **#id** elegimos una id, sin nada elegimos las **etiquetas.** Como en JS si colocamos otro elemento luego del primero este lo buscara dentro del padre. O con una coma **,** puedes seleccionar dos elementos diferentes entre sí.

Si colocamos luego del elemento **:first** o: **:last**, se nos elige el primero o el ultimo respectivamente.

Desde ahora **nomN =** tag, variable, id, clase, etc.

Funciones pre escritas

**Modificacion Dom**

**$(“nomN”).remove();:** Remueve una parte del DOM.

**$(“ nomN”).hide();:** Coloca un display none al elemento.

**$(“ nomN”).show();:** Muestra al elemento.

**$(“ nomN”).removeClass(“**clase**”);:** Remueve cierto tipo de clases.

**$(“ nomN”).clone();:** Clona una parte del DOM.

**$(“ nomN”).text();:** Nos regresa el texto de un elemento.

**$(“ nomN”).text(“**texto nuevo**”);:** Cambia el texto de un elemento.

**$(“ nomN”).html (“**tag nuevo con texto**”);:** Cambia el texto del elemento que tenga tal tag. Ejemplo **<h1> </h1>**

**$(“ nomN”).append(varN);:** Coloca en el elemento seleccionado el hijo que utilices en la **varN.** Lo coloca al final del padre.

**$(“ nomN”).prepend(varN);:** Coloca en el elemento seleccionado el hijo que utilices en la **varN.** Lo coloca al inicio del padre.

**$(“ nomN”).children();:** Nos regresa los hijos del elemento. Se puede agregar más **children()** al propio **children().**Puede ser usado en conjunto de **parent().** Puede utilizar posiciones con **[**número de posición**]**

**$(“ nomN”).parent();:** Nos regresa los padres del elemento. Se puede agregar más **parent()** al propio **parent().** Puede ser usado en conjunto de **children().** Puede utilizar posiciones con **[**número de posición**]**

**$(“ nomN”).next ();:** Nos regresa el hermano del elemento. Se puede agregar más **next()** al propio **next().**Puede ser usado en conjunto de **parent()** y **children().**

**$(“ nomN”).attr(“**atributo**”);:** Nos devuelve la información del atributo. También puede agregar atributos

**$(“ nomN”).attr(“**atributo**”, “**cambio del atributo**”);:** Cambia la información del atributo. También puede agregar atributos

**return false;** Evita el brinco o salto cuando un enlace nos lleva a una parte del HTML.

**Modificacion de CSS del DOM**

**$(“ nomN”).addClass(“**clase**”);:** Agrega clases al elemento.

**$(“ nomN”).removeClass(“**clase**”);:** Quita clases al elemento.

**$(“ nomN”).css(“**propiedad**”, “**especificación de propiedad**”);:** Cambia los atributos de clase del elemento

**$(“ nomN”).slideUp(**duración en milisegundos**);:** Crea una pequeña animación de subida, si es el primer hijo desaparece frente al segundo.

**$(“ nomN”).slideDown(**duración en milisegundos**);:** Crea una pequeña animación de bajada, si es el último hijo desaparece frente ante-ultimo.

**$(“ nomN”).slideToggle(**duración en milisegundos**);:** Es un conjunto de **slideUp** y **sildeDown** que cambia según cual este activo en tal momento.

**$(“ nomN”).FadeIn(**duración mls**);:** Lo muestra pero de forma animada.

**$(“ nomN”).FadeOut(**duración mls**);:** Lo esconde pero de forma animada

**$(“ nomN”).on(“click”, function() {**

**$(“ nomN”).animate(“**propiedad**”, “**especificación de propiedad**”,** duración de la animación, en milisegundos**);**

**});:** Anima el cambio de css cuando suceda un evento

**Modificacion del CSS en objetos**

Con esto podemos modificar de varias maneras el CSS del elemento. Por ejemplo:

**$(“ nomN”).css({**

**“width”: “200px”,**

**“color”: “blue”**

**})**

El **anímate()** también puede ser utilizado en objetos. Ejemplo:

**$(“ nomN”).on(“click”, function() {**

**$(“ nomN2”).animate({**

**“width”: “200px”**

**},** **1000);**

**});**

Eventos

Para realizar un código cuando suceda un evento se utiliza:

**$(“nomN”).on(“**evento**”; function() {**

código a suceder

**});**

A los eventos se los puede luego modificar también, ejemplo:

**$(“nomN”).on(“**evento**”; function(e) {**

**e.preventDefault();**

**});**

La **e** proviene del código **event**. En este caso estamos haciendo que la acción default que sucede en Chrome no se active en nuestro evento.

**Eventos específicos**

**“click”:** Sucede cuando damos click en el elemento.

**“mouseenter”:** Cuando el mouse esta sobre el elemento (funciona como el hover de CSS)

**“mouseleave”:** Cuando el mouse sale del elemento (contrario del **mouseenter**)

**Eventos en objetos**

Se pueden juntar eventos en un objeto asi los tenemos solo en una pieza de código. Ejemplo:

**$(“ nomN”).on({**

**click: function() {**

código a realizar

**},**

**mouseenter: function() {**

código a realizar

**},**

**mouseleave: function() {**

código a realizar

**}**

**})**

This

El this se utiliza dentro de funciones de evento (pueden ser otras, pero en raro caso). Este sirve para referirse al elemento que utilizamos en primer momento para realizar un cambio cuando se activa con él. Ejemplo:

**$(“nomN”).on(“click”; function() {**

**$(this).hide**

**})**

**nomN** se esconderá cuando demos click en él.

Ajax

Con Ajax podemos cargar diferente contenido a nuestra página. Se suele utilizar en eventos. Se llama con **$.ajax** para poder cargar el programa y se define el lugar que se llama. Luego se definen sus características en un objeto, las cuales son:

**success: funN** Función que realizara si el contenido se carga con éxito.

**type: “GET”** Como toma el contenido, en este caso GET.

**datatype: ”**tipo de data**”** El tipo de data, puede ser text, htmtl, java, etc.

Esta función que denominamos en el **success** debe realizar el comando para colocar el contenido. Esta función también tiene sus características como el **data** (tipo de data)**, status** (si se realizó con éxito la operacion)**, jqxhr** (código propio de Ajax para llamarlo.)

Dentro de esta se debe llamar al elemento que queremos cambiar y luego especificar qué tipo de dato estamos ingresando. Por último se utiliza un **console.log(status)** para saber el status de Ajax

Quedaría entonces, en un ejemplo que agregamos texto a un aside:

**$(“ nomN”).on(“click”, cargarAjax);**

**function cargarAjax() {**

**$.ajax(“promociones.text”, {**

**success: cargarContenido,**

**type: “GET”,**

**datatype: “text”**

**});**

**};**

**function cargarContenido(data, status, jqxhr) {**

**$(“aside”).text(data);**

**console.log(status)**

**}**

Existe un comando para cargar archivos html de forma mucho mas simple:

**(“nomN”).load(“**archivo html**”);**

Recorrer arreglos

Jquery nos permite recorrer arreglos de forma mucho mas simple. Se utiliza el **$.each** de la siguiente forma. Este también sirve con objetos.

**arglN =** arreglo

**$.each(arglN, function(i, v) {**

código a utilizar.

**})**

La **i** nos deja ver los números del arreglo, o la cantidad de elementos dentro del arreglo. En un objeto nos dará su llave.

La **v** nos da el valor de cada elemento, en su mayoría un texto o numero. En un objeto nos dará el contenido de su llave

Utilizar Plugins

Primero se debe conseguir el plugin desde alguna pagina de internet, se descarga y agrega a la carpeta js de nuestro archivo. Luego debemos conectar el index con el script del archivo js.

Luego debemos seleccionar el elemento donde se posiciona la animación o plugin, y ahí colocarle la **function.** Ejemplo:

**$(“#id”).animateNumber({ number: 10 })**

Crear barra Fija en nuestra pagina

//Menu-fijo

    var windowHeight = $(window).height();

    var barraAltura = $(".barra").innerHeight();

    $(window).scroll(function () {

        var scroll = $(window).scrollTop();

        if (scroll > windowHeight) {

            $(".barra").addClass("fixed");

            $("body").css({ "margin-top": barraAltura + "px" });

        } else {

            $(".barra").removeClass("fixed");

            $("body").css({ "margin-top": "0px" });

        }

    });

.fixed {

    position: fixed;

    width: 100%;

    top: 0;

    left: 0;

    z-index: 2;

}

Crear Menú activo

//Programa-de-Conferencias

    $(".programa-evento .info-curso:first").show();

    $(".menu-programa a:first").addClass("activo");

    $(".menu-programa a").on("click", function () {

        $(".ocultar").hide();

        $(".menu-programa a").removeClass("activo");

        $(this).addClass("activo");

        var enlace = $(this).attr("href");

        $(enlace).fadeIn(1000);

        return false;

    });