

## Documentación.

### DATEPICKER.

Hemos usado el widget datepicker con 2 elementos que son “desde” y “hasta”

El código es el siguiente.

```
$(function ($) {
    $.datepicker.regional['es'] = {
        closeText: "Cerrar",
        prevText: "<#x3C;Ant",
        nextText: "Sig>#x3E;",
        currentText: "Hoy",
        monthNames: ["enero", "febrero", "marzo", "abril", "mayo", "junio",
            "julio", "agosto", "septiembre", "octubre", "noviembre", "diciembre"],
        monthNamesShort: ["ene", "feb", "mar", "abr", "may", "jun",
            "jul", "ago", "sep", "oct", "nov", "dic"],
        dayNames: ["domingo", "lunes", "martes", "miércoles", "jueves", "viernes", "sábado"],
        dayNamesShort: ["dom", "lun", "mar", "mié", "jue", "vie", "sáb"],
        dayNamesMin: ["D", "L", "M", "X", "J", "V", "S"],
        weekHeader: "Sm",
        dateFormat: "dd/mm/yy",
        firstDay: 1,
        isRTL: false,
        showMonthAfterYear: false,
        yearSuffix: ""
    };
    $.datepicker.setDefaults($.datepicker.regional['es'])
});

$(function () {
    var dateFormat = "mm/dd/yy",
        from = $("#from")
        .datepicker({
            defaultDate: "+1w",
            changeMonth: true,
        })
        .on("change", function () {
            to.datepicker("option", "minDate", getDate(this));
        }),
        to = $("#to").datepicker({
            defaultDate: "+1w",
            changeMonth: true,
        })
        .on("change", function () {
            from.datepicker("option", "maxDate", getDate(this));
        });

    function getDate(element) {
        var date;
        try {
            date = $.datepicker.parseDate(dateFormat, element.value);
        } catch (error) {
            date = null;
        }

        return date;
    }
});
```

Al principio establecemos los parámetros de la región española con los parámetros del siguiente enlace: <https://cursoprogramador.wordpress.com/2015/08/19/poner-el-datepicker-de-jqueryui-en-castellano/>

Tras eso he utilizado el widget de 2 datepicker por eso cogemos el elemento con el id “from” y lo hacemos un datepicker, con la opción de cambiar de mes.

Lo mismo realizamos con el elemento de id “to”.

### AUTOCOMPLETE.

Hemos usado el widget autocomplete y como requería obtener datos de una api acorde con la temática del proyecto de un videoclub he usado la api **OMDB** que es de películas.

El código es el siguiente.

```

$(function () {

    function log(message) {
        $('#<div>').text(message).prependTo("#log");
        $('#log').scrollTop(0);
    }

    $('#películas').autocomplete({
        source: function (request, response) {
            $.ajax({
                url: "http://www.omdbapi.com/?s=" + $('#películas').val() + "&apikey=b8187916&page=1",
                dataType: "json",
                data: {
                    term: request.term
                },
                success: function (data) {
                    var pel = cogerPelículas(data);
                    console.log(pel);
                    response(pel);
                }
            });
        },
        minLength: 2
    });

    $(function () {
        $(document).tooltip();
    });

    function cogerPelículas(data) {
        var array_películas = [];

        for (let i = 0; i <= 9; i++) {
            console.log(i);
            array_películas.push(data.Search[i].Title)
        }
        return array_películas
    }

})

```

El elemento con el id “película” es el autocomplete, lo que realizamos es una petición ajax a la url de **omdb** con el contenido del input de id “película”. Cuando se realiza llama a la función **cogerPelículas** y se le pasa el resultado de la petición ajax, esta función recoge de los resultados de la petición el nombre y se la pasa para el autocomplete.

## ANIMACIONES.

Las animaciones realizadas en el proyecto son en forma de Toggles.

Tenemos 3 toggles, “Sobre nosotros”, “Dirección”, “Correo”.

El código es el siguiente:

```

$(function () {
    function runEffect() {
        $("#effect1").toggle("fold", 500);
    }

    $("#button1").on("click", function () {
        runEffect();
    });
});

$(function () {
    function runEffect() {
        $("#effect2").toggle("clip", 500);
    }

    $("#button2").on("click", function () {
        runEffect();
    });
});

$(function () {
    function runEffect() {
        $("#effect3").toggle("highlight", 500);
    }

    $("#button3").on("click", function () {
        runEffect();
    });
});

```

Tenemos botones con ids distintas como se muestra que son los que activan los divs con ids effects.

Cada uno tiene un efecto distinto.

- 1º tiene el efecto “fold”
- 2º tiene el efecto “clip”
- 3º tiene el efecto “highlight”

Todos los efectos duran 0’5 segundos.

## OTROS ELEMENTOS.

He usado otros elementos de JQueryUI como un Tab, con el siguiente código.

```

$(function () {
    $("#tabs").tabs();
});

```

```

<div id="tabs">
  <ul>
    <li><a href="#tabs-1">Alquilar</a></li>
    <li><a href="#tabs-2">Películas</a></li>
    <li><a href="#tabs-3">Sobre Nosotros</a></li>
  </ul>

```

Esto ordena en 3 partes los apartados de Alquiler, Películas y Sobre Nosotros.

Dentro de Alquiler hay un “Acordeón” con el siguiente código.

```

<div id="tabs-1">
  <div id="accordion">
    <h3>Datos personales</h3>
    <div>
      <form class="formulario">
        <div class="form-row">
          <div class="form-group col-md-6">
            <label for="inputNombre">Nombre</label>
            <input type="text" class="form-control" id="inputNombre" placeholder="Nombre"
              title="Introduce aquí tu nombre completo">
          </div>
          <div class="form-group col-md-6">
            <label for="inputApellido">Apellidos</label>
            <input type="text" class="form-control" id="inputApellido" placeholder="Apellidos"
              title="Donde pone apellidos va apellidos por supuesto">
          </div>
        </div>
      </form>
    </div>
    <h3>Cuenta</h3>
    <div>
      <div class="form-row">
        <div class="form-group col-md-6">
          <label for="inputEmail">Correo</label>
          <input type="email" class="form-control" id="inputEmail" placeholder="email"
            title="Aquí el e-mail de tu cuenta">
        </div>
        <div class="form-group col-md-6">
          <label for="inputPassword">Password</label>
          <input type="text" class="form-control" id="inputPassword" placeholder="Password"
            title="Aquí la contraseña de tu cuenta">
        </div>
      </div>
    </div>
    <h3>Fecha de Alquiler</h3>
    <div>
      <label for="from">Desde</label>
      <input type="text" id="from" name="from" title="Fecha de adquisición de la película">
      <label for="to">Hasta</label>
      <input type="text" id="to" name="to" title="Fecha de devolución, se cobrará por el retraso">
    </div>
    <h3>Titulo a Alquilar</h3>
    <div>
      <div class="ui-widget">
        <label for="películas">Película: </label>
        <input id="películas" title="Película que quieres alquilar">
      </div>
    </div>

```

```

$(function () {
  $('#accordion').accordion();
});

```

Convierte en un acordeón con las pestañas Datos personales, Cuenta, Fecha de Alquiler, Titulo a Alquilar.

También como extra he añadido un buscador de títulos en el videoclub con el siguiente código.

```

function buscarpelículas() {
  limpiarPelículas();
  $.ajax({
    url: "http://www.omdbapi.com/?s=" + $('#busqueda').val() + "&apikey=b8187916&page=1",
    success: function (res) {
      enmaquetarPelículas(res)
    }
  })
}

function enmaquetarPelículas(datos) {
  for (let i = 0; i <= 9; i++) {
    $('#resbusqueda').append('<div class="película"><p>' + datos.Search[i].Title + '</p>')
  }
}

function limpiarPelículas() {
  $('#resbusqueda').empty()
}

```

La función `BuscarPelículas` ocurre cuando en el elemento del tab película introducimos un título y pulsamos el botón, lo primero que hace es llamar a la función `limpiarPelículas` que vacía el contenedor con el id “resbusqueda” que es un div que contiene el resultado de la búsqueda.

Realiza una petición ajax a la misma api que para el autocomplete, con el valor del input con id “busqueda”

Cuando ocurre llama a la función `enmaquetarPelículas` y se le pasa como parámetro los datos obtenido de la petición ajax.

Esta función recorre los resultados y añade un div con la imagen y el título de la película.