

TEMA 4: Integración de contenido interactivo

Objetivos de esta tarea:

El desarrollo de esta tarea consta de dos partes con dos objetivos diferenciados

- A. Maquetar con la propiedad `display: flex;`
- B. Aplicar contenido interactivo.

Consideraciones previas:

Tomaréis como base la actividad **corregida AQ4.1_M3UV** de la quincena 4 del curso (versión "columnas").

Recordad que debíais tener:

- a) las siguientes secciones: Cabecera, menú, cuerpo (contenidos principales), aside (contenidos extra) y pie de página.
- b) la página **html**: columnas.html
- c) los archivos **css**: estilo.css y columnas.css.

Antes que nada debéis revisar el correcto diseño y funcionalidad de dicha actividad en función de las correcciones que se te propusieron.

Se trata de dejar lo más pulido posible el diseño y funcionalidad de dicha tarea antes de comenzar con esta nueva.

Este ejercicio tiene **dos** partes (leed el documento hasta el final!!):

- A. Maquetar con cajas flexibles
- B. Aplicar contenido interactivo con jQuery.

Parte A: Maquetar con Cajas flexibles

Objetivo:

Maquetar con la propiedad `display: flex;`

Proceso:

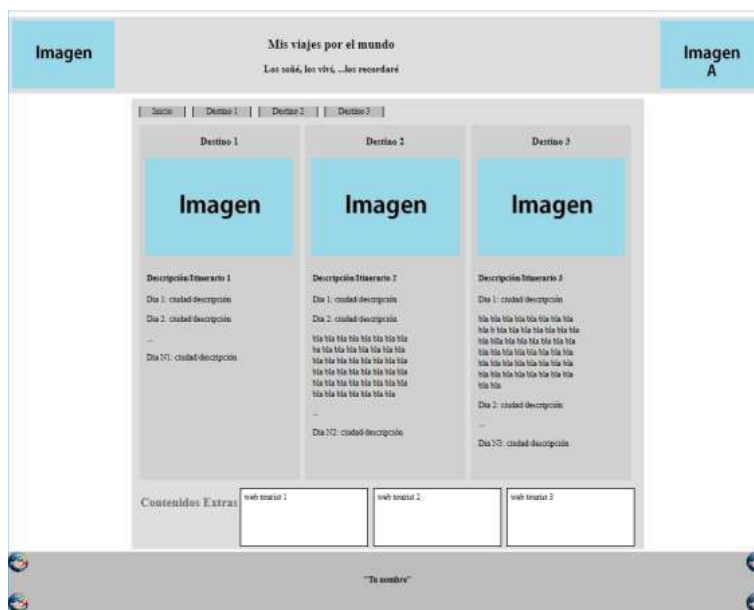
Haz una copia con el nombre **AQ7.1_M3UV**

Se trata de **cambiar** la implementación de TODAS las cajas flotantes **por cajas flexibles**.

La distribución interna de las cajas ha de seguir la misma dirección que ya tenía en vuestra página web original, de forma que la visualización de la página sea lo más parecida posible a la que ya tenáis.

Acordaos de las páginas **fotos.html** y **videos.html**. También debéis usar cajas flexibles.

Os recuerdo que ... Al aplicar la hoja **estiloA.css**, la página se visualizaba así:



prototipo aplicando **columnas.css**

NOTA:

El ejercicio se ha de implementar exclusivamente **con cajas flexibles**, independientemente del resto de valores para la propiedad *display* que hayamos visto en clase.

Atendiendo a lo explicado en los apuntes y documentos anexos:

```
display: inline;          (vs float)
display: block;
display: inline-block; (vs float)
display: table;
display: table-row;
display: table-cell;
```

display: flex; **aplicad esta propiedad**

Bibliografía:

w3c

<http://www.w3.org/TR/css-display-3/>

<http://dev.w3.org/csswg/css-flexbox/#flex-containers>

w3schools

http://www.w3schools.com/cssref/pr_class_display.asp

MDN-Usando las cajas flexibles CSS

https://developer.mozilla.org/es/docs/Web/Guide/CSS/Cajas_flexibles

CSSCTRICKS Una guía completa

<http://css-tricks.com/snippets/css/a-guide-to-flexbox/>

Parte 2: jQuery. Aplicar contenido interactivo

Utiliza la última versión de la librería jQuery: **jquery-3.2.1.js**

Objetivo:

Usar una determinada combinación de propiedades CSS y métodos jQuery para dar cierta interactividad a la página, viendo y analizando la diferencia real entre el tratamiento y las consecuencias de usar, por un lado, la propiedad **visibility** en CSS, y por otro lado, cómo se ve afectada la propiedad **display** al usar diferentes métodos jQuery: **toggle**, **slideToggle**, **fadeToggle**, **animate**, **css**).

A tener en cuenta:

Es muy importante que repases la propiedad CSS *visibility* y la funcionalidad de los métodos jQuery propuestos.

La propiedad CSS **visibility** afecta a la opacidad (pero no a la altura y anchura del elemento), por tanto el espacio ocupado por el elemento se respeta ya que tampoco afecta directamente a la propiedad CSS **display**.

Los métodos jQuery propuestos (toggle, SlideToggle y fadeToggle), afectan a la propiedad CSS **display**, a las dimensiones (*width/height*) e incluso, en ocasiones, a la propiedad CSS **opacity** del elemento:

toggle() afecta al ancho, alto (haciendo que el elemento “desocupe” su espacio) y opacidad del elemento, alcanzando el valor *display:none*
http://www.w3schools.com/jquery/eff_toggle.asp y <http://api.jquery.com/toggle/>

slideToggle() sólo afecta a la altura del elemento, llegando a valer 0px (por tanto, también “desocupa” su espacio), también alcanzando el valor *display:none*
http://www.w3schools.com/jquery/eff_slidetoggle.asp y <http://api.jquery.com/slidetoggle/>

fadeToggle() Visualiza o esconde los elementos seleccionados animando su opacidad. Además *display* alcanza el valor *none*, y por tanto, también “desocupa” su espacio. http://www.w3schools.com/jquery/eff_fadetoggle.asp y <http://api.jquery.com/fadetoggle/>

animate() Ejecuta una animación personalizada de un conjunto de propiedades CSS cuyos valores sean un valor numérico. <http://api.jquery.com/animate/>

css() Permite modificar el valor de una o varias propiedades CSS.
<http://api.jquery.com/css/>

Dependiendo del efecto que queramos conseguir a la hora de tratar elementos que aparezcan/desaparezcan de nuestra web, debemos usar diferentes métodos o propiedades.

Proceso:

Modifica el código de dicha Actividad para incluir los siguientes efectos:

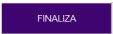
- a. **Inicialmente**, al cargar la página sólo debe aparecer el pie de página. El resto de elementos deben permanecer ocultos. (Utiliza la propiedad *Visibility*).

Así pues, Al cargar por primera vez la página, sólo debería aparecer el pie de página en su sitio, o sea bastante abajo en la página. Tened en cuenta que debéis establecer la propiedad **visibility:hidden** a las secciones cabecera, menú y contenidos/extras.

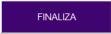
http://www.w3schools.com/cssref/pr_class_visibility.asp

- b. Dentro del pie de página inserta 4 botones <button>, (NO de formulario), cada uno de un color. Con un aspecto *similar* al de la siguiente figura. (por supuesto sigue tu patrón de color)



- c. Cuando el usuario haga clic en cada uno de los tres primeros <button> ("Cabecera", "Menú", "Principal+Extras") aparecerá el correspondiente encabezado, menú y "Principal+Extras". (Recuerda que también había una sección con los contenidos extra)
- d. Cuando el usuario vuelva a hacer clic en cualquier de los tres <button> desaparecerá el correspondiente encabezado, menú y "Principal+Extras" de la pagina.
- e. Cuando el usuario haga clic en el cuarto <button> , desaparecerán los 4 botones (del pie de página) de forma gradual (de uno en uno) quedando en el pie de página un texto:

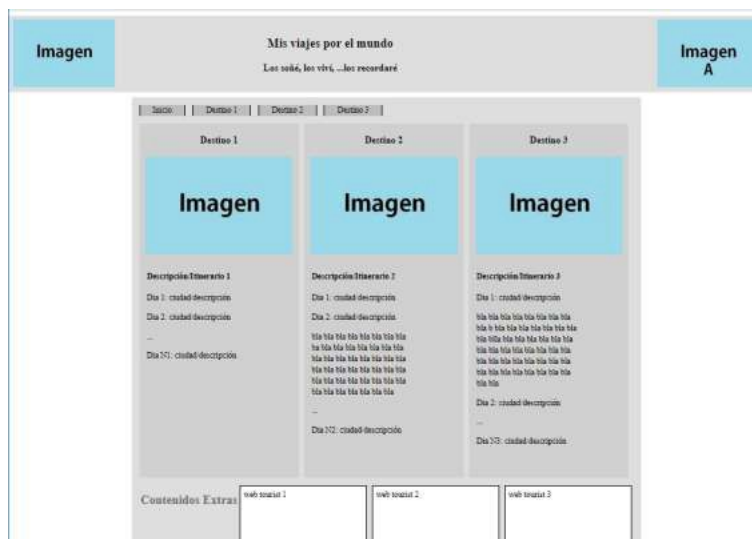
"Aquí va el nuevo pie de página. Haz click para recuperar los botones".

Si el usuario hace clic en el último <button> , antes de mostrar todos las secciones de la página, ya no se podrán mostrar ya que todos los botones quedarán invisibles.

Para poder seguir mostrando/escondiendo las diferentes secciones de la página, debemos mostrar de nuevo los cuatro botones del pie de página.

Para ello implementaremos la función que corresponda, para que al hacer clic sobre el texto del pie de página: **"Aquí va el nuevo pie de página. Haz click para recuperar los botones"**, vuelvan a aparecer los botones y desaparezca este texto.

La Figura muestra la pagina al hacer clic en los tres primeros <button> y luego en el último.



Aquí va el nuevo pie de página. Haz "click" aquí para recuperar los botones

Por supuesto en tu ejercicio deberán aparecer tus diferentes secciones de la página de "Tus Tres Últimos Viajes"

Entregar:

Un archivo comprimido .zip, .rar, .7z llamado

Con todo lo necesario para la correcta visualización de todas las páginas

Apellido1_Apellido2_Nombre_AQ7.1_M3UV

que contenga los archivos necesarios para la correcta visualización y corrección del ejercicio.

Recuerda incluir:

- Patrón de color: URL y documentación básica
- Elección de 2 ó 3 fuentes: documentación básica

Bibliografía:

Api jQuery <https://api.jquery.com/>

w3schools <http://www.w3schools.com/jquery>