

¿Cómo se hizo?

Practica 1: Museo Responsive



**UNIVERSIDAD
DE GRANADA**

Museo Socrates

Grupo 3A (Miércoles)

DAVID MARTÍN CEREZO

Índice

Introducción	3
Estructura de contenidos	4
Estructura de carpetas	4
Desarrollo del Proyecto	5
Uso correcto de marcas html	5
Selección de temática	5
Herramientas utilizadas	5
Implementación	5
Estructura común a todas las páginas	5
HEADER:	6
FOOTER:	8
BANNER:	8
Página index.html	9
PORTADA:	9
DESCRIPCIÓN:	9
CARRUSEL:	10
TARJETAS:	11
Página colección.html	12
Página visita.html	13
Página exposiciones.html	13
Página información.html	14
CALENDARIO:	14
Página experiencias.html	15
Página altausuarios.html	15
Página contacto.html	16
Página colecciones_x.html	16
Página obra_x.html	17
Aspectos Innovadores	18
Animaciones CSS	18
Propiedades CSS	18
Iframe HTML	18
Checkbox hack	18
Grid y flexbox	19
Propiedades -webkit-prefixed	19
Media queries	19
Conclusión	20

Introducción

En este documento detallaré en profundidad la realización de la práctica, el cual dividiré en estructura, desarrollo, aspectos que considere innovadores y por último la conclusión.

El proceso de creación de la página web del Museo Sócrates ha sido una experiencia retante y creativa. He tomado mi inspiración del Museo de Animal Crossing, decidí llevar esa misma sensación amigable a la web.

El Museo Sócrates, nombrado en honor al famoso filósofo griego y al guía del museo. Como mascota del museo, he utilizado a Sócrates, un sabio búho que guía a los visitantes en su búsqueda de conocimiento y arte.

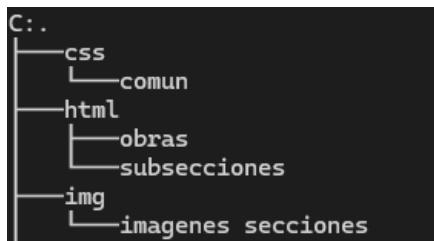
Al diseñar la página web, utilicé una combinación de imágenes propias y recursos seleccionados de Internet sin derechos de autor o en estado de uso legítimo para crear una experiencia visualmente atractiva y enriquecedora. Cada sección del museo fue diseñada de acuerdo con las exigencias del guión sin dejar de lado un punto de creatividad.

A través de este proyecto, he explorado en profundidad las posibilidades creativas de HTML5 y CSS3. He incorporado algunos elementos innovadores para mejorar la experiencia del usuario. Desde la implementación de un diseño adaptable hasta la creación de funciones interactivas, con numerosos detalles con el objetivo de ofrecer una experiencia inmersiva y memorable.

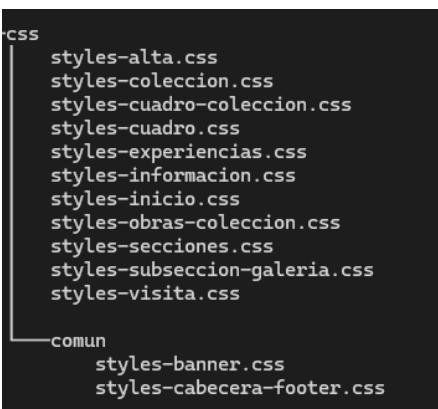
Estructura de contenidos

Estructura de carpetas

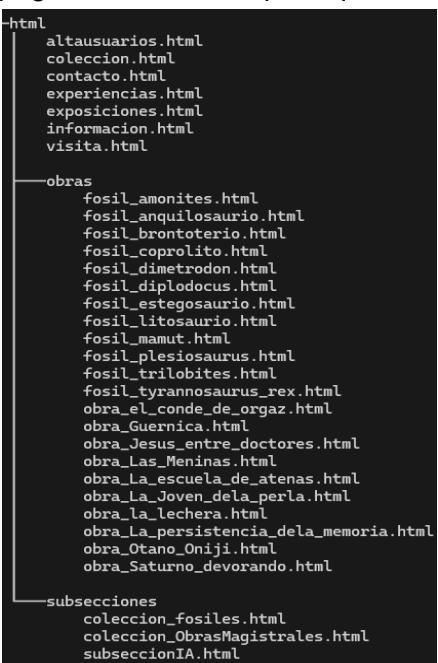
Para la organización del proyecto he utilizado la siguiente estructura de carpetas:



- En la carpeta “css” están todos los estilos del museo



- “comun” lo utilizo para estilos que se aplican a muchos html como por ejemplo el estilo del header y footer o los banner que utilizo para decorar la el museo.
- “html” como su nombre indica almacena todos los archivos .html que componen la página subdivididos para que trabajar con los enlaces relativos sea más cómodo.



Desarrollo del Proyecto

Uso correcto de marcas html

He seguido las instrucciones de diseño que se daban en el guión de la práctica como se indican en cada apartado de la web. Todos los archivos HTML de esta práctica han sido verificados en <https://validator.w3.org/> para comprobar que cumplan con los estándares.

Selección de temática

Para el museo pensé en diversas temáticas, se me sugirió hacerlo basado en el museo de el famoso videojuego de Nintendo “Animal Crossing” lo que me pareció una idea muy amigable y que a la vez facilita las colecciones, experiencias y visita al museo al estar basado en algo ya existente.

Herramientas utilizadas

Para el desarrollo de un proyecto complejo como este ha sido de mucha ayuda ciertas herramientas que facilitan el desarrollo:

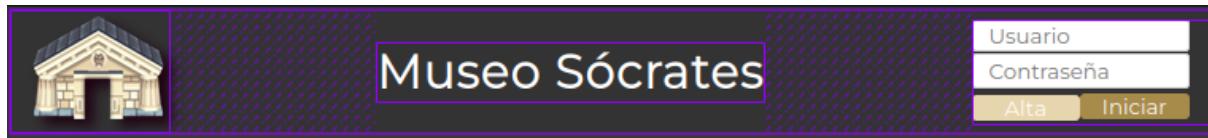
- Todo la práctica ha sido desarrollada en Visual code Studio
- Para facilitar la estructuración en ocasiones he utilizado generadores de CSS Grid y Flexbox aunque en muchos casos he acabado adaptándome a mi gusto.
- Páginas web como W3Schools y CodePen para tomar inspiración y aprender a realizar diseños complejos.
- Procesadores de texto basados en GPT para debugging y generación del texto e información de parte de la página web.
- El navegador principal en el desarrollo ha sido Chrome pero el más utilizado para debugging ha sido Firefox ya que tiene un modo desarrollador muy útil

Implementación

Estructura común a todas las páginas

Como se indicaba en el guión de la práctica el header y el footer debían ser comunes a todas las páginas por lo que opte por hacerlos en un hoja de estilo propia y aplicarla en todos los html necesarios, esto facilita las modificaciones al no tener que tocar cada hoja de estilo de cada página.

HEADER:



Lo he realizado de la siguiente manera:

```
<!-- HEADER -->
<header>
  <ul class="head-top">
    <!-- Logo -->
    <li>
      | | <a href="index.html"></a>
    </li>
    <!-- Nombre museo -->
    <li class="nombre-museo">
      | | <h1>Museo Sócrates</h1>
    </li>
    <!-- Login -->
    <li class="form-head-top">
      | | <form id="loginform" action="html/altausuarios.html">
      | |   <input type="text" name="usuario" placeholder="Usuario" required>
      | |   <input type="password" placeholder="Contraseña" name="password" required>
      | |   <button type="submit" title="Alta" name="Alta">Alta</button>
      | |   <button type="submit" title="Inicio" name="Inicio">Iniciar</button>
      | |
    </li>
  </ul>
```

```
.head-top{
  list-style-type: none;
  display: flex;
  justify-content: space-between;
  align-items: center;
}
```

He creado una lista desordenada **** para tener los elementos ordenados, después mediante css y las clase que le asigne aplique a esa lista desordenada un **display: flex** y **justify-content: space-between** esto hace que el contenido se espacien entre sí, **align-items: center** lo utilizo para colocar de forma facil en nombre del museo de manera centrada, todo esto resulta en la parte superior del header.

Al logotipo del museo le he aplicado un **drop-shadow** para darle un profundidad.

```
button[type=submit] {
  width: 90%;
  background-color: var(--logo-color);
  color: white;
  margin: 2px 0;
  border: none;
  border-radius: 3px;
  cursor: pointer;
}

#loginform button[type="submit"] {
  width: 45%; /* Boton doble ancho de ... */
```

El formulario de inicio de sesión o alta lo he realizado como se puede ver anteriormente mediante un formulario que de forma provisional redirecciona a "**altausuarios.html**". Sobre el estilo de este he colocado los botones mediante **display: block** para que estuvieran como se puede ver, también he quitado los bordes con, para colocar los botones de inicio y alta paralelos mediante reducir su ancho a menos de la mitad y se alineen a la misma altura.

La segunda parte del header es la barra de navegación, la cual diseñe de la siguiente manera:

Para pantallas grandes es una barra clásica creada con **ul** y **justify-content: space-between;** para que se vea esta manera

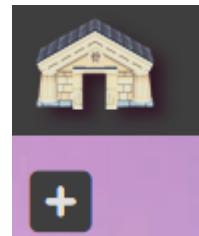
Inicio	Colección	Visita	Exposiciones	Información	Experiencias	
--------	-----------	--------	--------------	-------------	--------------	--

```
<!-- Barra de navegación -->
<nav class="nav-menu">
    <input type="checkbox" id="check">
    <label for="check" class="checkbtn">
        <i class="fa-solid fa-circle"></i>
        <i class="fa-solid fa-square-plus"></i>
        <i class="fa-solid fa-square-minus"></i>
    </label>
    <ul class = "nav-list">
        <li class="ini"><a href="index.html">Inicio</a></li>
        <li class="des"><a href="html/coleccion.html">Colección</a></li>
        <li class="des"><a href="html/visita.html">Visita</a></li>
        <li class="des"><a href="html/exposiciones.html">Exposiciones</a></li>
        <li class="des"><a href="html/informacion.html">Información</a></li>
        <li class="des"><a href="html/experiencias.html">Experiencias</a></li>
    </ul>
</nav>
```

Pero la parte interesante y más compleja de la navegación es su variante responsive para dispositivos móviles.

Para aprovechar la pantalla de mejor manera en móviles opte por un menú desplegable que solo se muestra cuando el usuario quiere navegar en la web.

Haciendo uso de las “*media queries*” aplicó al **** las siguientes modificaciones cuando la pantalla es menor de 760px:



1. El botón del menú (**checkbtn**) se vuelve visible.
2. La lista de navegación se muestra como una lista en bloque y se posiciona fija en la parte superior de la pantalla, cubriendo toda la pantalla cuando se activa.
3. Los elementos de la lista se muestran uno debajo del otro y ocupan todo el ancho disponible.
4. Se muestra un ícono de "menos" encima del ícono de "más" cuando el menú está activo `#check:checked ~ .checkbtn .fa-square-minus {display:block;}`

He utilizado la etiqueta **<i class="">** para crear los botones mediante una librería de iconos llamada “Font Awesome” :

- **fa-square-plus** lo utilizo por defecto para mostrar al usuario que puede ver más información dándole a ese botón
- **fa-square-minus** lo utilizo para superponerlo a el anterior al abrirse el menú y mostrar que volviendo dar al botón se quita el menú mostrado
- **fa-circle** lo utilizo de forma estética para dar más contraste a los botones anteriores ya que son transparentes en su interior.

FOOTER:

```
Realizado por D. Martín Cerezo ©
```

```
¿Como se hizo?
```

El footer está formado por dos partes con sus respectivos link las cuales estan estilizados mediante **display: flex** y **justify-content: space-between**. Tambien utilizo un “*media queries*” para ajustar los tamaños de texto a las pantallas pequeñas.

```
<!-- FOOTER -->
<footer>
  <p>Realizado por <span><a href="html/contacto.html">D. Martín Cerezo </a></span>@.</p>
  <span><a href="como_se_hizo.pdf">¿Como se hizo?</a></span>
</footer>
```

BANNER:

```
<!-- Sección de fija del museo -->
<section id="Coleccion" class="presentacion">
  <div class="it-txt"><h1> ¡Nuestro Museo! </h1></div>
  <div class="it2-img"></div>
  <div class="banner"></div>
</section>
```

Justificación de div: Los **divs** se utilizan como contenedores para organizar y estructurar el contenido de la sección. Cada div contiene un tipo específico de contenido (texto, imagen, banner) que se puede estilizar y posicionar individualmente.

Para estilizar establezco los estilos generales en la clase “*presentacion*”: **display: flex**, el centrado vertical y horizontal de los elementos, y márgenes tanto arriba como abajo para dar espacio a la sección.

Los estilos específicos se aplican a los elementos internos como el texto **.it-txt** y la imagen **.it2-img**. Después defino un banner con una imagen de fondo **.banner** que se posiciona de manera absoluta detrás del contenido principal.

Por último utilizo “*media queries*” para ajustarlo a pantallas pequeñas



Como se puede ver el texto tiene un efecto de degradado el cual he creado mediante el siguiente código:

```
-webkit-text-fill-color: transparent;
-webkit-background-clip: text;
```

Teniendo un **background** (en mi caso un **gradient**) la primera crea una máscara con el texto que sólo deja ver el fondo en el texto combinado con la segunda propiedad que hace transparente el texto crea el efecto que se puede observar más arriba. Este lo utilizaré numerosas veces en toda la página web, por no ser redundante solo lo mencionare aquí.

La imagen ilustrativa de la sección tiene una animación básica para dar dinamismo la cual acerca y aleja la imagen la cual mediante la propiedad para animaciones **infinite** hace que ocurra de manera infinita como indica su nombre.

```
@keyframes animation {
  0% { transform: scale(1); }
  50% { transform: scale(1.05); }
  100% { transform: scale(1); }
}
```

Página index.html

PORTADA:

Para la página de presentación del museo he decidido hacer un efecto parallax y para la explicación de museo he realizado una simulación de texto de videojuego en el que la mascota del museo habla. Esto lo he realizado de la siguiente manera:

Sección de Presentación con Efecto Parallax:

Esta sección (cont1) tiene un fondo de imagen con un efecto parallax: **y position:relative** para poder ubicar objetos **absolute**.

Contiene una imagen de Sócrates **div-socra-c1** y texto de bienvenida **div-texto** estilizado con degradados anteriormente explicados colocados en el **cont1** con **position:absolute** para poder ubicarlos de manera libre.

También incluye animaciones para la aparición gradual de los elementos y el efecto de zoom al pasar el cursor sobre la imagen de Sócrates.

```
background-image: url('../img/mu  
background-attachment: fixed;  
background-position: center;  
background-repeat: no-repeat;  
background-size: cover;
```

Justificación de div: Los **divs** se utilizan en este caso para poder ubicar la imagen y texto en en la sección de manera gráfica.

DESCRIPCIÓN:

Para la descripción de la página reutilice el efecto parallax que ya había implementado en la portada y en este caso quise hacer una descripción más creativa con un efecto de bocadillo como si te estuviera hablando en el videojuego.

Lo realice de la siguiente manera:

1. **.div-bocadillo2** contiene un bocadillo de diálogo **img** y cuatro párrafos de texto que simulan un diálogo con Sócrates. Estos párrafos tienen clases específicas **line-2** para controlar su estilo y animaciones.
2. Las animaciones **anim-escrib1**, **anim-escrib2**, **anim-escrib3**, **anim-escrib4** controlan la aparición gradual de cada línea de texto simulando que Sócrates está hablando.
3. Además, se defino una animación **fadeln** para la entrada gradual de los elementos de la sección y otra **desaparecer**) para ocultar los elementos.

La animación **blinkTextCursor** lo que hace es pasar el color del borde derecho de la caja de texto de negro a transparente para dar la sesacion de animacion de escribir.

```
@keyframes blinkTextCursor{  
from{border-right-color: black;}  
to{border-right-color: transparent;}}  
}
```

La animación **escrib** aumenta el ancho de la caja de texto para que vayan apareciendo las letras gradualmente.

```
@keyframes escrib{  
from{width: 0;}  
to{width: 19.3em;}}
```

Estas dos animaciones juntas crean el efecto de escritura en la animación anteriormente mencionada.

CARRUSEL:

Al visitar páginas web de museos para tomar inspiración y aprender formas de utilizar el css, observe que en casi todos los museos había al menos un carrusel con imágenes que iba mostrando novedades y partes destacadas del museo. Por lo que me pareció interesante y un reto considerable para añadir a mi página web.

Lo realice de la siguiente manera:

Este bloque de código HTML y CSS crea un carrusel de novedades con tres elementos deslizables. Aquí hay una descripción detallada de cada parte:

Parte HTML:

- La sección principal con la clase **ContainerCenter** que contiene todo el carrusel.
- Tres entradas de radio (`input`) con los IDs **Slider1**, **Slider2**, y **Slider3**, respectivamente. Estos se utilizan para controlar qué elemento del carrusel se muestra actualmente.
- Dentro de la sección de carrusel Carrusels, hay tres elementos **.Carrusel**, cada uno con una imagen de fondo y un artículo que contiene información sobre el mismo.
- También hay flechas de navegación a la izquierda y a la derecha **ArrowLeft** y **ArrowRight** para desplazarse por los elementos del carrusel.
- Finalmente, hay un conjunto de **selectores** que muestran cuál de los elementos está actualmente seleccionado.



Parte CSS:

- Defino estilos para las secciones del carrusel **Carrusels** y para cada elemento dentro de ellas **Carrusel**. Estos incluyen el tamaño, la posición y el efecto de transición para cada vez que se pulsa la las fechas o los selectores inferiores.
- Estos estilos se ajustan dependiendo del estado del selector de radio **:checked**. Esto permite que los elementos del carrusel se desplacen cuando se selecciona un radio.

Funcionamiento del carrusel:

La parte que presenta un reto del carrusel es su funcionamiento en si ya que hay que jugar con las propiedades css, checkbox y los label para hacerlo funcionar, esto se logra de la siguiente manera:

- El carrusel utiliza la técnica de desplazamiento horizontal para cambiar entre las diferentes imágenes y sus respectivas descripciones.
- Las imágenes y las descripciones están encapsuladas dentro de contenedores **<div class="Carrusel">** dentro de la sección **Carrusels**.
- El desplazamiento se realiza cambiando la propiedad **transform: translate(x,x)** de cada contenedor **.Carrusel** utilizando transiciones CSS.
- Cuando se hace clic en una de las entradas de radio (**Slider1**, **Slider2**, **Slider3**), se activa un estado específico del carrusel, por ejemplo **{ transform: translate(200%, 0); }** lo que provoca que se desplace a una posición específica donde se muestra el contenido correspondiente.

Justificación de div: Los **div** en este caso los utilzo para rotar las opciones del carrusel dentro de su sección.

TARJETAS:

Cuando estaba diseñando la página al terminar el carrusel me dio la sensación de que era un final muy abrupto para una página principal en la que se esperaría más información por lo que al final de index.html decidí crear unas tarjetas con información atractiva del museo, esto lo realice de la siguiente manera:

- En primer lugar defino un contenedor principal llamado .parent con una rejilla (`display: grid`) que se adapta automáticamente al tamaño del contenedor y que tiene una separación entre cajas del 5%; **gap: 5%**.
- El ancho de las columnas de la rejilla se establece utilizando `grid-template-columns` con la función **repeat(auto-fill, minmax(20rem, 2fr))**, lo que significa que cada columna debe tener un ancho mínimo de 20 rem y un ancho máximo de 2 fracciones del espacio disponible que al poner una pantalla grande completa con 4.
- Se aplican estilos a las imágenes `` para que tengan un tamaño máximo y una apariencia adecuada, incluida una sombra cuando se pasa el ratón sobre ellas.
- También aplico estilos para los elementos de texto como los párrafos `<p>` y negritas ``

Por ultimo debido a que tuve algunos problemas debido a que el css grid hacia overflow sobre el footer tuve que adaptar este para evitar que tapara la parte inferior

- Aumentó gradualmente el margen inferior del contenedor que forma el css grid **.parent**
- Para dispositivos móviles directamente desactivo el css grid y lo transformó en **display:flex** en forma de columna con el margen inferior en automático para que evite el footer.

Por último sobre el index.html cabe destacar que la parte con información la introduce dentro de una etiqueta `<main>` para tener la información y la presentación del museo separadas y poder aplicar un margen que considere.

```
@media all and (max-width: 1250px) {  
    .parent {  
        margin-bottom: 10vh; /* 10% del alto de la pantalla */  
    }  
}  
  
@media all and (max-width: 1050px) {  
    .parent {  
        margin-bottom: 40vh; /* 30% del alto de la pantalla */  
    }  
}  
  
@media all and (max-width: 600px) {  
    .parent {  
        display: flex;  
        flex-direction: column;  
        margin-bottom: auto; /* Evita que el footer se empile debajo */  
    }  
    .parent section{  
        margin-bottom: 5%;  
    }  
}
```



Pintura histórica

Explora nuestra colección de pinturas clásicas y sumérgete en la historia con esta icónica obra maestra que retrata el legendario entierro del Conde de Orgaz, una joya del arte renacentista.



La diversidad del mar

Descubre la fascinante vida marina a través de nuestra sección dedicada al océano, donde podrás explorar la biodiversidad submarina y conocer más sobre la extraordinaria especie del pez Jurel.



La belleza de la naturaleza

Sumérgete en el mundo natural y contempla la gracia y elegancia de los insectos en nuestra colección de arte inspirada en la naturaleza, donde cada detalle te transportará a la majestuosidad del reino animal.



El legado del pasado

Explora el fascinante mundo de la paleontología y sumérgete en un viaje a través del tiempo para descubrir los misteriosos secretos que los fósiles guardan sobre la evolución y la historia de nuestro planeta.



Escultura clásica

Admira la obra maestra del Renacimiento italiano, el icónico David de Miguel Ángel, y sumérgete en el mundo de la escultura clásica en nuestra destacada sección dedicada a las esculturas que han marcado la historia del arte.



Arquitectura histórica

Explora la grandeza de la arquitectura antigua con nuestra selección de monumentos emblemáticos, como el majestuoso Partenón, que te transportará a épocas pasadas y te maravillará con su belleza y esplendor.

Página colección.html

Para realizar esta página del museo me base en la del museo del prado que es la referencia que se da en el guión, para hacer que cambiara la información del interior de "la ventana de búsqueda" lo que hice fue utilizar un **<iframe>**

- Utiliza una estructura de **<section>** .parent como contenedor principal estructurado con un **CSS grid** que da formato a la ventana de navegación.
- Dentro de este contenedor, hay dos **<section>** secundarios: uno para el menú **sidebar** y otro para el contenido principal **body**.
- El menú **sidebar** contiene los enlaces agrupados por categorías para poder navegar fácilmente entre las obras del museo cambiando entre las colecciones que a su vez redirigen a las obras.
- También incluyo un **<input> checkbox** que se oculta visualmente para pc, el cual se utiliza para mostrar u ocultar el menú cuando se hace clic en la etiqueta **Filtrar** para dispositivos móviles.
- En dispositivos móviles la estructura de la página (.parent) pasa a ser un **display:flex** en forma de columna para adaptar a dispositivos móviles y facilitar el menu desplegable de navegación en filtrar

Menú desplegable:

1. Cuando se hace clic en el texto "**Filtrar**", la etiqueta **<label>** está asociada al **checkbox** (el cual está oculto) y, por lo tanto, hace que el checkbox cambie de estado (por defecto está desmarcado).
2. La regla CSS **.toggle-checkbox:checked + .collapsible + .sidebar** se encarga de mostrar u ocultar el menú **sidebar** dependiendo del estado del checkbox.
3. Si el checkbox está marcado **checked**, el menú se muestra **display: block**; si no está marcado, el menú se oculta **display: none**.

Filtrar

TIPO

- Pintura
- Paleontología
- Acuario
- Entomología
- Exposiciones

OBJ

La característica más sobresaliente de Dimetrodon es la enorme vela dorsal, similar a una aleta.

Vivió en lo que es hoy el oeste de Norteamérica durante el período Jurásico Superior.

AnteriorSiguiente

Página visita.html

Para realizar la página visita de la web quise poner el mapa de guia y unas imágenes para hacer esta pagina mucho más visual, lo he realizado de la siguiente manera:

Utilizó la etiqueta **<section>** para dividir la página en cada sección del museo teniendo en el inicio un mapa con los nombre tenido estos un texto indicando a qué parte corresponde cada ubicación del mapa, estos tienen cada uno un enlace que redirige a cada sección (mediante **id="X"**) del museo en la que hay una pequeña información de la sección junto con una imagen de la misma.



Página exposiciones.html

Para la página de exposiciones he hecho unas tarjetas que ocupan el ancho de pantalla con la información y una imagen de la exposición inspirado en las del museo Thyssen.

Las realice de la siguiente manera:

Luego, dentro de la etiqueta **<main>**, hay dos secciones **.parent** estructurado con un **CSS grid repeat(auto-fill, minmax(100%, 1fr))**, cada una contiene varias secciones de exposiciones. Cada sección de exposición **section** contiene un enlace **<a>** que lleva a una página de la colección general.

Tambien se establecen estilos de transición y cambio de color al pasar el mouse sobre las secciones de exposiciones.

Por último utilizo media queries para ajustar el tamaño de las imágenes en dispositivos con pantallas más pequeñas. En dispositivos móviles, las imágenes se ajustan para que ocupen menos porcentaje del ancho de la pantalla.



Página información.html

Para la página de información tome el ejemplo, como en la página anterior, del museo Thyssen la cual me pareció muy concisa y limpia. Está la realice de la siguiente manera: La página la encapsule en una etiqueta **<main>** la cual dentro tiene un sector con un **css grid** como ya realice en páginas anteriores el cual tiene las siguientes propiedades **repeat(auto-fill, minmax(20rem, 3fr));** estas propiedades hacen que sea automáticamente responsive.

CALENDARIO:

El calendario está organizado en tres secciones principales, una para cada mes (abril, mayo y junio). Cada sección tiene una clase **.mes** para estilizar.

Dentro de cada sección del mes, se encuentra una tabla **<table>** que muestra los días de la semana y los días del mes. Los días del mes se presentan en filas **<tr>** y columnas **<td>**, donde cada celda representa un día específico.

Para diferenciar los días cerrado y abierto he definido clases especiales para los días con eventos importantes, como días de entrada gratuita **** con el color del museo y días de cierre **** en color rojo.

Calendario de apertura						
ABRIL 2024						
L	M	X	J	V	S	D
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Por último he añadido una sección con una leyenda para entender mejor el calendario

LEYENDA		
Día normal: 27	Día gratuito: 27	Cerrado: 27

Página experiencias.html

Para la página de experiencias he vuelto a utilizar css grid con minmax que cumple la función que necesito para mostrar las tarjetas de opiniones de forma responsive, lo realizado de esta manera:

- Cada opinión de visitante está encapsulada dentro de un **<section class="section0">**. Cada sección contiene:

- Una imagen del visitante.
- Un artículo con el nombre del visitante y su calificación en forma de monedas hecho mediante imágenes a las cuales algunas se les pone un efecto en blanco y negro y a otras no según la calificación.
- Un párrafo con la descripción de la experiencia del visitante.



Al final de las opiniones, hay un formulario **<section class="reseña-form full-width">** para que se aplique el formato del css y además la clase **full-width** con **grid-column-start: 1** y **grid-column-end: 0** estas propiedades lo que hacen que se utilice al completo la cuadrícula del grid.

The screenshot shows a user profile for 'Canela' with a 5-star rating. Below it is a text area for sharing experiences with fields for name and text, and a yellow 'Enviar' button.

Página altausuarios.html

Para la página de registro he hecho un formulario flotante en la página centrado en el medio con unos campos para llenar con campos cada uno adecuado a la información que se solicita todos los campos **required** y un **checkbox** con los términos y condiciones simulado, también he añadido una opción para el caso de no haber entrado correctamente en el alta usuarios.

```
<section class="form-register">
  <h4>Formulario Registro</h4>
  <!-- Implementar etiqueta form para siguiente práctica -->
  <form action="procesar_registro.php" method="POST">
    <input class="controls" type="text" name="nombres" id="nombres" placeholder="Ingrese su Nombre" required>
    <input class="controls" type="text" name="apellidos" id="apellidos" placeholder="Ingrese su Apellido" required>
    <input class="controls" type="email" name="correo" id="correo" placeholder="Ingrese su Correo" required>
    <input class="controls" type="password" name="correo" id="pwd" placeholder="Ingrese su Contraseña" required>
    <input class="controls" type="password" name="correo" id="pwd2" placeholder="Confirme su Contraseña" required>
    <label class="container">
      <input type="checkbox">
      <span class="checkmark">Estoy de acuerdo con los <a href="#">Términos y Condiciones</a></span>
    </label>
    <input class="botons" type="submit" value="Registrar">
  </form>
  <p><a href="#">¿Ya tengo Cuenta?</a></p>
</section>
```

The screenshot shows a floating input field for 'Nombre' with a placeholder 'Ingrese su Nombre'. To its right is a checkbox labeled 'Estoy de acuerdo con los Términos y Condiciones' and a large yellow 'Registrar' button. At the bottom right is a link '¿Ya tengo Cuenta?'.

Página contacto.html

La página de contacto usa la misma hoja de estilos css que altausuario.html por lo cual es muy similar pero en este caso el formulario hay campos que no están **required** y el formulario se envía por correo. También he incluido una imagen de mi y un subtítulo para hacer un poco más interesante el formulario.



Al **textarea** le he aplicado un estilo `field-sizing: content;` este ha sido recientemente añadido a css el cual ajusta el texto al contenido que hay escrito.

```
<h4>Formulario de Contacto</h4>
<form action="mailto:davidmcerezo@corre.ugr.es" method="post" enctype="text/plain">
    <input class="controls" type="text" name="nombres" id="nombres" placeholder="Ingrese su Nombre" required>
    <input class="controls" type="text" name="apellidos" id="apellidos" placeholder="Ingrese su Apellido">
    <input class="controls" type="email" name="correo" id="correo" placeholder="Ingrese su Correo" required>
    <input class="controls" type="text" name="asunto" id="asunto" placeholder="Asunto del Correo" required>
    <textarea name="mensaje" class="controls" id="mensaje" placeholder="Redacte aquí su mensaje"></textarea>
    <input class="botones" type="submit" value="Enviar">
</form>
<p>Gracias!</p>
```

Página colecciones_x.html

Para las páginas de colecciones decidí usar la misma estructura **CSS grid** la ya dominaba de páginas anteriores como index,informacion, etc. Ademas esta es perfecta ya que al comodarse al tamaño del index de colecciones.html da muy buen resultado tanto en dispositivos móviles como pc.

Las tarjetas con los obras para visitar las he hecho de la siguiente manera:

1. En el HTML, cada obra de arte está dentro de un `<a>` para hacerla clickeable y dirigir al usuario a una página con más detalles sobre la obra.
2. Cada obra se encuentra dentro de un `<section>` con una clase específica que determina su apariencia visual y diseño en la página.
3. La información de cada obra (título, autor y fecha) se muestra dentro de un `<article>` para mantener la semántica.
4. Incluyo también reglas de estilo adicionales para asegurar una buena experiencia de usuario en dispositivos móviles mediante media queries que ajustan el diseño según el ancho de la pantalla.



Saturno devorando a su hijo

Francisco de Goya
1820-1823



Ōtani Oniji III

Tōshūsai Sharaku
1794



Las Meninas

Diego Velázquez
1656

Página obra_x.html

Para la página de las obras he decidido poner una imagen de la máxima calidad posible en el inicio para que se vea a detalle el cuadro junto a una pequeña descripción y una parte de recomendaciones implementadas con iframe para que se pueda navegar con facilidad en el museo, lo he realizado de la siguiente manera:

- El contenedor principal se organiza con **flexbox** y **column** para distribuir los elementos verticalmente.
- La imagen de la obra se muestra dentro de **<section class="image-container">**.
- Los detalles de la obra, como el título, autor, fecha y descripción, se encuentran dentro de **<article class="description">**.
- También se incluye un **<iframe>** que muestra obras relacionadas situado en la derecha de la página en pc y en la parte de abajo en móvil.
- Por ultimo utilizo media queries para ajustar la pagina para dispositivos móviles

53

Saturno devorando a su hijo

Autor: Francisco de Goya
Fecha: 1820-1823
Descripción: Representa al titán Crono, como es habitual indiferenciado de Chronos, o Saturno en la mitología romana, en el acto de devorar a uno de sus hijos. La figura era emblema alegórico del paso del tiempo, pues Crono se comía a los hijos recién nacidos de Rea, su mujer, por temor a ser destronado por uno de ellos.

Más información: El tema de Anabella está relacionado, según Freud, con la melancolía y la destrucción, y estos rasgos están presentes en las Pinturas negras. Con expresión terrible, Goya nos sitúa ante el horror caníbal de las fauces abiertas, el gigante avejentado y la masa informe del cuerpo sanguinolento del supuesto hijo.

Obras relacionadas



Guernica

Pablo Picasso
1937



La joven de la perla

Johannes Vermeer
1665-1667

Aspectos Innovadores

Animaciones CSS

En esta práctica he utilizado numerosas veces las animaciones css que aportan un aspecto muy interactivo y divertido a las páginas web.



Propiedades CSS

Al informarme para realizar la práctica explore muchas propiedades css las cuales no se habían visto en clase y son tremadamente útiles para solucionar problemas o hacer efectos sorprendentes.

- field-sizing: content; para adaptar un textarea a lo escrito.
- Posicionamiento: relative, absolute, fixed, etc.
- Las propiedades background:background-attachment:fixed; background-position:center; etc...
- Efectos de sombreado text-shadow, box-shadow, drop-shadow...

Iframe HTML

En mi museo virtual he usado en varias páginas los iframe para tener una ventana hacia otra página, no se explico en clase, se incitó a que lo utilizáramos y a mi me ha resultado muy útil.

Checkbox hack

Este es simplemente el nombre que le da la comunidad a utilizar checkbox y label para realizar cosas que no son formularios como por ejemplo el menú desplegable para móvil o el botón de Filtrar desplegable que he implementado en la práctica.

También lo he utilizado para hacer el carrusel de exposiciones.

Grid y flexbox

Uno de los pilares de mi práctica ha sido la utilización de css grid y flexbox para estructurar todas las páginas y hacerlas responsive, aunque es cierto que se mencionaron en clase para dar pistas al empezar esta práctica no se vieron en profundidad.

Propiedades -webkit-prefixed

Aunque estas no son nativas y las versiones más nuevas no están actualizadas en todos los navegadores son muy útiles para hacer ciertas cosas que no están por defecto en CSS.

Yo he utilizado:

```
-webkit-text-fill-color: transparent;  
-webkit-background-clip: text;
```

¡Nuestro Museo!

Media queries

Para hacer el diseño lo más responsive posible ha sido fundamental el uso de las media queries que no se explicaron en clase pero son muy importantes para hacer una página totalmente responsive.

Conclusión

Probablemente esta práctica haya sido en la que más tiempo he invertido de la carrera sumando documentación y desarrollo, sin embargo, ha sido una experiencia enriquecedora que he disfrutado, ha sacado mi lado más creativo a la hora de hacer la página. El proceso de hacer la práctica ha sido lento, aun empezando desde el primer día, ya que he descartado y cambiado muchas partes de la web a lo largo de este mes. Esta página aunque bebe mucho de otras web de museos famosos sigue teniendo mi estilo personal en cuanto a colores y diseño en general.

Todo esto me ha hecho adquirir unos conocimientos de css y html muy aceptables en un mercado en el que muchos desarrolladores utilizan funcionalidades como *Tailwind CSS* o *bootstrap*.

En resumen se me ha quedado un buen sabor de boca respecto al desarrollo front-end aun siendo css un lenguaje muy confuso y complicado al principio, por esto seguiré investigando y actualizándome en este ámbito, y ¿por qué no?, barajar la opción de desarrollarme profesionalmente en este campo.