

Laboratorio #1

HTML, DOM y CSS

1. HTML

HTML es un lenguaje de marcado para la elaboración de documentos HTML (páginas web). Describe la estructura y el contenido semántico de un documento web. El contenido dentro de un documento HTML está etiquetado con elementos como `<head>`, `<title>`, `<body>`, `<p>`, `<div>`, entre otros.

Códigos ISO Latin-1 pueden ser ubicados en el código HTML para escribir códigos especiales (acentos, diéresis, entre otros) y símbolos que no pueden ser generados en HTML regular. A continuación, el preparador se encargará de explicar el código ejemplo.

Un documento HTML está compuesto por elementos HTML que definen su estructura y despliegue en el navegador. La mayoría de los elementos HTML están compuestos por una etiqueta de inicio y fin que define su contenido.

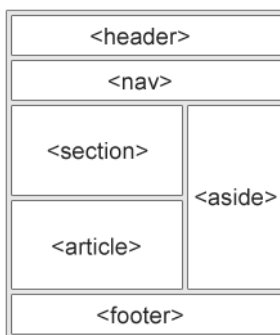
`<etiqueta>contenido</etiqueta>`

Por ejemplo, los elementos: `` y `<table>`.

Un elemento HTML puede poseer atributos. Un atributo provee información adicional de un elemento y se define en la etiqueta de inicio del elemento HTML. Los atributos están especificados en forma de pares nombre/valor.

`<html lang="es-ES">`

El elemento HTML `<div>` define una división o una sección en un documento HTML, usualmente utilizado como un contenedor para otros elementos HTML. El elemento `<div>` es considerado no-semántico. Mientras que los elementos `<header>`, `<nav>`, `<section>`, `<article>`, `<aside>` y `<footer>` son considerados elementos semánticos. Los elementos semánticos son elementos con significado, ya que la propia etiqueta declara el tipo de contenido que posee, por ejemplo:



Un formulario HTML es utilizado para adquirir datos de los usuarios. El elemento `<form>` define un formulario, y pueden utilizar diferentes mecanismos para solicitar los datos de entrada como lo son casillas

de verificación o los botones de radio, y permite definir botones de envío para enviar desde el formulario los datos adquiridos a un servidor.

Dentro de un formulario podemos encontrar varios elementos:

- `<textarea>`
- `<select>`
- `<input type="valor">`: text, radio, checkbox, submit, reset, search, email, phone, url, tel, range, number, date, datetime, datetime-local, month, week, time, color.

Nombre:

Sexo:

☐ Masculino

☐ Femenino

Número de miembros de la familia:

Idiomas que conoce:

☐ Francés

☐ Inglés

Países: Colombia ▾

Otros:

Palabra clave:

Configuración de Audio:

Volumen

Escoge una semana: Semana --, 2018

Escoge el color:

junio de 2018 ▾ ↑ ↓

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

Borrar Esta semana

Palabra clave:

Configuración de Audio:

Volumen

Escoge una semana: Semana --, 2018

Escoge el color:

○

230

R

100

G

101

B ↕

Además se pueden incluir atributos como *pattern* donde el valor del campo debe cumplir un patrón que se define mediante expresiones regulares, o *placeholder* que sirve para incluir un texto por defecto en el campo, o *required* que permite indicar que el campo es obligatorio, o *autofocus* para indicar que el campo tiene por defecto el foco. Por ejemplo:

```
<input type="text" name="quantity" pattern="[0-9]*" placeholder="Ingrese un número"/>
```

También tenemos las etiquetas `<audio>` y `<video>` que nos permiten incrustar video y audio fácilmente en nuestra página web. Además, nos proporcionan toda una API para poder controlar su manera de visualización en la página. Un ejemplo de esto sería:

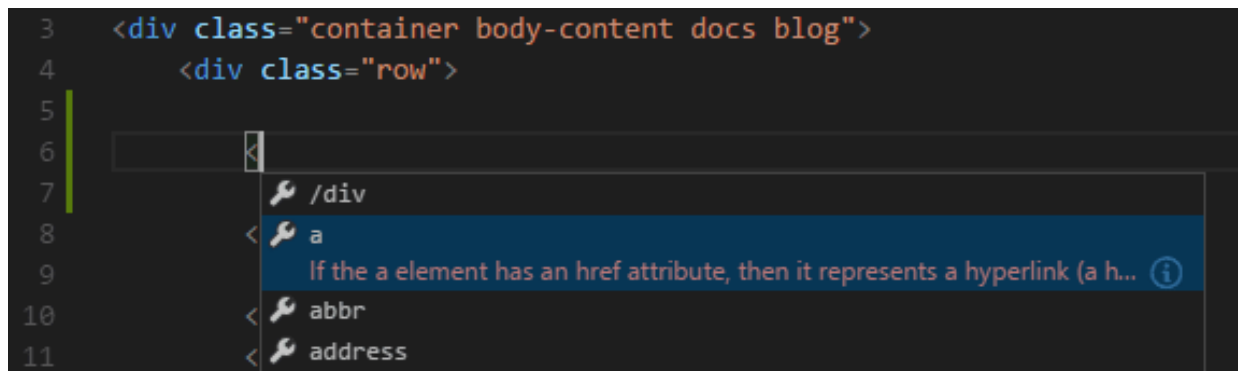
```
<video controls width="500" height="300" autoplay loop muted>
  <source src="video.mp4" type="video/mp4">
  <p>Tu navegador no soporta video</p>
</video>
```

Existen muchos atributos que se le pueden pasar a una etiqueta de video o audio, alguna de las más conocidas son:

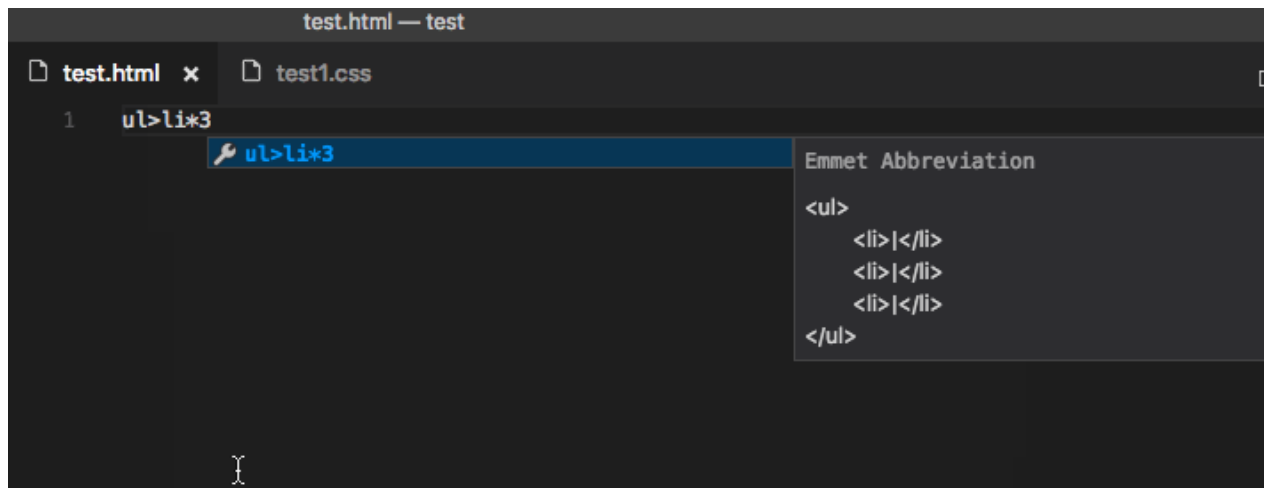
- **Width/Height:** Permite modificar el ancho y altura de video.
- **Autoplay:** Inicia automáticamente el video/audio una vez es cargado.
- **Loop:** Reproduce el video/audio desde el comienzo una vez que llega a su fin.
- **Muted:** Silencia el video/audio.

2. HTML y Visual Code

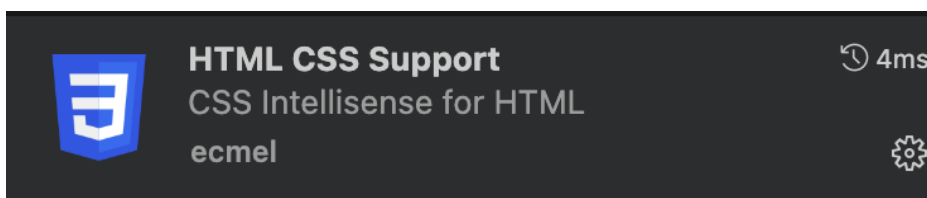
Visual Studio Code proporciona soporte básico para la programación HTML lista para usar. Hay resaltado de sintaxis, terminaciones inteligentes con IntelliSense y formato personalizable. Esta herramienta también incluye un excelente soporte para Emmet. Por ejemplo, a medida que se escribe HTML, ofrece sugerencias a través de HTML IntelliSense. En la imagen siguiente, puede ver un cierre de elemento HTML sugerido `</div>` así como una lista de elementos sugeridos específicos del contexto.



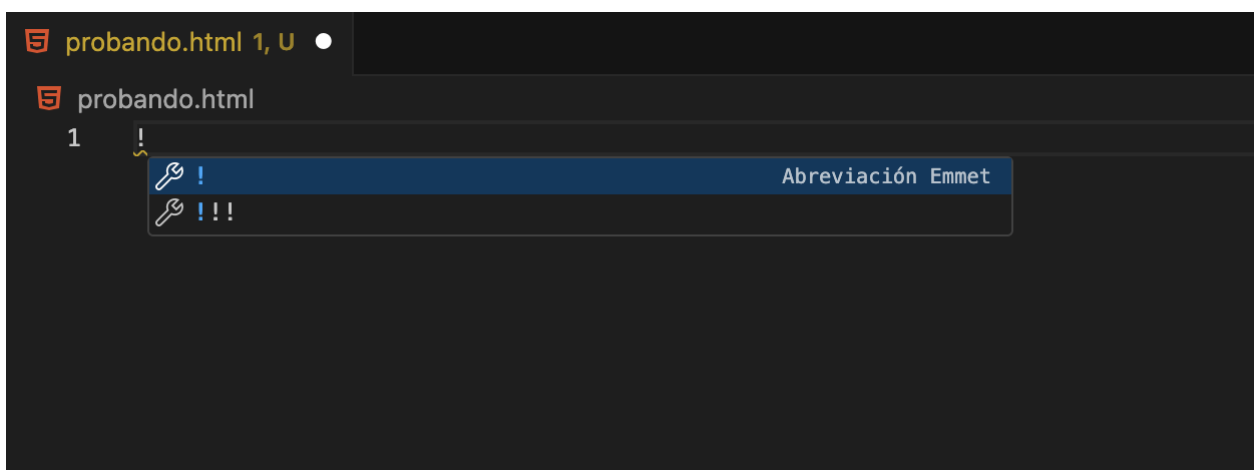
VS Code también admite la expansión de fragmentos de Emmet. Las abreviaturas de Emmet se enumeran junto con otras sugerencias y fragmentos en la lista de autocompletado del editor.



Además, se puede instalar una extensión para agregar más funciones. Por ejemplo la extensión HTML CSS support.



Una vez instalada, se puede crear la estructura de una página HTML escribiendo "!" y presionando *enter*.

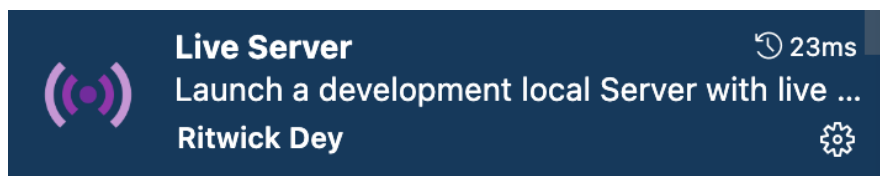


```

probando.html U
probando.html > html > head > meta
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9
10 </body>
11 </html>
    
```

Más información en: <https://code.visualstudio.com/docs/languages/html>

Con Visual Code también podemos ver nuestro código HTML corriendo desde un servidor web. Para esto solo hay que instalar la extensión Live Server.



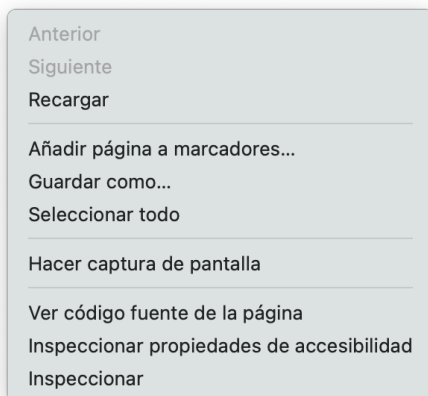
Luego, seleccionar la página HTML, seleccione la página web dar botón derecho del mouse y escoger "Open with Live Server" del menú (ver imagen).



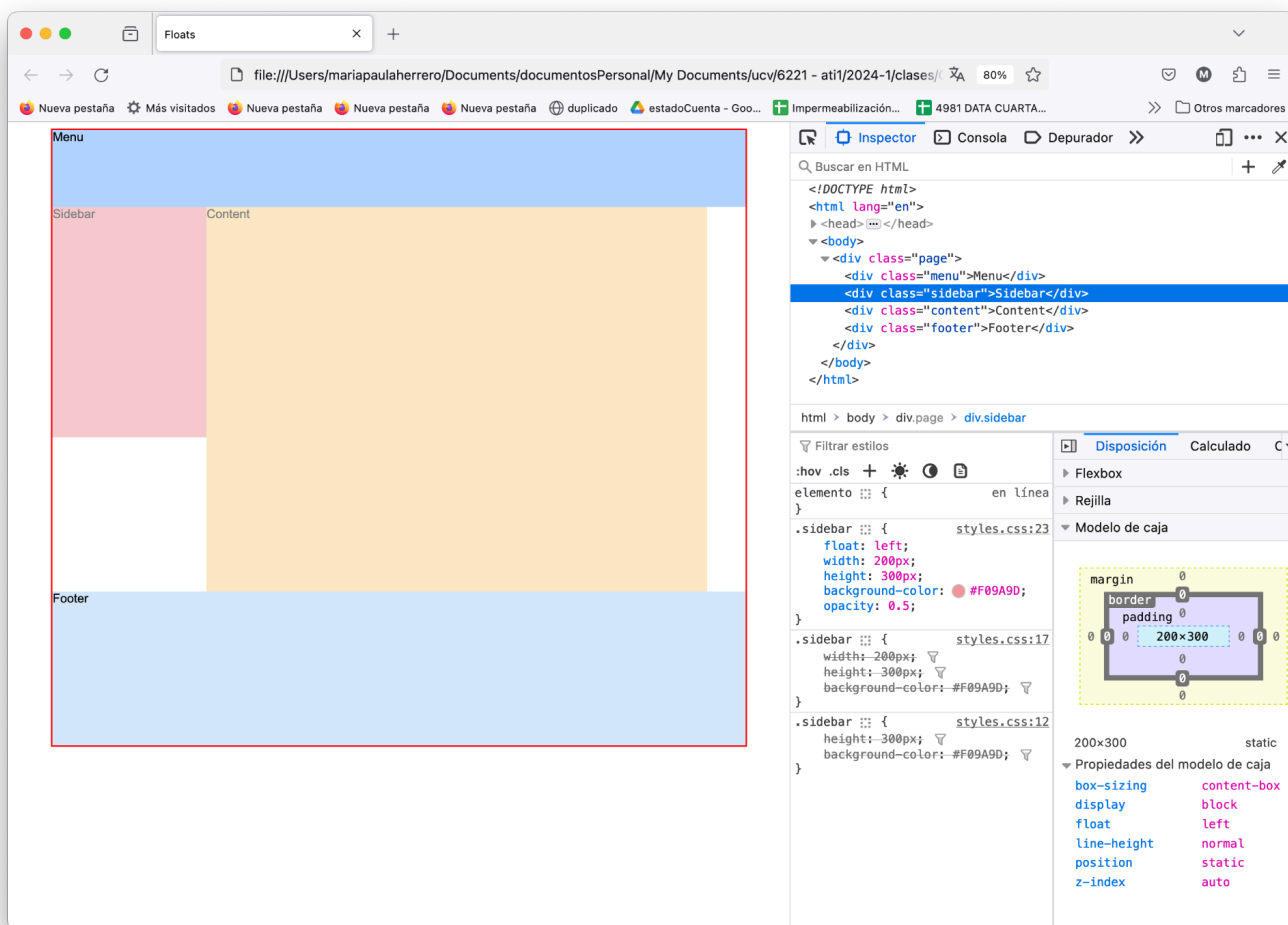
3. DOM

El DOM es una interfaz de programación para documentos HTML, XML y SVG. Provee una representación lógica estructurada del documento en la forma de un árbol y define la manera en la que éste puede ser accedido y manipulado.

Desde el navegador web podemos ver el DOM resultante usando las herramientas para desarrollador. Para verlo sólo hay que presionar el botón derecho del mouse sobre la página web y escoger inspeccionar en el menú que se muestra (ver imagen).



Esto abre la herramienta para desarrolladores en la opción de Inspector. En el inspector vemos el HTML de la página como si fuera un árbol de nodos. Desde esta herramienta podemos abrir y cerrar partes del árbol.. Además, cuando seleccionamos una etiqueta HTML podemos ver los estilos que tiene asociado y la información que maneja del modelo de caja. En la imagen podemos ver el ejemplo de la etiqueta `<div>`, cuya propiedad *display* es *block* y sus dimensiones son 200x300, con padding, border y margin en 0.



4. CSS

Las hojas de estilo proveen formas de asociar información de presentación con los elementos estructurales de un documento HTML en una forma que no corrompa su estructura subyacente. Ellas permiten describir la semántica de presentación (la vista y formato) de un documento escrito en un lenguaje de marcado.

Para incluir CSS podemos incrustar la etiqueta `<style>` en un documento HTML o incluir una hoja de estilos CSS externa, mediante una etiqueta llamada `<link>`. El elemento `<link>` debe ser incluido como hijo de la etiqueta head y debe incluir, al menos, dos atributos que permiten establecer la relación con el archivo CSS. Primero, el atributo href que le permite indicar la ruta al archivo css. Luego, rel, que permite indicar que es una hoja de estilos y si se desea se puede incluir el atributo type donde se indica explícitamente que el tipo del documento es CSS. La sintaxis finalmente es de la forma:

```
<link rel="stylesheet" type="text/css" href="ruta/al/archivo.css">
```

El asignar un valor a una cierta propiedad es la característica y función fundamental de CSS. Y el asignar esos pares propiedad/valor es lo que se llama en CSS una declaración. Es el elemento nuclear de definición de propiedades por lo cual se hace sumamente importante. La sintaxis de la declaración es simple, **propiedad:valor**, es importante observar los dos puntos que separan cada uno de estos elementos.

El bloque de declaración es el lugar donde se definen las propiedades aplicadas a un selector o grupo de selectores en una cierta regla. En el bloque de declaración se define a cuáles elementos del contenido se debe aplicar el (o los) estilos. El bloque de declaración se encuentra entre llaves y el mismo consiste de pares propiedad valor separados por punto y coma. Una sintaxis final de cómo se vería la sintaxis de una regla css sería:

```
selector {  
    propiedad: valor;  
    propiedad: valor;  
    ...  
}
```

Los selectores permiten al desarrollador indicar para cuáles elementos se deben aplicar un cierto grupo de reglas. Existen cuatro tipos de selectores, siendo el cuarto un derivado de los primeros tres:

- **Selector de id:** este selector permite aplicar sobre un elemento que cumpla con un identificador único, como es el atributo id de un elemento, el grupo de propiedades que se defina en la regla. La semántica de este identificador es:

```
#id {  
    <propiedades>  
}
```

La semántica del elemento para el id es: **<elemento id="id">**

- **Selector de clase:** el concepto de clase en CSS permite definir un estilo, y aplicar este estilo a múltiples elementos en el documento HTML. Para que un elemento pertenezca a una clase se define como un atributo del elemento. El selector de clase permite aplicar la definición del estilo a los elementos con atributo de clase equivalente. La semántica del identificador es:

```
.clase {  
    <propiedades>  
}
```

La semántica del elemento para la clase es: **<elemento class="clase">**

- **Selector de nombre:** este selector permite aplicar sobre un nombre de elemento el grupo de propiedades que se defina en la regla. Cuando hablamos de nombre de elementos esto refiere a las etiquetas, y las distintas clasificaciones que estas involucran. La semántica del identificador es:

```
nombre {  
    <propiedades>  
}
```

- **Selectores agrupados:** los selectores CSS pueden ser colocados de forma agrupada. Los selectores agrupados son simplemente listas de selectores a los cuales se aplican las propiedades indicadas por la regla. La semántica es semejante a las anteriormente definidas, con la única excepción que los selectores distintos son separados por medio de comas (,) resultando en la siguiente sintaxis:

```
selector_1, selector_2, ... {  
    <propiedades>  
}
```

- **Selectores anidados:** permiten definir una jerarquía de aplicación de las propiedades de la regla de forma vertical. Este aplica bajo el esquema que la regla es válida para aquellos elementos que se encuentren, en la estructura del árbol DOM, como hijos de cualquiera de los elementos que cumplan el atributo más externo o primero y que a su vez se encuentren como hijos de los elementos que cumplan el segundo atributo, y de esa manera con todos los atributos intermedios hasta que cumplan el atributo más interno o último atributo. La sintaxis para este tipo de selector es la siguiente:

```
selector_1 selector_2 {  
    <propiedades>  
}
```

Para comprender de manera más adecuada el funcionamiento de dicho selector se puede ilustrar mediante el siguiente ejemplo:

```
.error p {  
    ...  
}
```

En este caso todos aquellos párrafos hijos de los elementos con la clase "error" cumplirían esta regla.

- **Pseudo-clases:** estos elementos permiten realizar formateo en base a información que no se encuentra en el árbol del documento. Existen diversas pseudo-clases en CSS sin

embargo una de las más conocidas es :hover que permite especificar que vista css estará disponible si el elemento se encuentra siendo apuntado por el usuario. La sintaxis para el uso de pseudo clases es la siguiente:

```
selector:pseudo-clase {  
  <propiedades>  
}
```

- **Selector universal:** permite seleccionar cualquier elemento. La sintaxis para el uso de pseudo clases es la siguiente:

```
* {  
  <propiedades>  
}
```

Hay otros selectores más específicos, como los selectores de atributos con comparación a valores, otras pseudo-clases, pseudo-elementos, e inclusive negadores de selección. Pueden verificar los selectores disponibles en: <http://www.w3.org/TR/css3-selectors/>

Animación en CSS3

La animación en CSS3 consiste en cambiar gradualmente el estilo de un elemento por otro. Para hacer uso de la animación debes especificar los “Keyframes”, para esto se usa la regla de **@keyframes**.

Cuando especificas los estilos CSS dentro de una regla **@keyframes**, la animación cambiará gradualmente del estilo que posea en ese momento a un nuevo estilo, el tiempo de cambio entre un estilo y otro se puede controlar mediante el uso de la propiedad “animation-duration” la cual indica el tiempo total de la animación, y luego puedes distribuir el **@keyframes** en diferentes porcentajes basados en ese tiempo.

Importando CSS

Dentro de un archivo CSS se puede realizar la inclusión de las reglas de otro archivo CSS mediante el uso de **@import**, permitiendo que se puedan incluir reglas dentro de otro documento CSS. La sintaxis es simple:

```
@import “ruta/al/archivo.css”;
```

El **@import**, puede incluir luego de la ruta una lista de media queries que permiten identificar para cuál tipo de medio aplica esa importación.

5. Ejercicios

a) Hacer una página con HTML básico

1. Luego de instalar la extensión HTML CSS support abra Visual Code y escriba "!" y presionando **enter**.
2. Indique que el lenguaje de la página es en español.
3. Agregue `<title>Título de la Página</title>` dentro de `<head>`
4. Agregue dentro de `<body>`
`<h1>Mi primera cabecera</h1>`
`<p>Mi primer párrafo.</p>`
5. Agregue cinco nuevos elementos con las siguientes etiquetas: `<h2>`, `<a>`, ``, `<table>`, `` y `` a su documento HTML. Tres de ellos deben contener atributos correspondientes.
6. A continuación, copie el código de abajo y colóquelo en su documento HTML y comente lo que observa con el grupo.

```
<div style="color:#0000FF">  
  <h3>Lorem Ipsum</h3>  
  <p>Pellentesque habitant morbi tristique senectus et netus et turpis egestas.  
  Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante.  
  Donec eu libero sit amet quam egestas semper. Aenean ultricies mi vitae est.  
  Mauris placerat eleifend leo.</p>  
</div>
```

7. Copie el código de abajo en su documento HTML anterior y discuta las observaciones con el preparador.

```
<form>  
  <b>Nombre:</b> <br />  
  <input type="text" name="Nombre" value=""><br />  
  <b>Apellido: </b> <br />  
  <input type="text" name="Apellido" value=""><br /><br />  
  <input type="submit" value="Enviar">  
</form>
```

8. Copie el código de abajo en su documento HTML anterior y discuta las observaciones con el preparador.

```
<button type="button"  
onclick="document.getElementById('demo').innerHTML = Date()">  
  Presiona aquí para ver la fecha.  
</button>  
<div><p id="demo"></p></div>
```

9. Abra el DOM Inspector del navegador haciendo clic derecho sobre la página creada y escoja la opción inspeccionar. A continuación, observe el código a través del inspector e indique un camino jerárquico desde `<html>` a `<p id="demo">`
 10. Usando CSS cree un selector id en cuya propiedad se cambie el color de fondo a amarillo.
 11. Cree un selector de clase en cuya propiedad se cambie el color de fondo a azul.
 12. Cree un selector de nombre en cuya propiedad se cambie el color de fondo a rojo.
 13. Cree un selector anidado que cambie el color a un título secundario `<h2>` dentro de una división `<div>`. Puede usar cualquier tipo de selector visto anteriormente.
 14. Cree un selector de pseudo-clase que cambie el color de un párrafo `<p>` al pasarle por encima con el mouse
- b) Estudiar como se hace una animación de 4 segundos, que cambie gradualmente el color de fondo de un elemento `<div>` de rojo a amarillo, de amarillo a azul, de azul a verde y finalmente de verde a rojo. Copia y pega el siguiente código en un archivo html, ábrelo en un navegador y comenta lo que observas:

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      background-color: red;
      position: relative;
      -webkit-animation-name: ejemplo; /* Chrome, Safari, Opera */
      -webkit-animation-duration: 4s; /* Chrome, Safari, Opera */
      animation-name: ejemplo;
      animation-duration: 4s;
    }

    /* Chrome, Safari, Opera */
    @-webkit-keyframes ejemplo {
      0% {background-color:red; left:0px; top:0px;}
      25% {background-color:yellow; left:200px; top:0px;}
      50% {background-color:blue; left:200px; top:200px;}
      75% {background-color:green; left:0px; top:200px;}
      100% {background-color:red; left:0px; top:0px;}
    }
  </style>
</head>
<body>
  <div></div>
</body>
</html>
```

```
/* Standard syntax */
@keyframes ejemplo {
  0% {background-color:red; left:0px; top:0px;}
  25% {background-color:yellow; left:200px; top:0px;}
  50% {background-color:blue; left:200px; top:200px;}
  75% {background-color:green; left:0px; top:200px;}
  100% {background-color:red; left:0px; top:0px;}
}
</style>
</head>
<body>
  <div></div>
</body>
</html>
```

c) Hacer una animación. Para eso

1. Descomprime el archivo "lab2.zip"
2. Separar la hoja de estilo en dos (una para la cara y otra para los ojos). La hoja de estilo de la cara deberá importar la de los ojos, y la de la cara será importada por el documento HTML (el documento HTML no importará directamente la hoja de estilo de los ojos).
3. Crea el código CSS necesario para realizar la siguiente animación: comenzar mirando hacia el frente, luego mirar a la izquierda, mirar a la derecha, mirar hacia arriba, mirar hacia abajo y finalizar mirando hacia el frente. Repetir la animación infinitamente. Recuerda que la animación se debe sentir realista (evitar cambios bruscos). Ayúdate con la siguiente página para que conozcas más sobre las técnicas de animación en CSS3: http://www.w3schools.com/css/css3_animations.asp

d) Crear dos páginas web que muestren un directorio de estudiantes. Estas páginas serán perfil.html y index.html. Para esto siga los siguientes pasos:

1. Para crear la página de perfil, cree un directorio cuyo nombre sea su cédula de identidad y dentro cree el archivo perfil.html. Este archivo debe tener sus datos y se debe parecer a la siguiente imagen:



María Paula Herrero

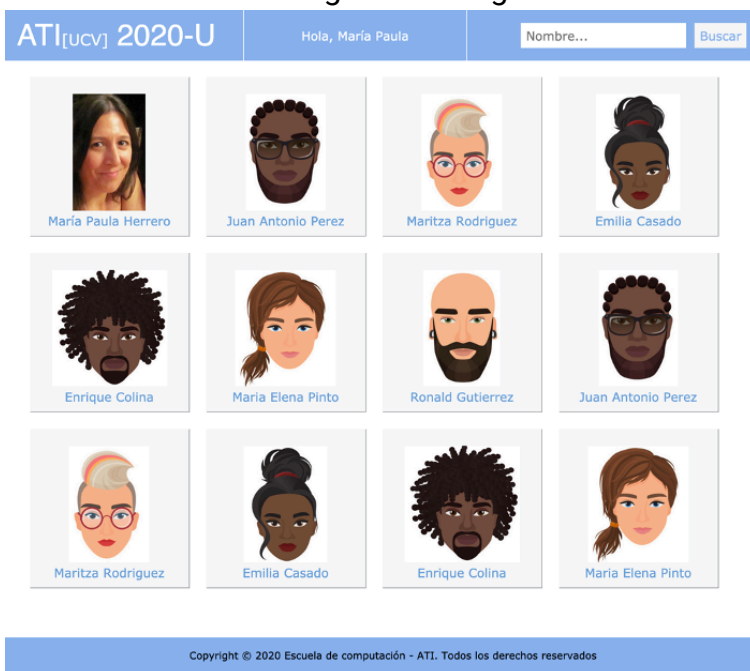
Soy licenciada en Computación de la UCV. Me desempeño en el área de ingeniería web y actualmente doy clases en la Escuela de Computación de la UCV. Además, desde el 2007 he venido desarrollando una segunda carrera como escritora e ilustradora. He participado en diversos talleres de escritura y actualmente estudio diseño gráfico en Prodiseno.

Mi color favorito es:	Azul
Mi libro favorito es:	El Señor de los Anillos
Mi estilo de música preferida:	Cualquiera menos regueton
Video juegos favoritos:	Pokémon Go
Lenguajes aprendidos:	c++, java, php, perl, python
Si necesitan comunicarse conmigo me pueden escribir a: mpaulaherrero@gmail.com	

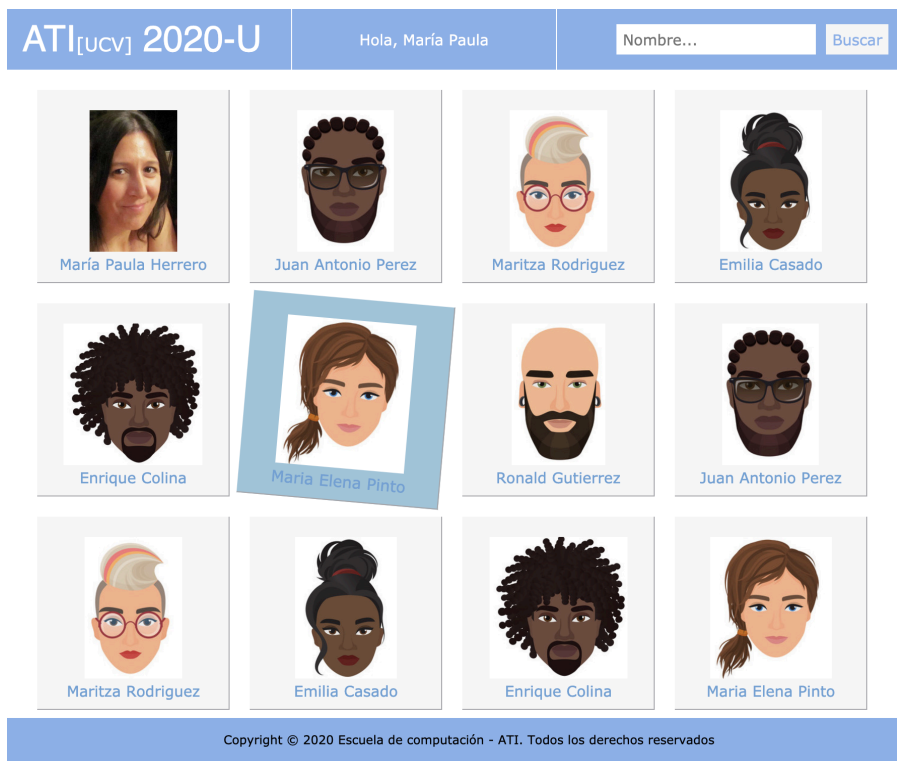
2. Para lograr esto tome en cuenta lo siguiente:

- Coloque dentro de la carpeta una foto suya identificada con el número de cédula y la extensión de la imagen (ejemplo, 12444133.jpg)
- Dentro de la etiqueta `<head>` agregue una etiqueta que indique una codificación que acepte caracteres especiales (es decir, que interprete los acentos, ñ, entre otros).
- Agregue una etiqueta para colocar el icono de la facultad como icono de la página (<http://www.ciens.ucv.ve/portalsig2/favicon.ico>)
- Agregue su nombre y apellido como el título de la página
- En el `<body>` coloque la foto al lado izquierdo
- Coloque al lado derecho los datos del perfil con la siguiente información: el nombre y el apellido juntos, en letra más grande y de otro color
- Abajo colocar la descripción de la persona, como un párrafo, con una fuente distinta a la de defecto, y en cursiva.
- Abajo colocar:
 - "Mi color favorito es: " con el valor del color favorito
 - "Mi libro favorito es: " con el valor del libro favorito
 - "Estilo de música preferida es: " con el valor del estilo de música
 - "Mis Video juegos favoritos son:" con el valor de los video juegos
 - "Lenguajes aprendidos:" con la lista de lenguajes de computación aprendidos o que maneja destacado en negritas

- Colocar "Si necesitan comunicarse conmigo me pueden escribir a:" y el correo electrónico con un enlace para que el navegador permita enviar un email.
3. Para el estilo tome en cuenta lo siguiente:
 - Agregue en la parte superior del archivo, dentro de la etiqueta `<head>`, la etiqueta `<style>` y definir los estilos para que la página web se vea como la imagen. Note que estamos usando el estilo dentro del HTML, no en un archivo externo. Asegurarse de usar los selectores correctos: selectores de etiqueta, creando un class o creando un id.
 - Revise que estén presentes en el diseño: la curva el borde superior izquierdo de la imagen, el borde y sombra a la caja del perfil y el color del enlace del email en "orange" y "red" luego de ser visitado.
 4. Haga commit y suba a GitHub todo lo que ha hecho hasta ahora con un mensaje que diga "primera versión página perfil"
 5. Dibuje el DOM resultante de perfil.html, lo puede hacer a mano o usando una herramienta de dibujo. Guarde la imagen en el proyecto en un directorio llamado docs que esté alojado en el directorio raíz. Luego haga commit y suba a GitHub la imagen con un mensaje que diga "árbol DOM de perfil"
 6. Para crear el index.html, descomprima el archivo dummies.zip y coloque su contenido en el directorio raíz de su proyecto. Luego, cree en el directorio raíz el archivo index.html y haga un html que se vea como en la siguiente imagen:



7. Para esto tome en cuenta que:
 - La etiqueta `<head>` debe tener la información para lograr una codificación que acepte caracteres especiales (es decir, que interprete los acentos, ñ, entre otros), el icono de la facultad como icono de la página y el título de la página como "ATI[UCV] 2024-1".
 - Todo lo que tenga que ver con estilo colóquelo dentro de una etiqueta `<style>` dentro de `<head>`. Use los selectores que crea convenientes. Incluya aquí todo el estilo que necesite para lograr que el listado de estudiantes se vea como en la imagen.
 - La estructura de la página HTML debe usar las etiquetas `<header>`, `<section>` y `<footer>`, tanto `<header>` como `<footer>` deben estar separadas gráficamente de `<section>` por una línea.
 - Dentro de `<header>` debe estar el `<nav>`, el cual debe tener 3 opciones (usar `` y `` para cada una), en la primera parte debe estar el texto "ATI[UCV] 2024-1" y lograr que "[UCV]" tenga un tamaño más pequeño. El texto "Hola, [Tu nombre]", coloque su nombre ahí. Por último, debe haber un formulario para hacer búsqueda
 - Dentro de `<section>` hacer un listado dummy con ciertas personas inventadas, usando una foto pequeña y un nombre cualquiera (usar `` y `` para cada una), para la primera opción use la imagen que está en el directorio con su CI y su nombre
 - Dentro de `<footer>` colocar el texto "Copyright © 2024 Escuela de computación - ATI. Todos los derechos reservados"
 -
8. Al tener todo listo, haga commit del proyecto con el mensaje "primera versión página de listado"
9. Ahora vamos a sacar el estilo incrustado en el HTML a un archivo a parte llamado style.css y guárdelo en el directorio css. Este archivo debe ser compartido tanto por perfil.html como por index.html. Al terminar haga commit del proyecto con el mensaje "hojas de estilo compartidas"
10. Agregue una animación CSS sobre cada opción de la lista (sobre cada estudiante), que incluya un efecto de zoom y de rotación sobre la imagen y texto de los alumnos cada vez que se pase por encima el ratón y cambie también el color de fondo a #99c5dd. El efecto debe quedar como se muestra en la siguiente imagen.



11. Al tener todo listo, haga commit con el mensaje “animación sobre los alumnos”

Al terminar, colocar en el reto del Aula Virtual el url del repositorio de GitHub que uso para hacer este ejercicio d y una imagen del historial de Git Graph que muestra su Visual Code.