



Universidad Don Bosco
Facultad de Ingeniería
Escuela de Computación

Ciclo I - 2025

Guía de laboratorio N° 10

Nombre de la práctica: Maquetación de páginas web.

Lugar de ejecución: Laboratorio UDB.

Materia: Lenguajes de Marcado y Estilos Web.

I. Objetivos.

- ✓ Utilizar las propiedades de CSS3 para crear diseños multi-columna.
- ✓ Desarrollar páginas web con diseños fijos.
- ✓ Desarrollar páginas web con diseños fluidos.

II. Introducción Teórica.

DISEÑOS MULTI-COLUMNA CON CSS3

Propiedad column-count

Una de las novedades de CSS3 es permitir disponer un bloque de texto en múltiples columnas con la única indicación de la cantidad de columnas en que queremos que lo divida.

La propiedad column-count nos permite dividir el texto en varias columnas. El valor de esta propiedad es un número, siendo este número el total de columnas a utilizar y automáticamente el navegador se encargará de agrupar el texto en las columnas indicadas.

```
selector {  
    column-count: cantidad de columnas;  
}
```

Para utilizar esta propiedad es preciso usar los prefijos propietarios para garantizar la compatibilidad con todos los navegadores y sus versiones.

Propiedad column-width

Con la propiedad column-width también generamos múltiples columnas. Tiene sentido emplearla cuando nuestro diseño es flexible, es decir que si modificamos el tamaño de la ventana del navegador su contenido se relocaliza.

Con la propiedad column-width especificamos el ancho de la columna en píxeles, porcentaje, etc. y luego se generan tantas columnas como entren en el contenedor.

La sintaxis es la siguiente:

```
Elemento {  
    column-width: ancho en pixeles o porcentajes;  
}
```

Propiedad column-gap

La propiedad column-gap nos permite especificar la separación entre las columnas. Su sintaxis es:

```
Selector {  
    column-gap: valor;  
}
```

Podemos utilizar cualquier unidad para indicar la separación entre columnas (px, em, etc.)

Propiedad column-rule

La propiedad column-rule nos permite definir y configurar una línea de separación entre columnas.

```
Selector {  
    column-rule: grosor estilo color;  
}
```

Nótese la similitud de la propiedad column-rule con la propiedad border (en su forma corta o “shorthand”) que estudiamos en el tema “modelo de caja”.

DISEÑOS WEB FIJOS Y FLUIDOS

Diseños web fijos

Un diseño web fijo es aquel que permanece inalterable sea cual sea el dispositivo donde se visualiza el sitio web. Es indiferente al tamaño de la pantalla y a la anchura del navegador web que la visualiza.

Sean cuales sean las características del dispositivo, pantalla o navegador, siempre se visualiza el mismo tamaño preestablecido por el diseñador web en píxeles.

Las características del diseño web fijo son las siguientes:

- ✓ *Tamaños fijos medidos en píxeles.*
- ✓ *No reaccionan al tamaño de la pantalla.*
- ✓ *Orientado a resoluciones de escritorio.*
- ✓ *No están optimizados para dispositivos móviles.*
- ✓ *Fáciles de crear.*
- ✓ *En resoluciones muy amplias el espacio no es aprovechado, mientras que en resoluciones pequeñas aparecerá una barra de desplazamiento horizontal, lo cual afecta la usabilidad del sitio*

Diseños web fluidos

Los diseños web fluidos (también conocidos como “diseños líquidos”) fueron creados para no tener un tamaño definido, y permitir que los monitores pueden condicionar el tamaño de salida.

Usando porcentajes (%) podremos definir el tamaño de los elementos de nuestra página web para definir un diseño fluido que se adapte a cualquier monitor.

Las características de los diseños web fluidos son las siguientes:

- ✓ *Diseños web adaptables al tamaño del navegador o “viewport”.*
- ✓ *Se construyen definiendo los anchos de los elementos con porcentajes.*
- ✓ *Requiere más planeación y verificación.*
- ✓ *Efectos negativos en pantallas muy pequeñas.*

ALTURAS/ANCHURAS MÁXIMAS Y MÍNIMAS

Cuando se diseña la estructura de una página web, se debe tomar la decisión de optar por un diseño de anchura fija o un diseño cuya anchura se adapta a la anchura de la ventana del navegador.

Sin embargo, la mayoría de las veces sería conveniente una solución intermedia: que la anchura de la página sea variable y se adapte a la anchura de la ventana del navegador, pero respetando ciertos límites. En otras palabras, que la anchura de la página no sea tan pequeña como para que no se puedan mostrar correctamente los contenidos y tampoco sea tan ancha como para que las líneas de texto no puedan leerse cómodamente.

CSS define cuatro propiedades que permiten limitar la anchura y altura mínima y máxima de cualquier elemento de la página. Las propiedades son max-width, min-width, max-height y min-height.

max-width	Anchura máxima
Valores	<medida> <porcentaje> none inherit
Se aplica a	Todos los elementos salvo filas y grupos de filas de tablas
Valor inicial	none
Descripción	Permite definir la anchura máxima de un elemento

min-width	Anchura mínima
Valores	<medida> <porcentaje> inherit
Se aplica a	Todos los elementos salvo filas y grupos de filas de tablas
Valor inicial	0
Descripción	Permite definir la anchura mínima de un elemento

max-height	Altura máxima
Valores	<medida> <porcentaje> none inherit
Se aplica a	Todos los elementos salvo columnas y grupos de columnas de tablas
Valor inicial	none
Descripción	Permite definir la altura máxima de un elemento

min-height	Altura mínima
Valores	<medida> <porcentaje> inherit
Se aplica a	Todos los elementos salvo columnas y grupos de columnas de tablas
Valor inicial	0
Descripción	Permite definir la altura mínima de un elemento

Las propiedades que definen la altura y anchura máxima y mínima se pueden aplicar a cualquier elemento, aunque solamente suelen utilizarse para estructurar la página. En general, las propiedades más utilizadas son **max-width** y **min-width**, ya que no es muy habitual definir alturas máximas y mínimas.

De esta forma, para conseguir un diseño de anchura variable pero controlada, se podrían utilizar reglas CSS como la siguiente:

```
#contenedor {
  width: 80%;
  min-width: 500px;
  max-width: 900px;
  margin: 0 auto;
}
```

Con esto estamos consiguiendo que el ancho del elemento con id “contenedor” sea del 80% pero que siempre se mantenga entre los límites de 900px y 500px.

III. Materiales y equipo.

No.	Cantidad	Descripción
1	1	PC con editores de textos Web: Sublime Text, Brackets, Notepad ++, y Bloc de nota y cualquiera de los navegadores siguiente en su última versión: Firefox, Google Chrome, Opera, Safari y Microsoft Edge.
2	1	Guía de Laboratorio #10 de Lenguajes de Marcado y Estilo Web

INDICACIONES GENERALES

1. Crear una carpeta en “mis documentos” y colocarle el nombre **N°Carnet_Guia10**, Nota: Cambiar N°Carnet por su número de carnet brindado por la Universidad.
2. Descargar los recursos de la guía, descomprimirlos y colocar todas las carpetas y archivos dentro de su carpeta de trabajo.

Ejercicio 1. Construyendo un estilo multicolumna estilo Pinterest

1. Creé un archivo HTML llamado “Multicolumna.html”.
2. Guardar el archivo en su carpeta de trabajo.
3. Escribir la estructura básica de un documento HTML en el archivo creado en el paso anterior.
4. Colocar como título de la página el texto “Múltiples columnas estilo Pinterest”.
5. Digitar el siguiente código HTML dentro del cuerpo (body) de la página web.

```
<div id="contenedor">
  <div id="columnas">
    <div class="pin">
      
      <h2>Cars</h2>
      <p></p>
    </div>

    <div class="pin">
      
      <h2>Monster Inc.</h2>
      <p></p>
    </div>

    <div class="pin">
      
      <h2>Ratatouille</h2>
      <p></p>
    </div>

    <div class="pin">
      
      <h2>Toy Story</h2>
      <p></p>
    </div>

    <div class="pin">
      
      <h2>Buscando a Nemo</h2>
      <p></p>
    </div>
  </div>
</div>
```

```

<div class="pin">
  
  <h2>Frozen</h2>
  <p></p>
</div>

<div class="pin">
  
  <h2>Intensamente</h2>
  <p></p>
</div>

<div class="pin">
  
  <h2>Zootopia</h2>
  <p></p>
</div>
<div class="pin">
  
  <h2>Ralph el demoledor</h2>
  <p></p>
</div>

```

```

<div class="pin">
  
  <h2>Shrek</h2>
  <p></p>
</div>

<div class="pin">
  
  <h2>Hotel Transylvania</h2>
  <p></p>
</div>

<div class="pin">
  
  <h2>Valiente</h2>
  <p></p>
</div>
</div>
</div>

```

6. Dentro de los recursos de la guía encontrará un archivo denominado "texto_ejemplo1.docx", ese archivo contiene todo el texto que deberá colocar dentro la página. Coloque el texto correspondiente en cada uno de los párrafos de los artículos.

7. Creé una carpeta denominada css. Dentro de esa carpeta, cree una hoja de estilo llamada pinterest.css y vincúlela a su página web.
8. Digite las siguientes reglas de estilo dentro del archivo creado en el paso anterior.

```
body {  
    background-color: #EEEEEE;  
}  
  
#contenedor {  
    width: 90%;  
    max-width: 1100px;  
    min-width: 700px;  
    margin: 50px auto;  
}
```

Note que con la regla de estilo anterior estamos diciendo que el elemento con id “contenedor” va a tener un ancho del 90% del viewport, pero siempre se mantendrá entre los límites de 1100px y 700px. Es decir, va a “fluir” mientras se encuentre entre esos valores, pero nunca será más grande que 1100px ni más pequeño que 700px.

9. A continuación conseguiremos que todo el contenido dentro del div con id “columnas” se muestre en 4 columnas mediante la siguiente regla de estilo:

```
#columnas {  
    column-count: 4;  
    column-gap: 10px;  
}
```

10. Agregue los prefijos propietarios en la regla de estilo anterior para garantizar la compatibilidad con los distintos navegadores y sus versiones.
11. Visualice los resultados en su navegador y notará que el contenido se presenta en 4 columnas pero el resultado no es aún muy estético.
12. Digite las siguientes regla de estilo:

```
.pin {  
    display: inline-block;  
    background: #FEFEFE;  
    border: 2px solid #FAFAFA;  
    box-shadow: 0 1px 2px rgba(34, 25, 25, 0.4);  
    margin: 0 2px 15px;  
    -webkit-column-break-inside: avoid;  
    -moz-column-break-inside: avoid;  
    column-break-inside: avoid;  
    padding: 15px;  
    padding-bottom: 5px;  
}
```

```

.pin p {
  font: 0.8em Arial, sans-serif;
  color: #333;
  margin: 0;
  text-align: justify;
  color: #424242;
}

.pin h2{
  font-size: 1em;
  margin: 0.5em 0;
  color: #212121;
}

```

13. Ahora conseguiremos que la imagen llene el ancho del div, para ello le daremos un ancho del 100%. Además haremos que la imagen se muestre con una opacidad del 70%.

```

.pin img {
  width: 100%;
  border-bottom: 1px solid #ccc;
  padding-bottom: 15px;
  margin-bottom: 5px;
  opacity: 0.7;
}

```

14. Visualice los resultados en su navegador, debería obtener un resultado similar al siguiente:



15. Ahora conseguiremos que la opacidad de las imágenes sea del 100% cuando se pasa el cursor sobre ellas.

```

.pin img:hover{
  opacity: 1;
}

```

16. Mediante una transición de 0.5s consiga que el cambio de la opacidad de las imágenes de haga de forma suave y progresiva.
17. Visualice los resultados en su navegador. Prueba cambiar el ancho del viewport de su navegador para que vea como el diseño “fluye” pero se mantiene siempre entre 1100px y 700px.

Ejercicio 2. Construyendo un diseño fijo

1. Creé un archivo HTML llamado “index.html”.
2. Guardar el archivo en su carpeta de trabajo.
3. Escribir la estructura básica de un documento HTML en el archivo creado en el paso anterior.
4. Colocar como título de la página el texto “Viajeros, agencia de viajes”.
5. Digitar el siguiente código HTML dentro del cuerpo (body) de la página web.

```
<section id="superior">
  <div id="encabezado">
    <nav>
    </nav>
    <div id="imgDesc">
    </div>
  </div>
</section>

<section id="contenido">

</section>
<footer>
  Copyright Lenguajes de marcado y estilo web 2016
</footer>
```

6. Coloque el siguiente código HTML dentro del nav de la página.

```

<ul>
  <li><a href="index.html" id="inicio">Inicio</a></li>
  <li><a href="planes.html" id="planes">Planes</a></li>
  <li><a href="galeria.html" id="galeria">Galeria</a></li>
</ul>
```

7. Coloque el siguiente código dentro del div con id “imgDesc”.

```

<div id="descripcion">
  <p>Viajeros es la mejor agencia de viajes de El Salvador. Te ofrecemos cobertura total
a todos los lugares turísticos del país. Tenemos planes turísticos que se adaptan a tu
bolsillo, tiempo y gustos. Deja de perder el tiempo, y confía tu diversión a los
expertos</p>
</div>
```

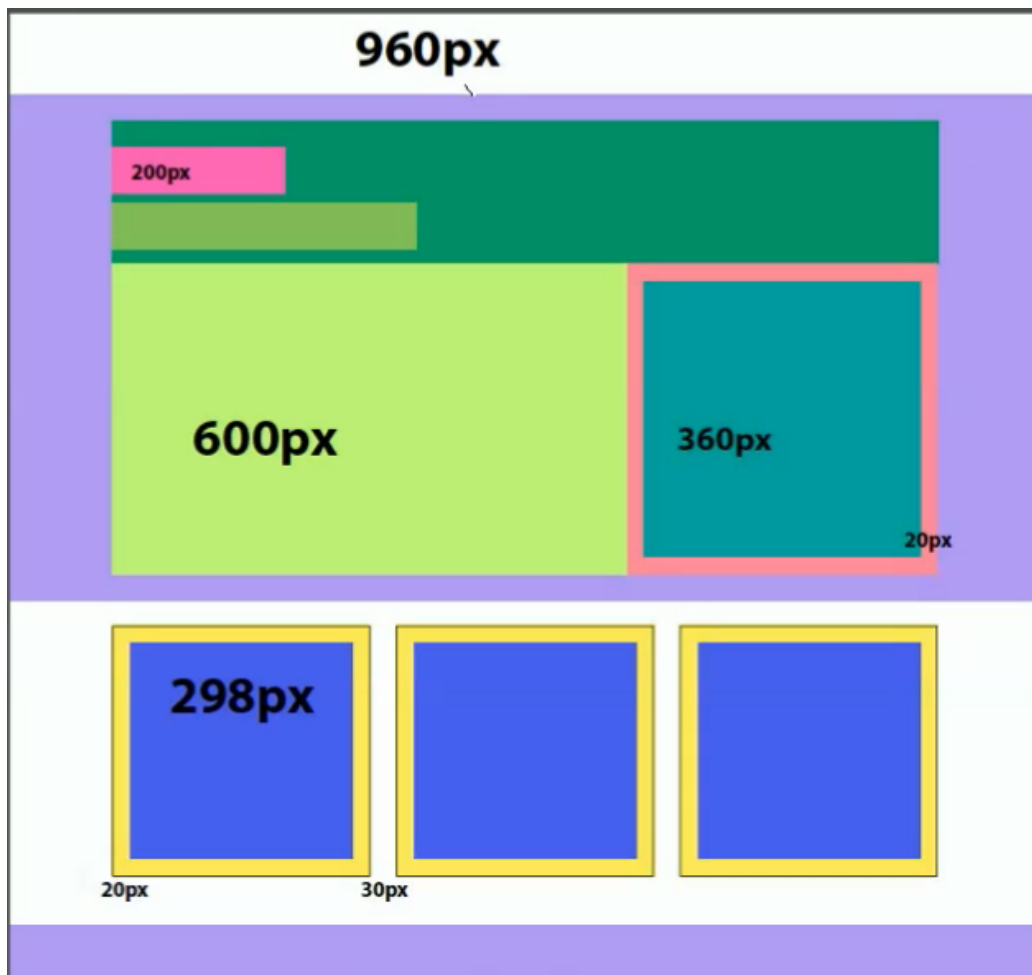

8. Coloque el siguiente código dentro de la sección con id “**contenido**”.

```
<div class="plan">
  
  <h1>Plan naturaleza</h1>
  Si te gusta sentir un contacto íntimo con la naturaleza, este plan es para ti. El plan
  incluye la visita a playas y montañas y caminatas en los lugares más paradisíacos del país.
  <p class="boton"><a href="planes.html#galeriaNaturaleza">Más información</a></p>
</div>
<div class="plan">
  
  <h1>Plan histórico</h1>
  Si lo tuyo es el turismo histórico, este plan te ofrece un recorrido por las distintas
  ruinas del país, así como una visita a los principales museos de guerra de El Salvador.
  <p class="boton"><a href="planes.html#galeriaHistorico">Más información</a></p>
</div>
<div class="plan">
  
  <h1>Plan Relax</h1>
  Si quieres recuperarte del estrés de la ciudad este plan podría interesarte. Incluye una
  visita por los lugares más paradisíacos y tranquilos del país.
  <p class="boton"><a href="planes.html#galeriaRelax">Más información</a></p>
</div>
```

9. Crear una nueva hoja de estilo, y guardarla dentro de su carpeta css con el nombre “**fijo.css**”.

10. Vincular su página web con la hoja de estilo creada en el paso anterior.

Nuestro objetivo será aplicar las distintas propiedades CSS estudiadas en el transcurso de la asignatura para crear un layout como el siguiente:



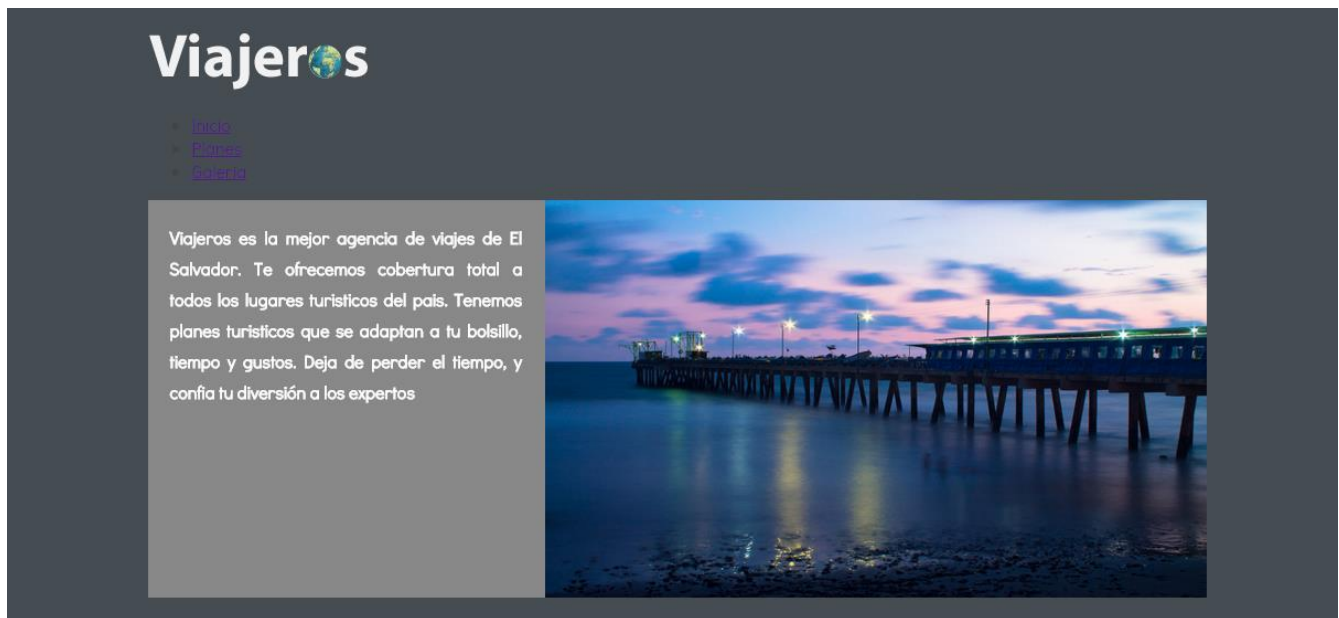
11. Coloque las siguientes reglas de estilo iniciales dentro de su archivo fijo.css

```
/* FUENTES */  
@import url(http://fonts.googleapis.com/css?family=Didact+Gothic);  
  
*{  
    font-family: 'Didact Gothic', sans-serif;  
    margin: 0;  
    padding: 0;  
}  
  
/*Estilos iniciales*/  
body{  
    margin: 0px;  
    background-color: #BBB;  
    color: #444;  
}  
  
h1{  
    font-size: 1.3em;  
    text-align: center;  
    color: #464d52;  
}
```

12. A continuación procedemos a estilizar toda la sección con id “superior”.

```
/* PARTE SUPERIOR */  
#superior{  
    width: 100%;  
    padding: 30px 0px;  
    background-color: #464d52;  
}  
  
#encabezado{  
    margin: 0px auto;  
    width: 960px;  
}  
  
#descripcion{  
    width: 360px;  
    height: 360px;  
    background-color: #888;  
    float: left;  
}  
  
#descripcion p{  
    margin: 20px;  
    color: white;  
    font-weight: bold;  
    line-height: 180%;  
    text-align: justify;  
}
```

13. Hasta el momento su encabezado debería lucir de la siguiente manera:



14. Ahora procederemos a darle estilo al **menú de navegación**, para ello crearemos dentro de la carpeta css una hoja de estilo llamada “**menú.css**”, la vincularemos con nuestra página web y escribiremos las siguientes reglas de estilo:

```
/*  MENU  */

nav{
    width: 960px;
    font-size: 1.4em;
}

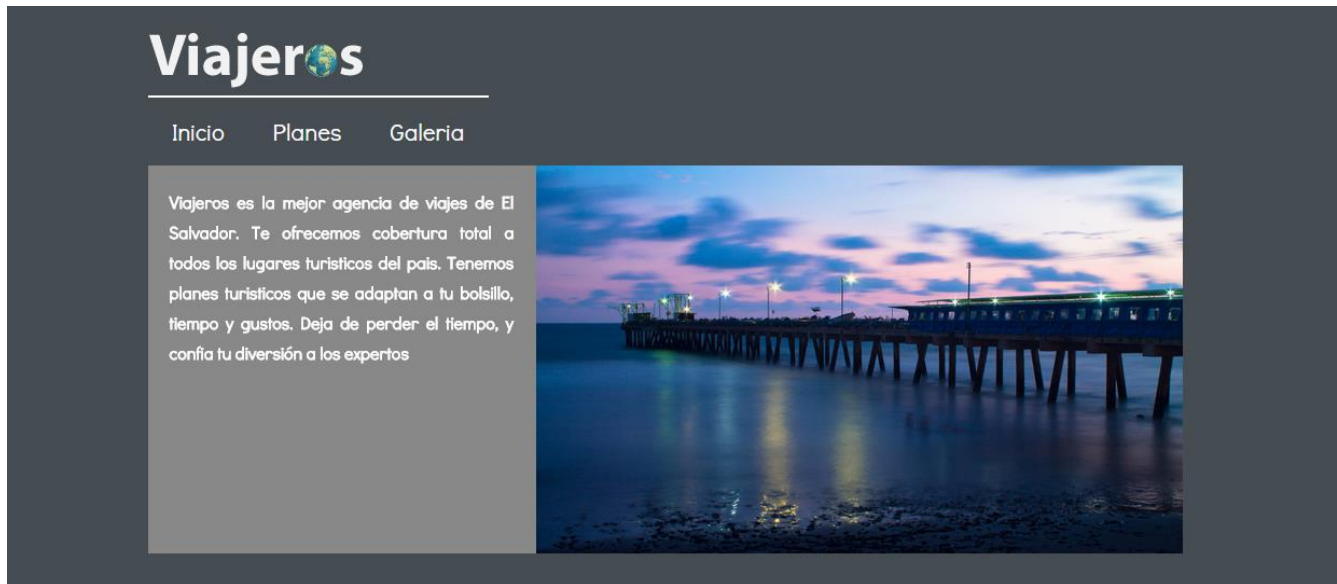
nav ul{
    list-style: none;
    margin: 0px 0px 65px 0px;
}

nav ul li{
    float: left;
    margin: 0px;
    padding: 10px 0px 0px 0px;
    border-top: solid 2px white;
}

nav ul li a {
    text-decoration: none;
    padding: 0.3em 0.5em;
    margin: 0em 0.5em;
    display: block;
    color: white;
}

}
```

15. Hasta el momento su encabezado debería lucir de la siguiente manera:



16. Ahora conseguiremos que los vínculos del menú cambien de color y escala cuando el usuario pase el cursor sobre ellos.

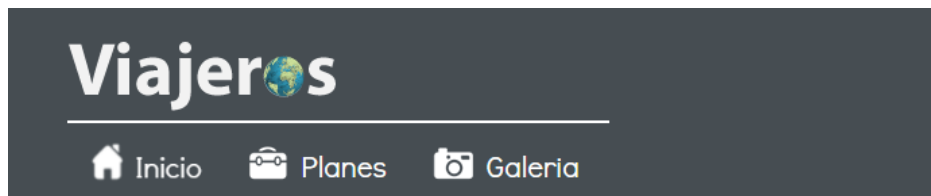
```
nav ul li a:hover{  
  color: #AAA;  
  transform:scale(1.25);  
}
```

17. Mediante transiciones consiga que tanto el cambio de color como el cambio de escala se hagan de forma suave y progresiva. La transición del color debe durar 1 segundo y la transición de la escala debe durar 3 segundos.

18. Finalmente agregaremos unos iconos en cada uno de los ítems del menú para mejorar la presentación del mismo.

```
nav ul li a#inicio{  
  background: url(../img/turismo/home.png) no-repeat 0px 3px;  
  padding-left: 40px;  
}  
nav ul li a#planes{  
  background: url(../img/turismo/planes.png) no-repeat 0px 3px;  
  padding-left: 40px;  
}  
nav ul li a#galeria{  
  background: url(../img/turismo/galeria.png) no-repeat 0px 3px;  
  padding-left: 40px;  
}
```

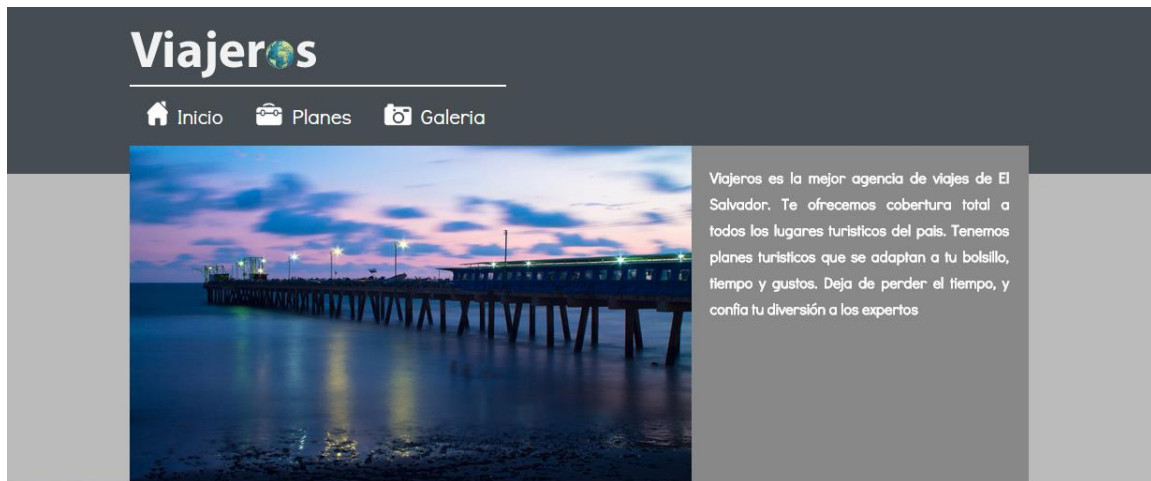
19. Si siguió los pasos al pie de la letra las indicaciones, su menú debería lucir de la siguiente manera:



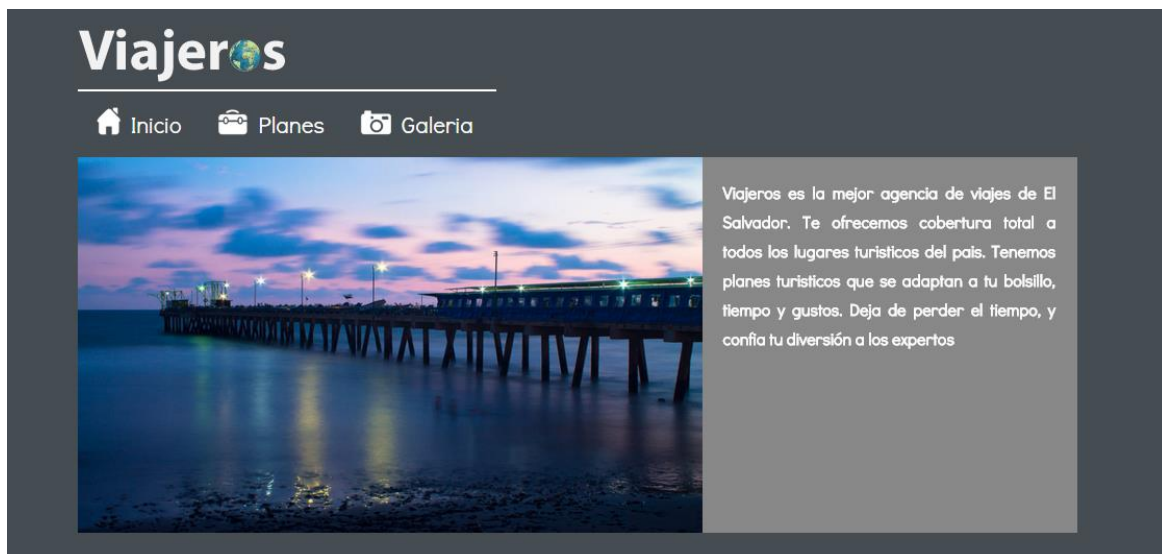
20. Con esto hemos finalizado el menú de navegación. **Cierre el archivo “menú.css” puesto que las siguientes reglas de estilo deberá colocarlas dentro del archivo “fijo.css”.**

21. Ahora finalizaremos el estilo de la sección con id “superior”. Consiga que la imagen que se encuentra dentro del div con id imgDesc flote a la izquierda (estas reglas de estilo deben colocarse en el archivo “fijo.css”)

El resultado obtenido debería ser similar al siguiente:



22. Note que la imagen está flotando a la izquierda, pero parece que se ha salido del contenedor. Este problema tan habitual cuando se usa la propiedad float ya lo hemos estudiado y resuelto en guías pasadas. La solución consiste en aplicar la regla “**overflow: hidden**” sobre el contenedor del elemento flotado (en este caso el contenedor sería el div con id “imgDesc”) para forzar que el contenedor “envuelva” a los elemento flotantes. Resuelva el error y visualice los resultados en su navegador. Debería obtener un resultado similar al siguiente:



23. Ahora procederemos a aplicar los estilos básicos para la sección con id “**principal**” y para el **footer** de la página.

```
/*Contenido principal de las paginas*/
#contenido{
    width: 960px;
    margin: 0px auto;
    clear: left;
}

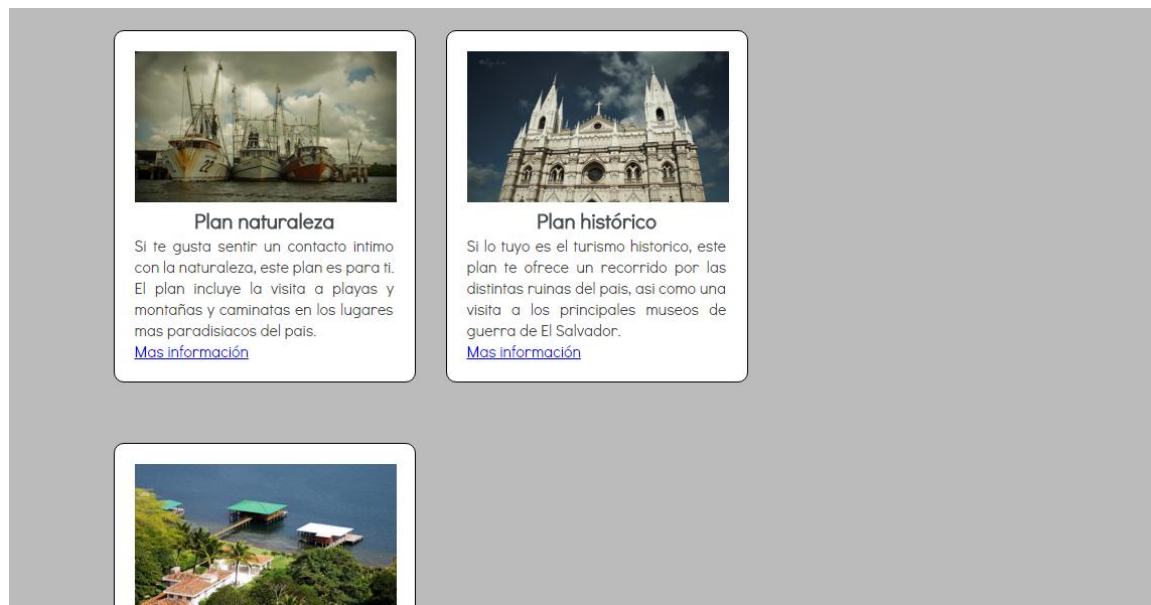
/* PIE DE LA PAGINA*/
footer{
    text-align: center;
    margin: 0;
    font-size: .9em;
    line-height: 2;
    clear: both;
    background-color: #464d52;
    color: white;
    width: 100%;
}
```

24. Con estos estilos ya tendríamos lista la plantilla que vamos a ocupar en todas las páginas de nuestro sitio web.
25. Ahora escribiremos los estilos particulares para la página index.html. En esta página tenemos algunos div con la clase “plan” los cuales queremos que se muestren en tres columnas.

```
/* CONTENIDO DE LA PAGINA INDEX */

.plan{
    width: 258px;
    padding: 20px;
    background-color:white;
    margin: 30px 15px;
    border: solid 1px black;
    border-radius: 10px;
    float: left;
    text-align: justify;
}
```

Visualice los resultados en su navegador y notará que aunque los anchos, márgenes y rellenos están colocados tal cual está en el layout presentado al inicio del ejercicio, la tercera columna no cabe y se va hacia abajo.



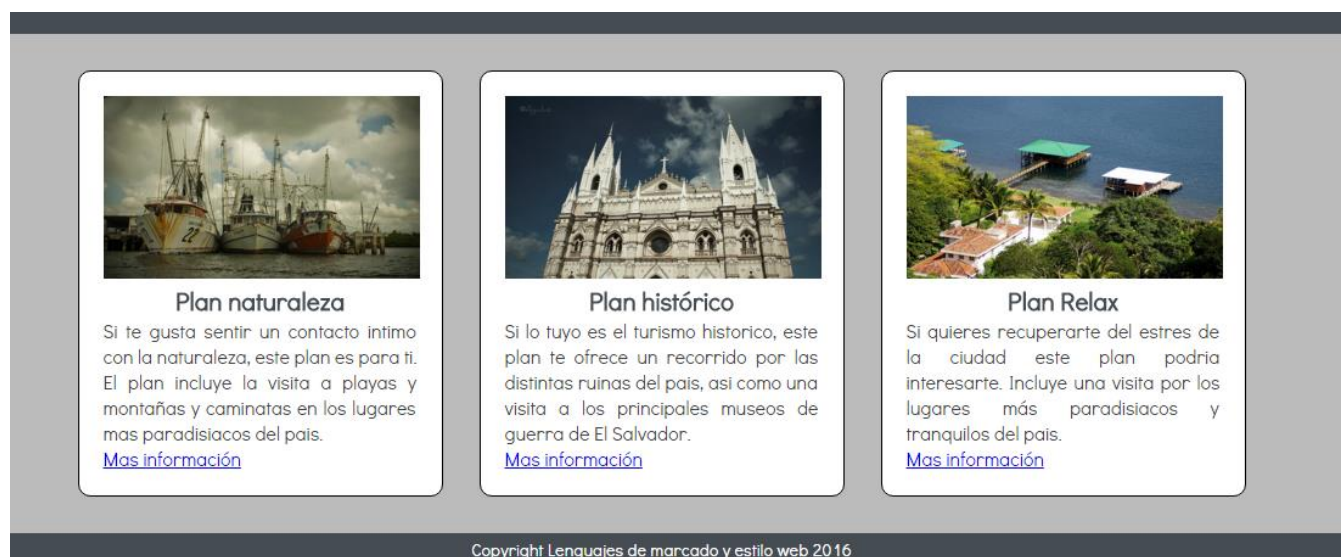
El problema se debe a que el primer div tiene un margen a la izquierda el cual no debería estar. Lo mismo sucede con el tercer div que tiene un margen innecesario a la derecha.

26. Elimine estos márgenes innecesarios mediante las siguientes reglas de estilo:

```
/* Primer div, cuarto div, .... */
#contenido div:nth-child(3n+1){
    margin-left: 0px;
}

/* Tercer div, sexto div, etc */
#contenido div:nth-child(3n){
    margin-right: 0px;
}
```

El resultado obtenido debería ser similar al siguiente:



27. Ahora haremos que los enlaces que los enlaces que tienen la clase botón, tengan dicha apariencia. Para ello, digite las siguientes reglas de estilo.

```
.boton{
  background-color:#464d52;
  padding: 7px;
  border-radius: 5px;
  color: white;
  margin-top: 10px;
  font-size: .9em;
  text-align: center;
  vertical-align: middle;
  cursor: pointer;
  transition: background 1s;
}

.boton:hover{
  background: #5077c7;
}

.boton a{
  text-decoration: none;
  color: white;
  width: 100%;
  display: block;
}
```

Página planes.html

Para construir las restantes páginas del sitio, la pagina index.html nos servirá como plantilla.

28. Cree una copia de su archivo index.html y guárdela con el nombre planes.html.
29. Dentro del archivo **planes.html** elimine todo el contenido de la sección con id “contenido” y reemplácelo por el siguiente código HTML.

```
<div class="planes" id="galeriaNaturaleza">
  <h1>Plan naturaleza</h1>
  
  <p>Si te gusta sentir un contacto íntimo con la naturaleza, este plan es para ti.
El plan incluye la visita a playas y montañas y caminatas en los lugares más paradisíacos del
país.</p>
  <p class="precio">5 días y 4 noches por $1000</p>
</div>
<div class="planes" id="galeriaHistorico">
  <h1>Plan histórico</h1>
  
  <p>Si lo tuyo es el turismo histórico, este plan te ofrece un recorrido por las
distintas ruinas del país, así como una visita a los principales museos de guerra de El
Salvador.</p>
  <p class="precio">5 días y 4 noches por $1000</p>
</div>
<div class="planes" id="galeriaRelax">
  <h1>Plan Relax</h1>
  
  <p>Si quieres recuperarte del estrés de la ciudad este plan podría interesarte.
Incluye una visita por los lugares más paradisíacos y tranquilos del país.</p>
  <p class="precio">5 días y 4 noches por $1000</p>
</div>
```


30. Añada las siguientes reglas de estilo dentro de su archivo fijo.css

```
/*  CONTENIDO DE LA PAGINA PLANES */

.planes{
  width: 900px;
  background: white;
  padding: 10px 30px;
}

.planes h1{
  text-align: left;
  font-size: 1.5em;
}

.planes .precio{
  font-size: 1.1em;
  margin: 20px 0px 5px;
  padding-bottom: 15px;
  border-bottom: solid 2px grey;
  font-weight: bold;
}
```

El contenido de su página planes.html debería lucir de la siguiente manera:

Plan naturaleza



Si te gusta sentir un contacto íntimo con la naturaleza, este plan es para ti. El plan incluye la visita a playas y montañas y caminatas en los lugares más paradisíacos del país.

5 días y 4 noches por \$1000

Página galeria.html

31. Creé una copia de su archivo index.html y guárdela con el nombre galeria.html.
32. Dentro del archivo **galeria.html** elimine todo el contenido de la sección con id “contenido” y reemplácelo por el siguiente código HTML:

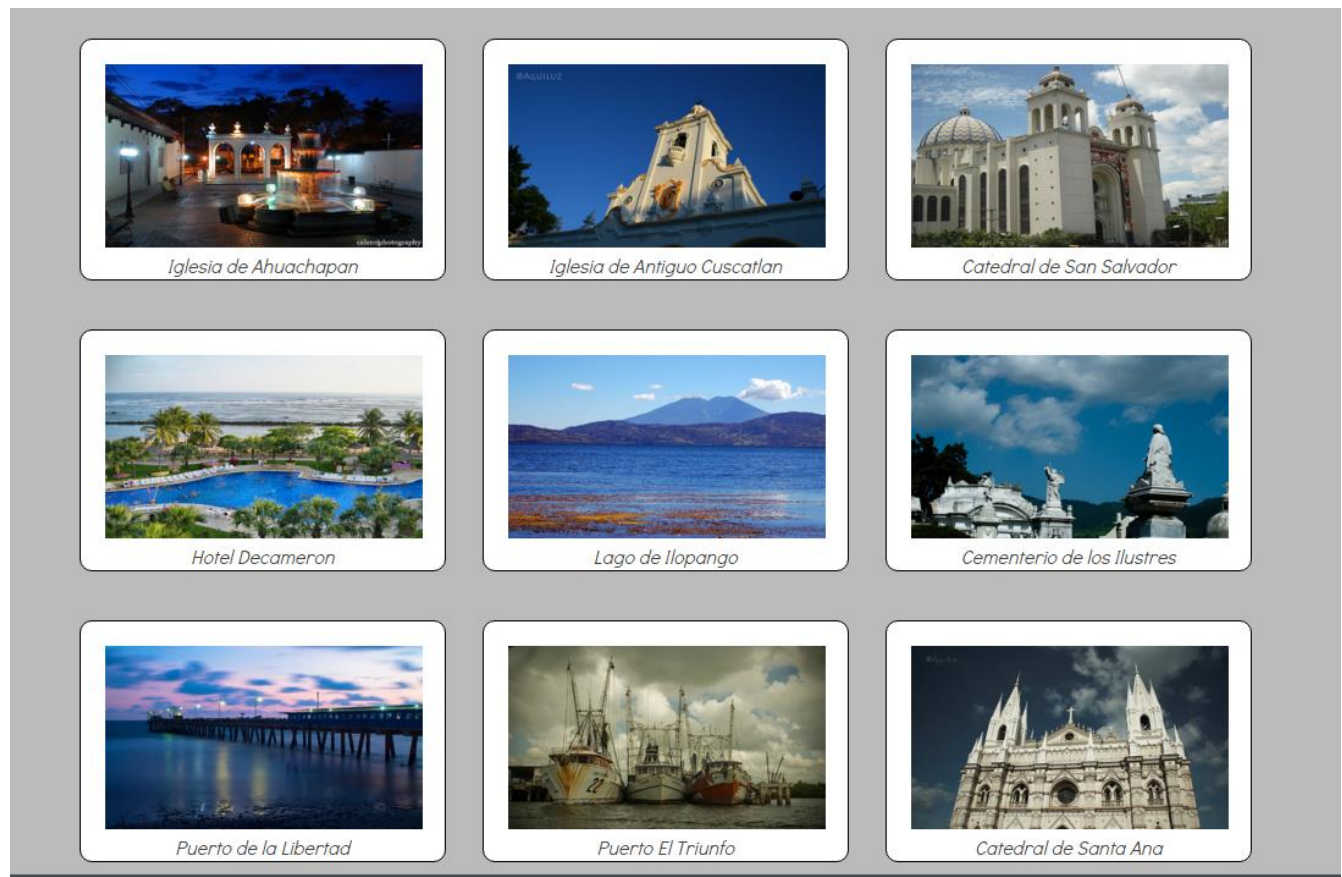
```
<div id="galeria">
  <div class="plan">
    
    <p>Iglesia de Ahuachapan</p>
  </div>
  <div class="plan">
    
    <p>Iglesia de Antiguo Cuscatlan</p>
  </div>
  <div class="plan">
    
    <p>Catedral de San Salvador</p>
  </div>
  <div class="plan">
    
    <p>Hotel Decameron</p>
  </div>
  <div class="plan">
    
    <p>Lago de Ilopango</p>
  </div>
  <div class="plan">
    
    <p>Cementerio de los Ilustres</p>
  </div>
  <div class="plan">
    
    <p>Puerto de la Libertad</p>
  </div>
  <div class="plan">
    
    <p>Puerto El Triunfo</p>
  </div>
  <div class="plan">
    
    <p>Catedral de Santa Ana</p>
  </div>
</div>
```

33. Añada las siguientes reglas de estilo dentro de su archivo fijo.css

```
/* CONTENIDO DE LA PAGINA GALERIA*/
#galeria .plan p{
  text-align: center;
  font-style: italic;
}

#galeria .plan{
  font-size: 1em;
  margin-bottom: 10px;
  padding-bottom: 1px;
}
```

El contenido de su página galeria.html debería lucir de la siguiente manera:



Ejercicios complementarios.

1. Consiga que la imagen que se muestra en la parte superior de su diseño fluido, se sustituya por un slider de imágenes manual o automático (las imágenes que debe mostrar en el slider son: puerto_la_libertad.jpg, santa_ana.jpg y salvador_del_mundo.jpg. Estas imágenes están dentro de los recursos de la guía).

Fecha de entrega: Próxima semana.