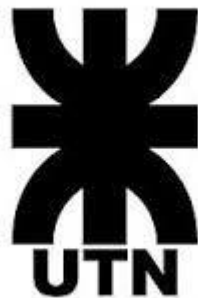


U2 HTML - PRACTICA 3

Agregando elementos a nuestra primera Página Web



UTN FRRO
Versión 1.1

Contenido

Versión	1
Pasos	2
#1 - Crear la carpeta para la práctica 3 dentro del repositorio existente.	2
#2 - Definir archivo inicial y agregar estilos de template.....	2
#3 - Definir la estructura básica de nuestra página	4
#4 - Agregar contenido a página principal.....	5
#5 - Refactoring de código CSS.....	5
#6 - Crear página de multimedia	6
#7 - Crear página de contacto	6
#8 - Testear nuestro sitio.....	6
#9 - Subir código a GitHub.....	6

Versión

ID	Autor	Cambio	Fecha
1.0	Fabrizio García		
1.1	Eliana Gomez	Cambio de formato	30/7/2022

Pasos

#1 - Crear la carpeta para la práctica 3 dentro del repositorio existente.

Continuando con la clase anterior, debe usarse el mismo repositorio y carpeta creadas en esa ocasión.

Dentro de nuestra carpeta “**utn-labcomII-practices**” crear una nueva carpeta llamada “**U02P03**”. Dentro de esa carpeta, crearemos un archivo llamado “**index.html**” donde escribiremos el código de esta práctica.

#2 - Definir archivo inicial y agregar estilos de template

Armar una estructura básica de archivo de HTML5.

Agregar el siguiente código dentro de la etiqueta **<head>** del documento HTML. Estos estilos ayudarán a que nuestra página tenga un mejor estilo y reconocer visualmente todos los contenidos de nuestra página.

```
<style>
  * {
    box-sizing: border-box;
  }

  body {
    font-family: Verdana, Geneva, Tahoma, sans-serif;
    color: #aaaaaa;
  }

  div.container {
    overflow: auto;
  }

  header {
    background-color: #e5e5e5;
    padding: 15px;
    text-align: center;
  }

  nav {
    float: left;
    width: 20%;
    text-align: center;
  }

  nav ul {
    padding-left: 0px;
  }

  nav li {
    list-style: none;
  }
```

```

nav li a {
  background-color: #e5e5e5;
  padding: 8px;
  margin-top: 7px;
  display: block;
  width: 100%;
  color: black;
  font-weight: 700;
}

main {
  float: left;
  width: 60%;
  padding: 0 20px;
}

.right {
  background-color: #e5e5e5;
  float: left;
  width: 20%;
  padding: 15px;
  margin-top: 7px;
  text-align: center;
}

footer {
  background-color: #e5e5e5;
  text-align: center;
  padding: 10px;
  margin-top: 7px;
}

/* For mobile phones: */
@media only screen and (max-width: 620px) {

  nav,
  main,
  .right {
    width: 100%;
  }
}
</style>

```

#3 - Definir la estructura básica de nuestra página

Nuestra página estará compuesta por un encabezado - “header”, un contenedor principal que estará dividido en 3 secciones (navegación - “nav”, contenedor principal - “main” y contenedor secundario “div.right”). Por último mostraremos un pie de página - “footer”.

Esta estructura básica es la que deberán tener las 3 páginas que implementaremos en nuestro sitio web, tanto el header como el nav y el footer deberán ser prácticamente iguales, lo que variará es el propio contenido que se encuentra dentro del elemento “main”.

Este es el código html a colocar dentro del elemento “body”:

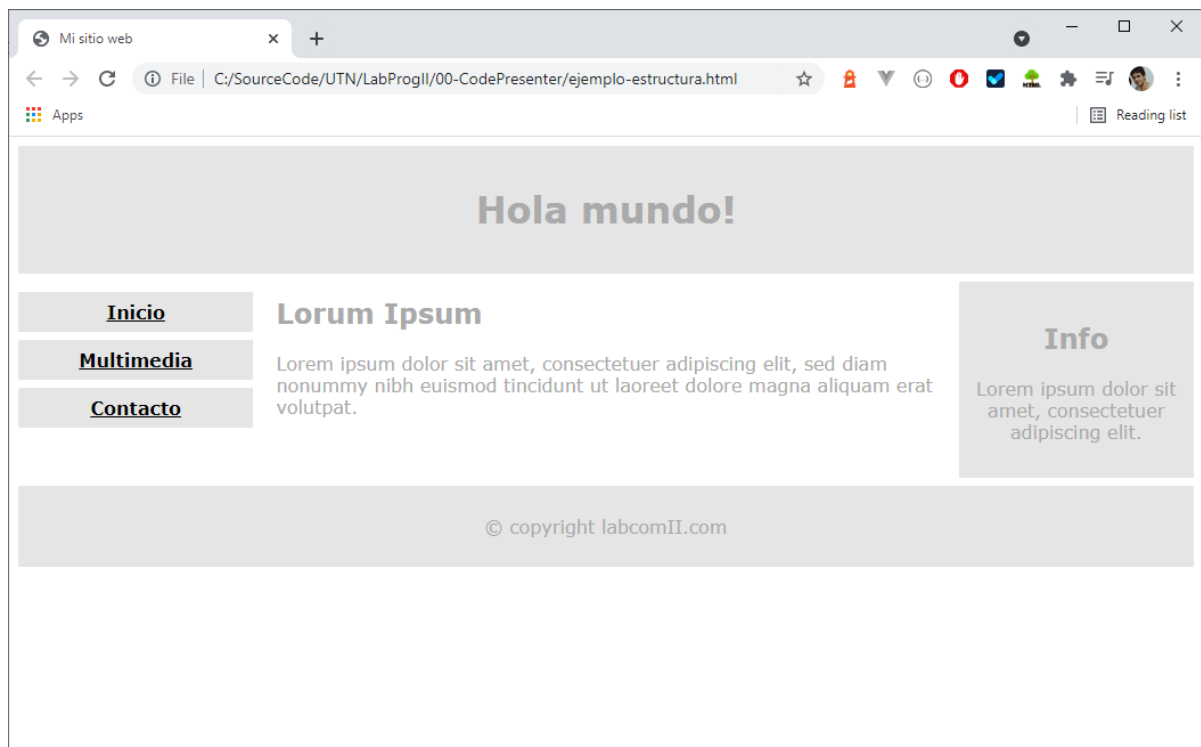
```
<!-- Encabezado -->
<header>
  <h1>Hola mundo!</h1>
</header>

<div class="container">
  <!-- Navegación / cotenedor izquierdo -->
  <nav>
    <ul>
      <li><a href="#">Inicio</a></li>
      <li><a href="#">Multimedia</a></li>
      <li><a href="#">Contacto</a></li>
    </ul>
  </nav>
  <!-- Contenedor principal -->
  <main>
    <section>
      <h2>Lorum Ipsum</h2>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed
diam nonummy nibh euismod tincidunt ut
      laoreet dolore magna aliquam erat volutpat.</p>
    </section>
  </main>
  <!-- Contenedor secundario / derecho -->
  <div class="right">
    <h2>Info</h2>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  </div>
</div>
<!-- Pie de Página-->
<footer>
  <p>© copyright labcomII.com</p>
</footer>
```

Una vez incorporado el código html en tu index.html, tomate unos minutos para analizarlo y compararlo con lo que renderiza el navegador.

***TIP:** Siempre que copies y pegues código de otra fuente, luego de hacerlo presiona “alt” + “shift” + “F” lo que hará que tu código se indente correctamente.*

Así se debería visualizar nuestro sitio:



#4 - Agregar contenido a página principal

Colocar un texto al header, footer y al contenedor derecho acorde a tu sitio.

Configurar la navegación para que se dirija a las páginas “multimedia.html” y “contact.html” (en los próximos pasos agregaremos tales archivos).

Luego, agrega contenido a la propia página dentro de “main”. Agregar por lo menos dos secciones, por ejemplo, una puede ser donde se listen 3 noticias y otra donde se listen 3 servicios.

#5 - Refactoring de código CSS

En este punto tenemos nuestro código CSS dentro del index.html, pero ¿qué pasa si queremos que todas nuestras páginas compartan el mismo código base?.

Pues tendremos que llevar ese código incrustado en el index.html a un archivo .css. Para ello debemos seguir los siguientes pasos:

1. Crear una carpeta llamada “css” y agregar un archivo llamado “index.css”.
2. Copiar el código que se encuentra dentro de los tags `<style></style>`, pegarlo dentro del index.css y guardar los cambios realizados.
3. Eliminar el elemento `<style></style>` que tenemos dentro del index con todo su contenido.

4. Indexar el archivo index.css a nuestra página agregando la siguiente línea también dentro del **head**: `<link rel="stylesheet" href="css/index.css">`
5. Guardar los cambios y asegurarse de que nuestro sitio se siga viendo correctamente.

#6 - Crear página de multimedia

Agregar la página "multimedia.html". Esta página debe tener la misma estructura que el index.html pero dentro del contenedor main agregar al menos una imagen, un video y archivo de audio.

#7 - Crear página de contacto

Agregar la página "contact.html". Esta página debe tener la misma estructura que el index.html pero dentro del contenedor main se debe agregar un formulario.

Se deberá utilizar una tabla para dar formato a nuestro formulario, la tabla debe contener dos columnas, en la primera se debe colocar el label y en la segunda se debe colocar el propio input como se muestra en este ejemplo:

Nombre:

Además, el formulario deberá contemplar los siguientes campos con sus respectivas validaciones:

- **Nombre:** texto | requerido
- **Email:** email | requerido
- **Edad:** number
- **Color preferido:** color
- **Comentario:** textarea | min:10

#8 - Testear nuestro sitio

Realizar un test integral de nuestro sitio web, asegurarnos que funcionan correctamente todas las funcionalidades, por ejemplo:

- navegación: que desde cada una de las páginas puede ingresar a las otras
- formulario: que las validaciones respondan correctamente a todos los escenarios.

#9 - Subir código a GitHub

Subir todo el código al repositorio.