



El futuro digital
es de todos

MinTIC

Hechos

QUE

CONECTAN



CICLO 3 EJE TEMÁTICO 5 FRONTEND

Universidad
Industrial de
Santander



Mision
TIC 2022

FRONTEND

Ruta de aprendizaje

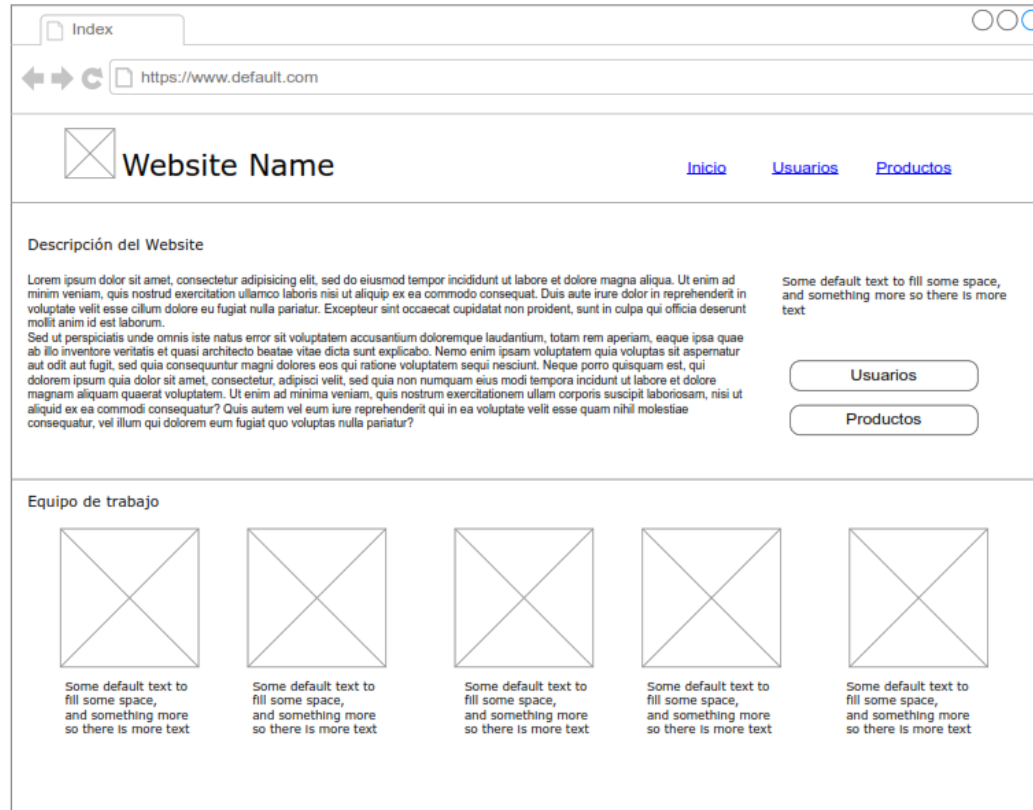


FRONTEND

En el entorno del desarrollo de software, la parte que es presentada para interacción con los usuarios, es llamada Frontend, la cual dispone de un conjunto de servicios desarrollados para una aplicación y son accedidas directamente por dichos usuarios. En una aplicación web, las tecnologías que soportan cada una de las pantallas desarrolladas, corren directamente sobre el navegador.

Palabras clave: WEB, Frontend, HTML, CSS, JS, Ajax.

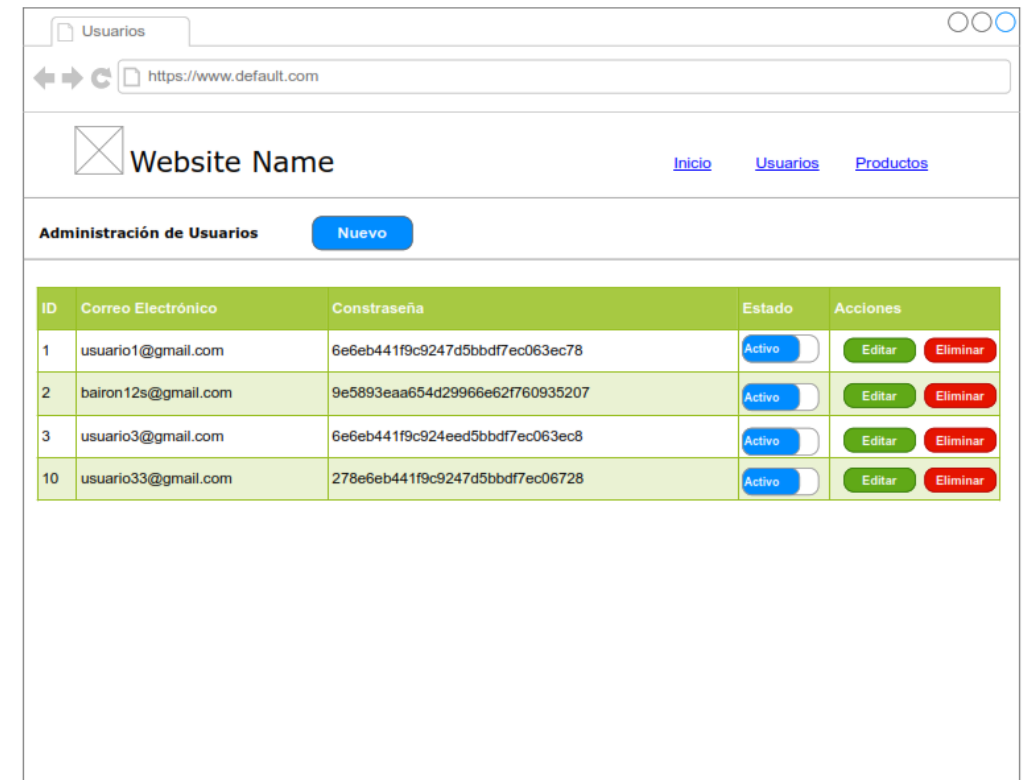
Diseño



- El diseño o diseño web es un área enfocada a planificar, diseñar, mantener y crear las **interfaces digitales**.
- Permite contestar la pregunta **¿Qué se va a desarrollar?**
- Tanto el Cliente como los Desarrolladores saben que se va a **entregar**
- Cuanto más **funcional** y **atractivo** sea el diseño de tu página, mejor será el resultado que obtengas.

Diseño - Wireframe

Wireframe es una de las fases iniciales en la creación de un producto digital y consiste en el diseño de la estructura que tendrá cada una de las pantallas o interfaces de ese producto. Es decir, un wireframe es el boceto de cómo serán cada una de las pantallas de una web o app.



Diseño - Wireframe

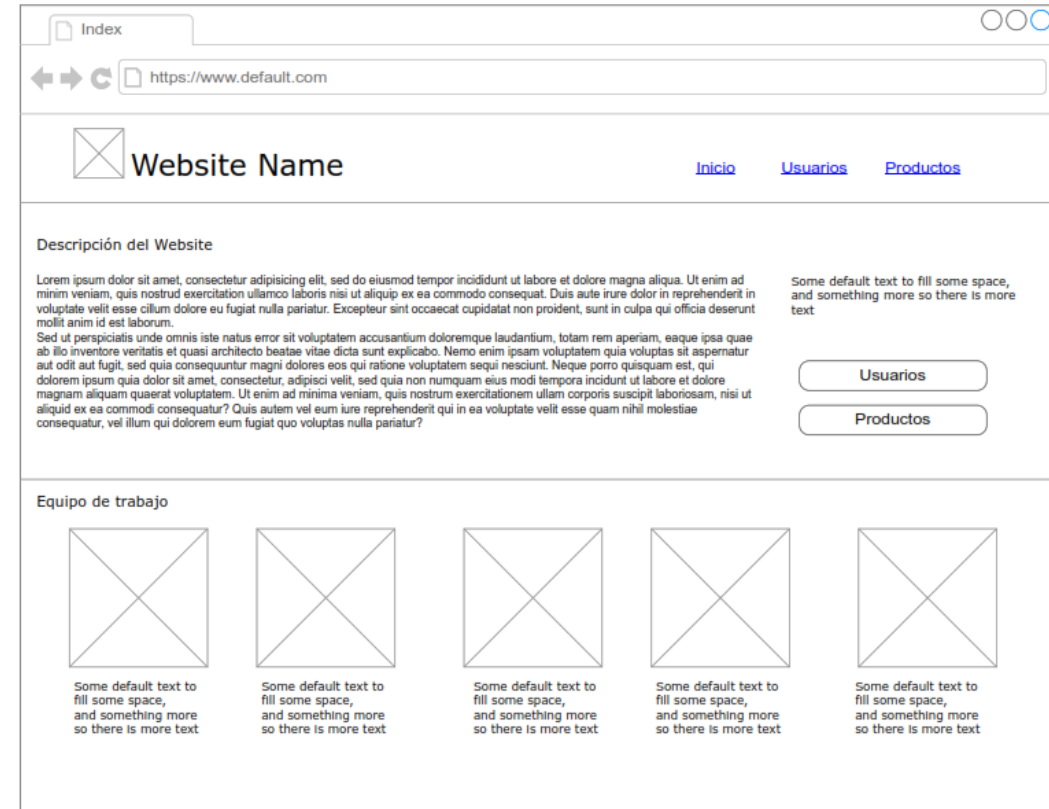
The wireframe shows a web browser window with a single tab labeled 'Productos'. The address bar contains 'https://www.default.com'. The header area includes a placeholder for a logo, the text 'Website Name', and three navigation links: 'Inicio', 'Usuarios', and 'Productos'. Below the header is a section titled 'Administración de Productos' with a blue 'Nuevo' button. The main content area features a table with the following data:

ID	Nombre	Valor de Compra	Valor de Venta	Cantidad	Acciones
1	Koca Kola	1.200	1.700	45	<button>Editar</button> <button>Eliminar</button>
2	Producto nuevo	3.450	4.000	90	<button>Editar</button> <button>Eliminar</button>
3	Jabón Super	1.800	2.200	110	<button>Editar</button> <button>Eliminar</button>
10	Bebida de café	2.500	3.000	27	<button>Editar</button> <button>Eliminar</button>

- La estética de los wireframes debe ser simple, los colores deben ser en escala de grises (blanco, negro y todos los grises en medio).
- No se deben usar más de dos fuentes de caracteres. A esta altura no es necesario emplear la tipografía como medio para comunicar jerarquía de la información.

Diseño - Wireframe

- No se debe recargar los wireframes con imágenes y videos, pero si donde van ubicados, para ello utilizamos un rectángulo con una equis para representar donde van las imágenes y un triángulo (play icon) para representar donde van los videos.
- Considera el tamaño de las pantallas donde vas a ejecutar tu aplicación y así mismo has un diseño para cada uno. Se debe tener en consideración los dispositivos, la orientación de las pantallas y el contexto de uso para ubicar qué recursos deben ir.



HTML (HyperText Markup Language)

El Lenguaje de Marcado de Hipertexto (HTML) es el código que se utiliza para estructurar y desplegar una página web y sus contenidos.

Estos contenidos podrían ser párrafos, una lista con viñetas, o imágenes y tablas de datos, etc.

HTML no es un lenguaje de programación!

Enlace para lectura:

https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/HTML_basics

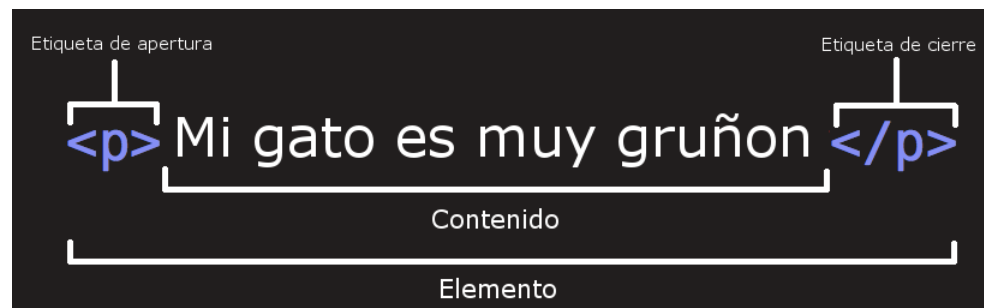


HTML (HyperText Markup Language)

- **HTML no es un lenguaje de programación;** es un lenguaje de marcado o etiquetado que define la estructura de tu contenido.
- HTML consiste en una serie de elementos que usarás para encerrar diferentes partes del contenido, para que estos se vean o comporten de una determinada manera.
- Las etiquetas de encierro pueden hacer de una palabra o una imagen un hipervínculo a otro sitio, se pueden cambiar palabras a cursiva, agrandar o achicar la letra, etc.

```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>CICLO III - MINTIC 2022 </title>
8 </head>
9 <body>
10
11 </body>
12 </html>
```

HTML (HyperText Markup Language)



Las partes principales del elemento son:

- La etiqueta de apertura: consiste en el nombre del elemento (en este caso, p), encerrado por paréntesis angulares (`<` `>`) de apertura y cierre. Establece dónde comienza o empieza a tener efecto el elemento —en este caso, dónde es el comienzo del párrafo—.
- La etiqueta de cierre: es igual que la etiqueta de apertura, excepto que incluye una barra de cierre (`/`) antes del nombre de la etiqueta. Establece dónde termina el elemento —en este caso dónde termina el párrafo—.
- El contenido: este es el contenido del elemento, que en este caso es solo texto.
- El elemento: la etiqueta de apertura, más la etiqueta de cierre, más el contenido equivale al elemento.

HTML (HyperText Markup Language)

En HTML existen dos tipos de elementos.

A nivel de Bloque:

Los elementos a nivel de bloque siempre empiezan con una nueva línea y ocupa todo el ancho disponible(Los elementos siempre se amplían de izquierda a derecha).

Los elementos que tendremos a nivel de bloque son:

```
<div>Hola Tripulantes</div>
```

```
<address> <article> <aside> <blockquote> <canvas> <dd> <div>
```

```
<dl> <dt> <fieldset> <figcaption> <figure> <footer> <form>
```

```
<h1> a <h6> <header> <hr> <li> <main> <nav> <noscript>
```

```
<ol> <p> <pre> <section> <table> <tfoot> <ul> <video>
```

HTML (HyperText Markup Language)

Elementos en línea:

Los elementos a nivel de línea no inician con una nueva línea y solo ocupan el ancho que sea necesario por el contenido.

Los elementos que tendremos a nivel de línea son:

```
<span>Hola Tripulantes</span>
```

```
<a> <abbr> <acronym> <b> <bdo> <big> <br>
```

```
<button> <cite> <code> <dfn> <em> <i> <img>
```

```
<input> <kbd> <label> <map> <object> <output> <q>
```

```
<samp> <script> <select> <small> <span> <strong> <sub>
```

```
<sup> <textarea> <time> <tt> <var>
```

Revisión de lo aprendido

Organizar el orden en el que deben ir las etiquetas principales de un documento HTML.

```
<body>
```

```
</html>
```

```
<head>
```

```
</body>
```

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

```
<!DOCTYPE html>
```

```
</head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Document</title>
```

CSS (Cascading Style Sheets)

El CSS u hojas de estilo en cascada, es el lenguaje que describe cómo nuestros elementos en HTML serán mostrados en una pantalla de algún dispositivo con navegador web.

Gracias al CSS tendrás la capacidad de determinar el diseño, color, fuente, entre otras características de tus elementos.

Este lenguaje CSS es de gran ayuda y ahorras mucho trabajo al momento de programar un sitio web, ya que puedes controlar el aspecto de múltiples páginas dentro del sitio, a la vez con una sola hoja de estilos o documento de CSS.

Enlace para lectura

<https://developer.mozilla.org/es/docs/Web/CSS>

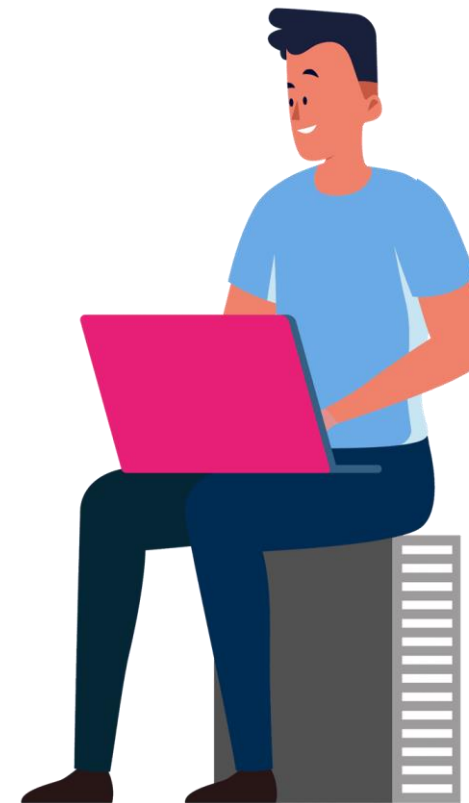


CSS

El CSS y el HTML se renderiza en los navegadores con una separación del contenido y la presentación o estilos de este contenido, y este proceso se lleva a cabo en el siguiente orden:

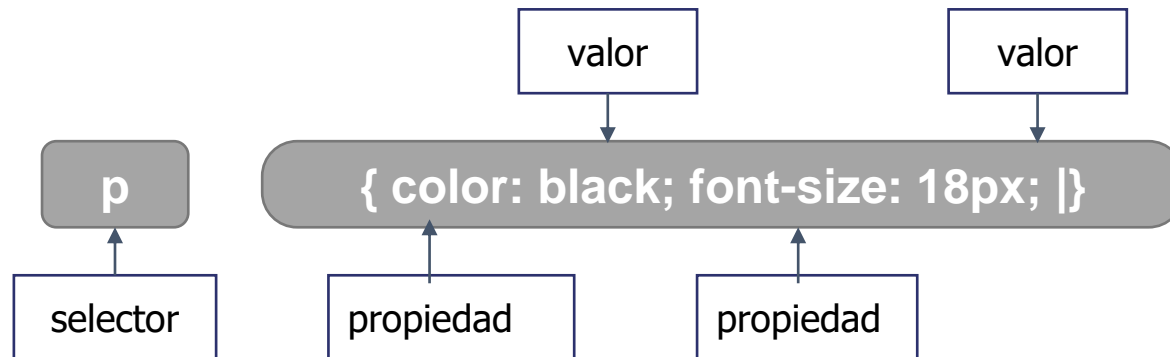
1. El navegador convierte HTML y CSS en un DOM (Objetos definidos en la página) este DOM funciona como un documento para la memoria del ordenador, de forma que puede combinar el contenido del documento en texto con su hoja de estilo.
2. El navegador muestra el contenido del DOM al usuario.

Los componentes de las páginas web son solamente interpretados del lado del cliente (navegador web).



Sintaxis CSS

- El conjunto de reglas CSS se constituye de un selector y un bloque de declaración.
- Los selectores son apuntadores a los elementos o etiquetas del HTML a las que se le aplicará los estilos.
- Los bloques de declaración contienen una o más declaraciones separadas por punto y coma(;).
- Para cada declaración se define un nombre de la propiedad CSS acompañado de su valor y separados por (;).
- Los bloques de declaración están definidos por llaves.



Selectores de CSS

- **Selector de tipo** define todos los elementos que coincidan con el nombre especificado.



El selector ***p*** definirá las propiedades de todos los elementos ***p*** del documento.

Selectores de CSS

- **Selector de clase** selecciona a todos los elementos del documento HTML con el atributo de clase especificado.



Para la selección de los elementos que contengan la clase ***texto*** se escribe primero el carácter de ***punto*** (.), seguido del nombre de la clase.

Los nombres de las clases no deben empezar por números.

Selectores de CSS

- **Selector de id**, el cual hace uso del atributo id de un elemento HTML para la selección específica del mismo.



El identificador debe ser único dentro del documento HTML, se usará para la selección única del mismo elemento y se escribe primero el Hashtag o numeral (#), seguido del nombre del identificador.

El nombre de identificación no deben empezar por números.

Selectores de CSS

- **Selector universal**, el cual hace referencia a todos los elementos HTML.



El identificador universal * afectará a todos los elementos del documento HTML.

Selectores de CSS

- **Selector universal**, el cual hace referencia a todos los elementos HTML.



El identificador universal * afectará a todos los elementos del documento HTML.

Aplicación del CSS al HTML

Para la aplicación de CSS a HTML existen tres métodos:

- **Hoja de estilo externa:** Este método se utiliza para reutilizar CSS ya que se puede aplicar a los documentos necesarios, lo que significa que los CSS están en un archivo externo con extensión .css y se vinculan desde un elemento <link> de HTML haciendo referencia a su ruta(local o en la nube) mediante el atributo href.

```
<link rel="stylesheet" href="styles.css">
```

- **Hoja de estilo interna:** Cuando no hay archivos CSS externos se usan los CSS mediante el elemento <style> contenido dentro del elemento <head> del documento HTML.

```
<style> h1 {color: blue;} ..... </style>
```

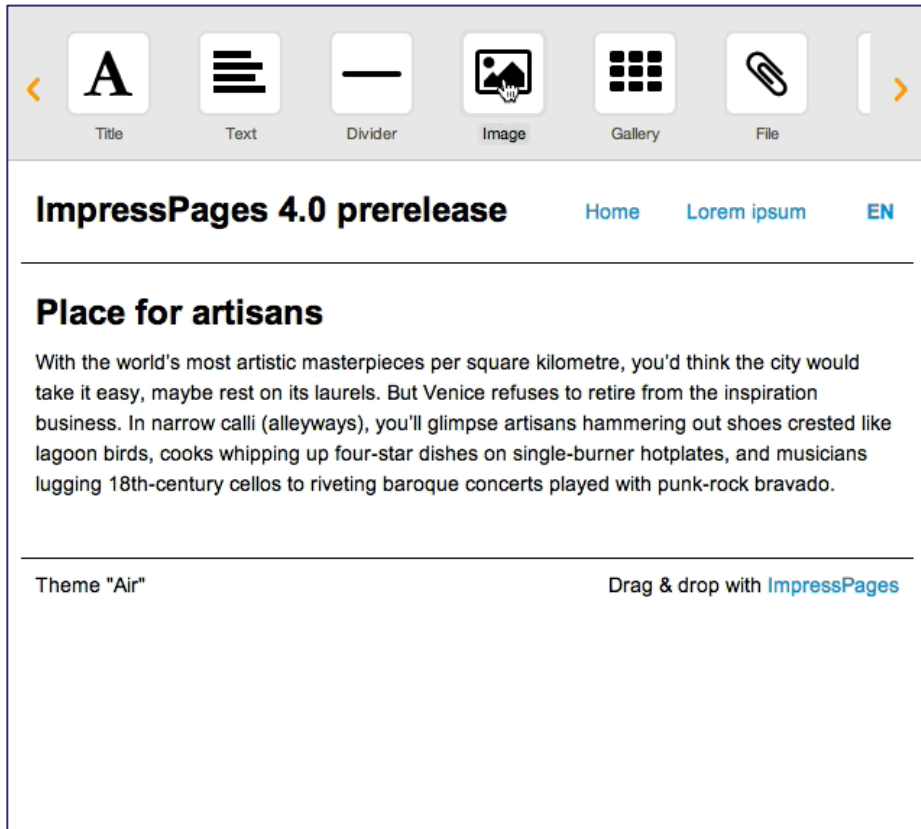
- **Estilos en línea:** Las declaraciones en línea de CSS que solo afectan al elemento HTML mediante el atributo style.

```
<p style="color:blue;">Este es mi viaje como tripulante por HTML</p>
```

No se recomienda su uso, solo en casos que sean requeridos.

JavaScript

Lenguaje de programación ligero para ejecutar secuencias de comandos en páginas web. Dispone de varios framework de trabajo desarrollados con base en este lenguaje que permite volver dinámica la interacción con los sitios web.



JavaScript permite la interacción de arrastrar y soltar elementos dentro de un documento html y entre otras cosas más.

[ImpressPages 4.0 Drag&drop Column Widget.gif \(636×572\) \(wikimedia.org\)](#)

Aplicación del JS al HTML

Para la aplicación de JS a HTML existen dos métodos:

- **JavaScript externo:** Este método se utiliza para reutilizar el JS ya que se puede aplicar a los documentos necesarios, lo que significa que los JS están en un archivo externo con extensión .js y se vinculan desde un elemento `<script></script>` de HTML haciendo referencia a su ruta(local o en la nube) mediante el atributo src.

```
<script src="js/principal.js"></script>
```

- **JavaScript interno:** Cuando no hay archivos JS externos se usan los JS mediante el elemento `<script></script>` al final del elemento `<body>` del documento HTML.

```
<script> function SaludarTripulante(){ alert("Un Saludo Tripulante."); } </script>
```


Declaraciones de variables

En Js existen tres formas de declarar variables:

- **var:**
Se usa para declarar variables de tipo global y local.
var number = 10;
- **let:**
Se usa para declarar variables locales o con ámbito de bloque.
let number = 10;
- **const:**
Se usa para declarar variables de sólo lectura local o con ámbito de bloque.
const pi = 3.1416;

Funciones

Las funciones en JavaScript son bloques o un conjunto de instrucciones la cual realiza un proceso o calcula un valor.

Para definir una función consta de:

- Palabra reservada function.
function addUser() ...
- Conjunto de parámetros de funciones, entre paréntesis y separados por comas.
function addUser(name, age ...)
- La definición de la función esta debe de estar encerrada en llaves.
function addUser(name, age ...) { ... }

XMLHttpRequest

Es un objeto JavaScript que proporciona una forma de obtener información desde una URL, haciendo uso del protocolo HTTP sin tener que recargar la página completa.

Para crear una instancia de XMLHttpRequest:

```
let solicitud = new XMLHttpRequest();
```

Iniciar a la solicitud:

```
solicitud.open('GET', "URL");
```

Especificar el tipo de respuesta:

```
solicitud.responseType = 'json';
```

Enviar la solicitud:

```
solicitud.send();
```

Obtener los datos de la solicitud:

```
solicitud.onload = function() {  
    const usuarios = solicitud.response;  
}
```

Revisión de lo aprendido

Teniendo presente el siguiente ejemplo, ¿Qué selector y propiedades se debe implementar para modificar la fuente a Arial de los elementos p (<p>) ?

```
<h3>Lorem ipsum dolor sit amet. </h3>

<p> Lorem ipsum dolor sit amet
consectetur, adipisicing elit. Aut
voluptates ullam est adipisci velit,
nulla illo accusamus magnam explicabo,
animi itaque assumenda nobis impedit
provident neque voluptatum omnis
nesciunt voluptatem?</p>

<button> Next</button>
```

a)

```
1 * {
2   font: Arial;
3 }
```

b)

```
1 .p {
2   font-family: Arial;
3 }
```

c)

```
1 p {
2   font-family: Arial;
3 }
```

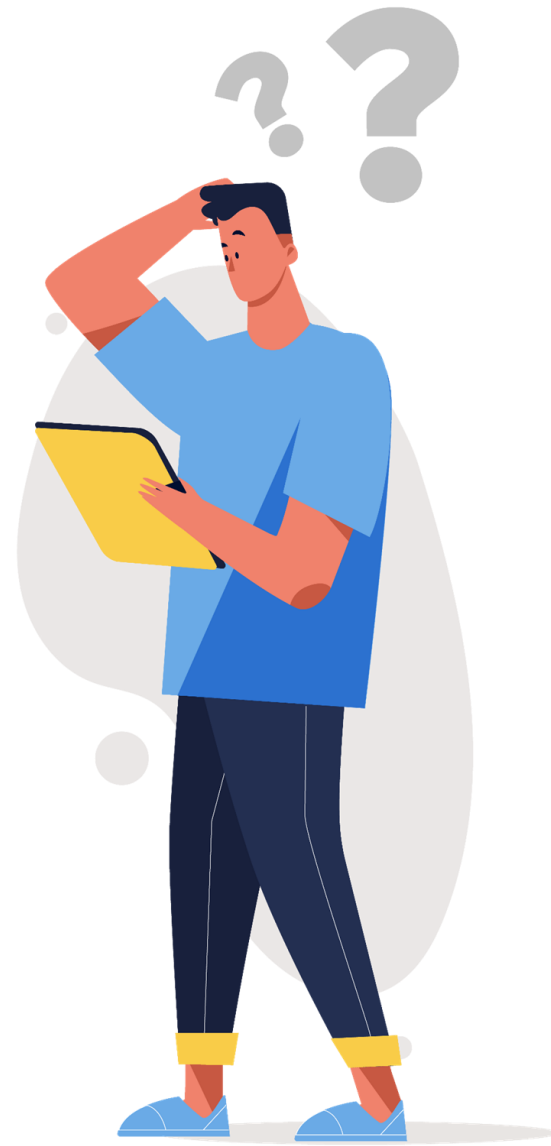
d)

```
1 p {
2   font: Arial;
3 }
```

Revisión de lo aprendido

Determine si la siguiente afirmación es falsa o verdadera:
El objeto XMLHttpRequest admite protocolos HTTP y el formato XML

- a) Verdadero
- b) Falso





El futuro digital
es de todos

MinTIC

Hechos

QUE

CONECTAN



CICLO 3 EJE TEMÁTICO 5 FRONTEND

Universidad
Industrial de
Santander



Mision
TIC2022