

# Semestre FJ 2024: #A1



1

name:

LUIS MARCO BARRIGA BAEZ

---

---

<sup>1</sup> Illustration by Freepik Stories

- *Use your own words for the answers.*
- *Write your references.*
- *English o Español o un mix.*
- *If you already know how to do something don't do it, you just have to write: "Done"*

# 1.HTML

## Introduction

- What is HTML?  
HTML significa lenguaje de marcado de hipertexto. Es el lenguaje de marcado estándar para la creación de páginas web y aplicaciones web.
- What is a Tag?  
Las etiquetas HTML son elementos que definen la estructura y el contenido de una página web. Consisten en etiquetas de apertura y cierre envueltas alrededor del contenido para indicar cómo debe mostrarse o comportarse ese contenido.
- What is a DOCTYPE? How do you create one for an HTML5 document?  
Una declaración DOCTYPE especifica el tipo de documento y la versión de HTML que se utiliza. Para un documento HTML5, la declaración DOCTYPE es simple:  
`<!DOCTYPE html>`.
- How do you create an encoding for an HTML5 document?  
Puede especificar la codificación de caracteres para un documento HTML5 utilizando la metaetiqueta dentro de la sección del encabezado: `<meta charset="UTF-8">`.
- Explain the head and the body tags.  
La etiqueta `<head>` contiene metainformación sobre el documento HTML, como título, juego de caracteres, estilos, secuencias de comandos y más. La etiqueta `<body>` contiene el contenido del documento HTML que es visible para los usuarios.

## Basic Tags

- Show an example of the **a** tag.  
`<a href="https://www.instagram.com">Instagram</a>`
- Explain the **div** and the **span** tag. What is the difference between them?  
La etiqueta `<div>` es un elemento a nivel de bloque que se utiliza para agrupar elementos HTML y aplicarles estilos o secuencias de comandos de forma colectiva.  
La etiqueta `<span>` es un elemento en línea que se utiliza para aplicar estilos o secuencias de comandos a una porción específica de texto o elementos en línea.
- Explain the **section**, **article**, **aside**, **header**, **footer** tag?  
Estos son elementos semánticos HTML5 que se utilizan para definir diferentes partes de una página web:

`<sección>`: Representa una agrupación temática de contenido.

`<artículo>`: representa un contenido autónomo.

`<aside>`: Representa contenido relacionado tangencialmente con el contenido que lo rodea.

`<encabezado>`: representa contenido introductorio que normalmente se coloca en la parte superior de una sección o página.

`<pie de página>`: representa el contenido del pie de página que normalmente se coloca en la parte inferior de una sección o página.

- Explain the purpose of the **h1....h6** tags.  
Estas etiquetas representan encabezados de diferentes niveles, siendo `<h1>` el más alto (el más importante) y `<h6>` el más bajo.
- Explain the attributes of the **img** tag. Write an example.  
La etiqueta `<img>` se utiliza para incrustar imágenes en un documento HTML. Los atributos incluyen `src` (fuente), `alt` (texto alternativo), `ancho`, `alto`, etc. Ejemplo: ``
- Create an unordered list. Create an ordered list.  
`<ul>`  
    `<li>Producto 1</li>`  
    `<li>Producto 2</li>`  
`</ul>`  
-----  
`<ol>`  
    `<li>Producto 1 ordenado</li>`  
    `<li>Producto 2 ordenado</li>`  
`</ol>`
- Create a basic table using html.

```
<table border="1">
  <tr>
    <td>Celda 1</td>
    <td>Celda 2</td>
  </tr>
  <tr>
    <td>Celda 3</td>
    <td>Celda 4</td>
  </tr>
</table>
```

## Other Tags

- Explain how you can create a **form** in html. Explain the input types. Show an example.

Se crea un formulario utilizando la etiqueta `<form>` y los elementos de entrada se utilizan para recopilar datos del usuario. Los tipos de entrada incluyen texto, contraseña, botones de opción, casillas de verificación, etc.

Ejemplo:

```
<form action="/submit" method="post">
  <label for="username">Username:</label>
  <input type="text" id="username" name="username">
  <input type="submit" value="Submit">
</form>
```

- Explain the **video** tag. Are there disadvantages to using the **video** tag?  
La etiqueta `<video>` se utiliza para incrustar vídeos en un documento HTML. Las desventajas incluyen problemas de compatibilidad del navegador, diferentes formatos de video necesarios para la compatibilidad con varios navegadores y un posible impacto en el rendimiento.

## 2. CSS

### Introduction

- What is CSS?  
CSS significa hojas de estilo en cascada. Es un lenguaje de hojas de estilo utilizado para describir la presentación de un documento escrito en HTML.

- What is the purpose of CSS?  
El propósito de CSS es separar el contenido de una página web de su presentación, permitiendo a los desarrolladores controlar el diseño, los colores, las fuentes y otros aspectos visuales de una página web.
- How do you apply a CSS to an HTML (explain the 3 options)  
CSS se puede aplicar a HTML de tres maneras: estilos en línea, estilos internos y hojas de estilo externas.
- What is a selector? What is a CSS rule?  
Un selector es un patrón que coincide con elementos de un documento. Una regla CSS consta de un selector y un bloque de declaración, que contiene una o más declaraciones separadas por punto y coma.

## Basic CSS

- **Explain a:**
  - **Tag Selector:** Selects all elements of a specific tag type.
  - **Class Selector:** Selects elements with a specific class attribute.
  - **Tag and Class Selector:** Selects elements that match both the tag and class criteria.
  - **Id Selector:** Selects elements with a specific ID attribute.
- What is a **class**? What is an **id**? How are they related to CSS?  
Las clases y los ID son atributos que se utilizan para identificar de forma única elementos o grupos de elementos HTML. Están relacionados con CSS porque proporcionan ganchos para diseñar elementos mediante selectores.
- Explain the **CSS Box Model**? Show an example.  
El modelo de caja CSS describe el diseño y la disposición de elementos en CSS. Consta de contenido, relleno, borde y margen.

Ejemplo:

```
.Ejemplo {
  width: 200px;
  height: 100px;
  padding: 20px;
  border: 2px solid black;
  margin: 10px;
}
```

- How can you change the font color using CSS?  
Puede cambiar el color de la fuente usando la propiedad de color.

Ejemplo:

```
.azul {  
  color: blue;  
}
```

## Advanced CSS

- Explain the static, relative, fixed and absolute position.
  - **Estático:** posicionamiento predeterminado; Los elementos se colocan de acuerdo con el flujo normal del documento.
  - **Relativo:** Posicionado con respecto a su posición normal.
  - **Corregido:** posicionado en relación con la ventana gráfica.
  - **Absoluto:** Posicionado en relación con el ancestro posicionado más cercano.
- Explain the display and the visibility properties. Show examples.
  - Display: especifica el comportamiento de visualización de un elemento.

```
.bloque {  
  display: inline-block;  
}
```
  - Visibility: Especifica si un elemento está visible u oculto.

```
.oculto {  
  visibility: hidden;  
}
```

## Create an html challenge

Select an old webpage from (<https://archive.org/web/>). Build a css/html page and try to copy that webpage.

## 3. Javascript

### Introduction

- What is Javascript?

JavaScript es un lenguaje de programación interpretado de alto nivel que se utiliza principalmente para crear páginas web y aplicaciones web interactivas. Permite la manipulación dinámica de contenido, la interacción del usuario y la modificación del comportamiento de las páginas web.

- How can you code JS in a HTML document? Show some examples.  
JavaScript se puede incluir en un documento HTML utilizando etiquetas `<script>`.

Aquí hay ejemplos:

```
<!DOCTYPE html>
<html>
<head>
  <title>JavaScript Ejemplo</title>
  <script>
    // El código va aquí
    console.log("Hola amigos de ITC");
  </script>
</head>
<body>
  <!-- HTML content here -->
</body>
</html>
```

- Declare a variable in JS.  
var x = 5;  
let y = "Hola";  
const PI = 3.14;
- Create a function in JS  
function greet(nombre) {  
 console.log("Hola, " + nombre + "!");  
}
- Create an “object” in JS  
var persona = {  
 firstName: "Luis",  
 lastName: "Marco",  
 age: 19  
};
- Explain loops and conditionals in JS  
Los bucles como for, while y condicionales como if, else, else if son fundamentales en JavaScript para controlar el flujo del programa y la iteración sobre los datos.

## Basic Javascript

- What is the DOM?  
El modelo de objetos de documento (DOM) es una interfaz de programación para documentos web. Representa la estructura de documentos HTML y XML como un

árbol de objetos que pueden manipularse con lenguajes de programación como JavaScript.

- Explain **setAttribute**, **getAttribute**, **getElementsByTagName** and **getElementById**.
  - **setAttribute**: establece el valor de un atributo en el elemento especificado.
  - **getAttribute**: recupera el valor del atributo especificado en el elemento.
  - **getElementsByTagName**: devuelve una colección HTML activa de elementos con el nombre de etiqueta proporcionado.
  - **getElementById**: Devuelve el elemento que tiene el atributo ID con el valor especificado.
- Explain **createTextNode**, **createElement**, **appendChild**
  - **createTextNode**: crea un nuevo nodo de texto.
  - **createElement**: crea un nuevo nodo de elemento con el nombre de etiqueta especificado.
  - **appendChild**: agrega un nodo al final de la lista de hijos de un nodo principal especificado.
- Show 2 examples of how you can manipulate the DOM using JS.
  - `document.getElementById("demo").innerHTML = "Hello, World!";`
  - `var newDiv = document.createElement("div");  
var newContent = document.createTextNode("Hello");  
newDiv.appendChild(newContent);  
document.body.appendChild(newDiv);`
- Explain what is an event?

Un evento en JavaScript es una acción o ocurrencia detectada por el programa que puede ser manejada por controladores de eventos. Los ejemplos incluyen clics del mouse, pulsaciones del teclado y carga de páginas.
- Show 2 examples of using **addEventListener**. (with different events)
  - `document.getElementById("myButton").addEventListener("click", function() {  
 alert("Button clicked!");  
});`
  - `document.getElementById("myElement").addEventListener("mouseover",  
function() {  
 console.log("Mouse over the element!");  
});`



- Explain what AJAX is. Show a very basic example.
  - AJAX (JavaScript y XML asíncronos) es una técnica para crear páginas web dinámicas que actualizan el contenido de forma asíncrona mediante el intercambio de datos con un servidor web detrás de escena.
  - ```
var xhttp = new XMLHttpRequest();
xhttp.onreadystatechange = function() {
  if (this.readyState == 4 && this.status == 200) {
    document.getElementById("demo").innerHTML = this.responseText;
  }
};
xhttp.open("GET", "ajax_info.txt", true);
xhttp.send();
```

## JS and other applications

- Do you know/work with a js frontend framework?  
Sí, los marcos de interfaz como React.js, Angular.js y Vue.js se usan comúnmente para crear interfaces de usuario interactivas y aplicaciones de una sola página (SPA) con JavaScript.

## 4. Frameworks and tools

### Introduction

- What is a framework?  
Un framework es un conjunto prediseñado de bibliotecas de código, herramientas y mejores prácticas que proporcionan un enfoque estructurado para desarrollar aplicaciones de software. Simplifica el proceso de desarrollo al ofrecer componentes reutilizables y convenciones predefinidas.
- What are the advantages and disadvantages of using one?  
Ventajas:  
  
Desarrollo rápido  
Coherencia en la estructura del código.  
Soporte comunitario y documentación.  
Características y funcionalidades integradas  
  
Desventajas:  
  
Curva de aprendizaje

Dependencia del marco

Flexibilidad limitada en algunos casos.

- Are you familiar with some of them (web related)?  
Sí, algunos marcos relacionados con la web incluyen React.js, Angular.js, Vue.js, Express.js, Django, Ruby on Rails, Laravel, etc.

## Prototyping

- Research what Figma is and what it could be used for in web development?  
Figma es una herramienta de diseño basada en la nube que se utiliza para crear interfaces de usuario, estructuras alámbricas, prototipos y proyectos de diseño colaborativo. Permite a los equipos colaborar en tiempo real y ofrece funciones para diseñar diseños responsivos, crear prototipos interactivos y compartir activos de diseño.
- Check if you can generate html/css code from Figma. Try. Explain your result(s). It's ok to fail in this experiment.  
No supe hacer este código con Figma