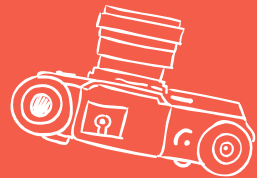




PROGRAMACIÓN WEB I

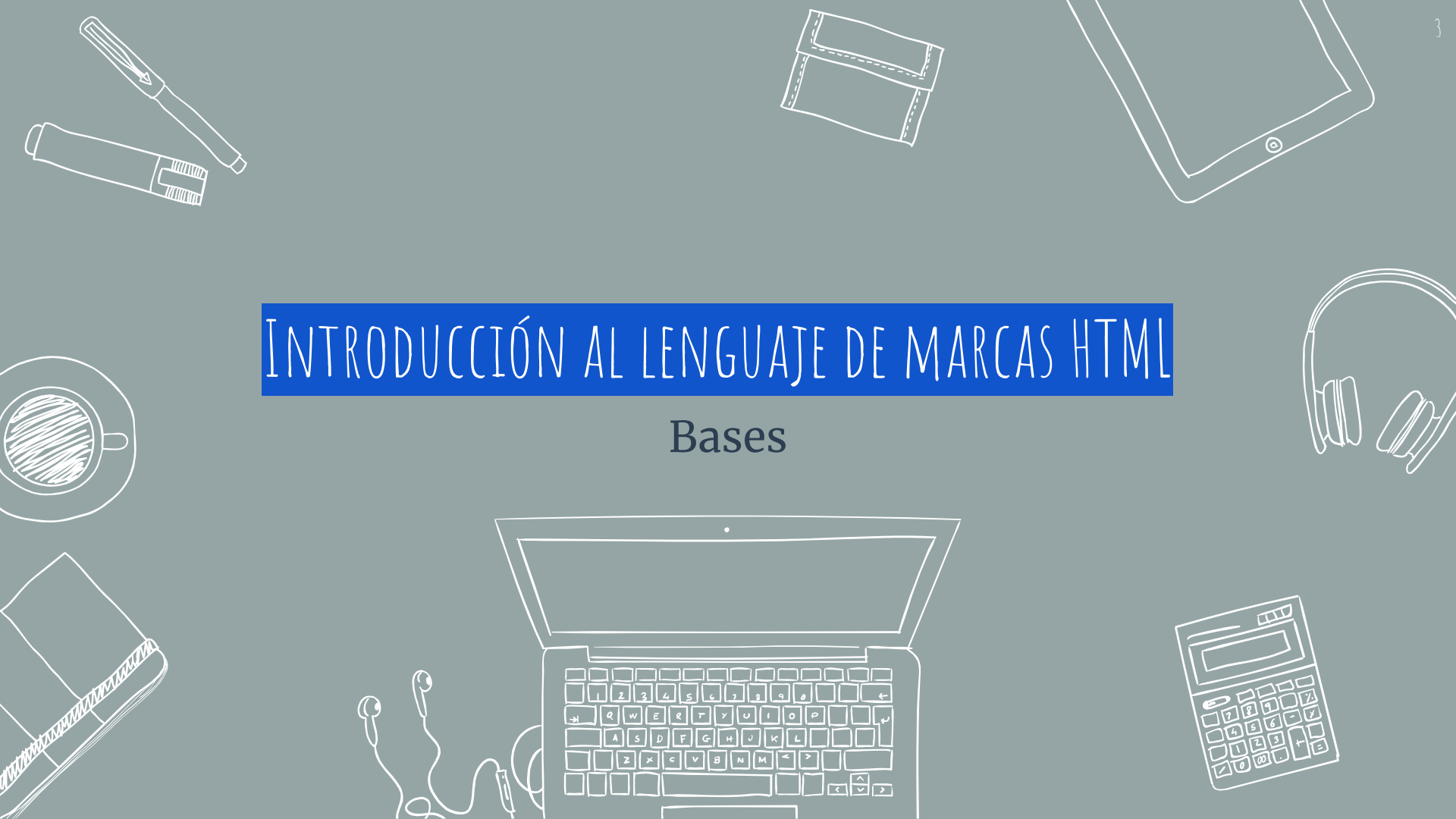
PROFESOR: HERNÁN ROLDÁN

CLASE 3



INTRODUCCIÓN AL LENGUAJE DE MARCAS HTML

Bases





TABLAS

Las tablas en HTML se utilizan para mostrar datos de manera organizada en filas y columnas. Son muy útiles para representar información estructurada, como ser, listado de usuarios, listado de productos y precios, listado de cursos, etc.

La etiqueta `<table>` define una tabla HTML.

Una tabla HTML consta de un elemento `<table>` y uno o más elementos `<tr>`, `<th>` y `<td>`.

El elemento `<tr>` define una fila de tabla, el elemento `<th>` define un encabezado de tabla y el elemento `<td>` define una celda de tabla.

Nota: Para saber más sobre contenedores puedes ingresar [aquí](#).



TABLAS

La etiqueta `<table>` define una tabla HTML.

Una tabla HTML consta de un elemento `<table>` y uno o más elementos `<tr>`, `<th>` y `<td>`.

El elemento `<tr>` define una fila de tabla, el elemento `<th>` define un encabezado de tabla y el elemento `<td>` define una celda de tabla.

Nota: Para saber más sobre contenedores puedes ingresar [aquí](#).



FORMULARIOS

Los formularios HTML son elementos utilizados para recopilar información de los usuarios a través de una página web. Un formulario puede contener diversos tipos de campos, como campos de texto, casillas de verificación, botones de radio, menús desplegables, entre otros, que permiten a los usuarios introducir información o seleccionar opciones.

Para crear un formulario HTML, se utilizan etiquetas especiales de formulario y diversos tipos de campos de entrada, que se pueden personalizar para adaptarse a las necesidades del sitio web.

El tag HTML que se utiliza para crear un formulario es el tag `<form>`

Nota: Para saber más sobre contenedores puedes ingresar [aquí](#).



FORMULARIOS: INPUT

Algunos de los controles input más usados en los formularios son:

- Texto: “text”.
- Contraseña: “password”.
- Email: “email”.
- Número: “number”.
- Fecha: “date”.
- Selección única: “radio”.
- Selección múltiple: “checkbox”.
- Menú desplegable: “select”.
- Botón de envío: “submit”.
- Botón de reinicio: “reset”.

Nota: Para saber más sobre contenedores puedes ingresar [aquí](#).

INTRODUCCIÓN AL CSS

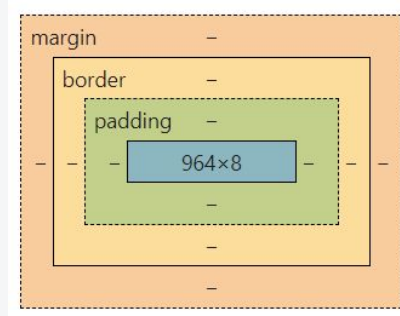
Bases



BOX MODEL: ¿QUÉ ES?

El Box Model (modelo de caja) es un concepto fundamental en CSS que describe cómo se estructura y dimensiona un elemento HTML. Cada elemento en una página web se representa como una caja rectangular con contenido, relleno (padding), borde (border) y margen (margin).

En CSS, estos cuatro componentes son considerados como “capas” que rodean el contenido de un elemento HTML, y cada una de ellas puede ser estilizada individualmente mediante propiedades CSS.



BOX MODEL: DESCRIPCIÓN DE LOS COMPONENTES

1. **Contenido (content):** Es el área donde se muestra el contenido del elemento HTML. Por ejemplo, en un elemento de texto, el contenido sería el texto mismo.
2. **Relleno (padding):** Es el espacio entre el contenido y el borde. El relleno se puede utilizar para agregar espacio adicional dentro de la caja del elemento, o para aumentar el tamaño visual del elemento sin cambiar su contenido.
3. **Borde (border):** Es la línea que rodea el elemento HTML. El borde se utiliza para crear una separación visual entre el contenido del elemento y su entorno, y puede ser estilizado con diferentes colores, estilos y grosores.
4. **Margen (margin):** Es el espacio entre el borde y el siguiente elemento HTML en la página. El margen se utiliza para crear un espacio adicional entre los elementos y para separar visualmente el contenido del elemento de su entorno.

Nota: Para saber más sobre Box Model puedes ingresar [aquí](#).

CSS: ¿QUÉ ES?

CSS significa Hojas de Estilo en Cascada (del inglés Cascading Style Sheets) y es el lenguaje de estilos utilizado para describir la presentación de documentos HTML.

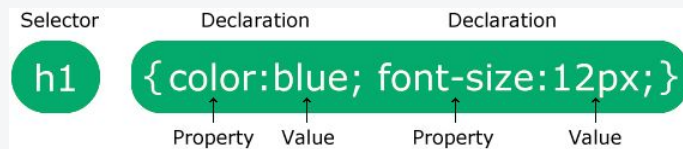
CSS no es un lenguaje de programación en el sentido tradicional, es decir, no puede realizar operaciones matemáticas o lógicas complejas.

- CSS es el lenguaje que usamos para diseñar un documento HTML.
- CSS describe cómo deben mostrarse los elementos HTML.

Nota: Para saber más sobre CSS puedes ingresar [aquí](#).

SINTAXIS CSS

Una regla CSS consta de un selector y un bloque de declaración.



- El selector apunta al elemento HTML al que queremos aplicar estilo.
- El bloque de declaración contiene una o más declaraciones separadas por punto y coma.
- Cada declaración incluye un nombre de propiedad CSS y un valor, separados por dos puntos.
- Varias declaraciones CSS se separan con punto y coma, y los bloques de declaración están rodeados por llaves.

CÓMO AGREGAR CSS

Hay tres formas de insertar una hoja de estilo:

- CSS en línea.
- CSS interno.
- CSS externo.

EJEMPLO CSS APLICADO A UN ELEMENTO HTML

En el ejemplo de abajo, todos los elementos `<p>` estarán alineados al centro, con un color de texto rojo.

```
p {  
    color: red;  
    text-align: center;  
}
```

Ejemplo Explicado:

- `p` es un selector en CSS (apunta al elemento HTML al que se desea aplicar el estilo, en este caso el elemento `<p>`).
- `color` es una propiedad, y `red` es el valor de la propiedad.
- `text-align` es una propiedad, y `center` es el valor de la propiedad.

SELECTORES

Un selector de CSS selecciona los elementos HTML a los que se desea aplicar el estilo.

- Los selectores de CSS se utilizan para “encontrar” (o seleccionar) los elementos HTML a los que se desea aplicar el estilo.

SELECTORES (CONT.)

Podemos dividir los selectores de CSS en cinco categorías:

- Selectores simples (la selección de los elementos se basa en el nombre del elemento, id, clase).
- Selectores combinadores (la selección de los elementos se basa en función de una relación específica entre ellos).
- Selectores de pseudoclase (la selección de los elementos se basa en función de un determinado estado).
- Selectores de pseudoelementos (la selección aplica a una parte de un elemento).
- Selectores de atributos (la selección se basa en función de un atributo o valor de atributo).

SELECTORES: ELEMENTOS BASADOS EN EL NOMBRE DEL ELEMENTO

A continuación un ejemplo de los selectores de CSS más básicos.

Selectores simples basados en el nombre del elemento.

```
p {  
  color: red;  
  text-align: center;  
}
```

En el ejemplo de arriba, todos los elementos `<p>` en la página estarán alineados al centro, con un color de texto rojo.

SELECTORES: ELEMENTOS BASADOS EN EL ID DEL ELEMENTO

Selectores simples basados en el id del elemento.

```
#parrafo {  
    color: red;  
    text-align: center;  
}
```

El selector de id usa el atributo id de un elemento HTML para seleccionar un elemento específico.

La identificación de un elemento es única dentro de una página, por lo que el selector de identificación se usa para seleccionar un elemento único.

Para seleccionar un elemento con un id específico es necesario escribir un carácter hash (#), seguido del nombre de la clase.

SELECTORES: ELEMENTOS BASADOS EN LA CLASE DEL ELEMENTO

Selectores simples basados en la clase del elemento.

```
.parrafo {  
  color: red;  
  text-align: center;  
}
```

El selector de clase selecciona elementos HTML con un atributo de clase específico.

Para seleccionar elementos con una clase específica es necesario escribir un carácter de punto (.), seguido del nombre de la clase.

Se debe usar el carácter hash (#), seguido de la identificación del elemento.

SELECTORES: ELEMENTOS BASADOS EN LA CLASE DEL ELEMENTO (CONT.)

También es posible especificar únicamente aquellos elementos HTML que se vean afectados por una clase.

```
p.parrafos_negrita {  
  color: red;  
  font-weight: bold;  
}
```

El selector de clase selecciona elementos HTML con un atributo de clase específico.

Para seleccionar elementos con una clase específica es necesario escribir un carácter de punto (.), seguido del nombre de la clase.

SELECTORES: ELEMENTOS BASADOS EN LA CLASE DEL ELEMENTO (CONT.)

Los elementos HTML también pueden referirse a más de una clase.

```
p.parrafos_negrita {  
    font-weight: bold;  
}
```

```
p.parrafos_cursiva {  
    font-style: italic;  
}
```

En el ejemplo de arriba, el elemento `<p>` tomará el estilo de acuerdo con las clases `class="parrafo_negrita"` y `class="parrafo_cursiva"`.

SELECTORES: SELECTOR UNIVERSAL

El selector universal (*) selecciona todos los elementos HTML de la página.

El selector universal CSS (*) coincide con elementos de cualquier tipo.

```
*{  
  font-color: green;  
  font-size: 15px;  
  font-weight: bold;  
}
```

En el ejemplo de arriba, todos los elementos HTML tomarán el mismo estilo.

SELECTORES: SELECTORES AGRUPADOS

El selector de agrupación selecciona todos los elementos HTML con las mismas definiciones de estilo.

```
h1, h2, p{  
  font-color: green;  
  font-size: 15px;  
  font-weight: bold;  
}
```

En el ejemplo de arriba, los elementos `<h1>`, `<h2>` y `<p>` tomarán el mismo estilo.

MOMENTO DE EJERCITAR LO APRENDIDO.



¡Importante!

Las herramientas que usaremos son:

- . Editor VSC (Visual Studio Code).
- . Extensión Live Server.



EJERCICIO PARA VALIENTES

- 1) En el editor de código (VSC) continuar trabajando en las páginas y estructura de directorios ya creados:
 - **index.html** (archivo)
 - **listado.html** (archivo)
 - **contacto.html** (archivo)
 - **estilos** (carpeta)
 - **imagenes** (carpeta)
- 2) Agregar un div que contenga el título de una noticia tecnológica, una imagen relacionada a dicho título y un párrafo con la noticia propiamente dicha. Usando CSS interno aplicar a todos los elementos HTML lo siguiente: relleno (padding), margen (margin) y a la imagen también agregarle un borde (border).

Importante: La imagen debe estar alojada en el directorio de imágenes.

EJEMPLO DE CÓMO DEBERÍA QUEDAR EL EJERCICIO

Cómo funciona una Inteligencia Artificial: lo que hay que saber



Una inteligencia artificial es un sistema informático diseñado para simular la inteligencia humana y realizar tareas que normalmente requieren de la intervención humana. La mayoría se basa en algoritmos de aprendizaje automático y técnicas de procesamiento de datos para aprender y mejorar su rendimiento a medida que procesan más información.

BIBLIOGRAFÍA Y ENLACES ÚTILES

Bibliografía utilizada:

- [MDN Web Docs \(mozilla.org\)](https://developer.mozilla.org/)
- [W3Schools Online Web Tutorials](https://www.w3schools.com/)

Herramientas de soporte utilizadas:

- [Visual Studio Code - Code Editing. Redefined](https://code.visualstudio.com/)
- [Live Server - Visual Studio Marketplace](https://marketplace.visualstudio.com/items?itemName=live-server)