



Recibe una cálida:

¡Bienvenida!

Te estábamos esperando 😊 

➤ Aplicación de estilos y responsividad

Manejando Hojas de Estilo

Plan formativo: Desarrollo de Aplicaciones Full Stack Java Trainee V2.0

HOJA DE RUTA

¿Cuáles **skill** conforman el programa?



REPASO CLASE ANTERIOR


En la clase anterior trabajamos :

- ✓ Crear una estructura básica de formulario
- ✓ Comprender el uso de div
- ✓ Entender el uso y beneficios de span

LEARNING PATHWAY

Nº2 . ¿Sobre qué temas trabajaremos?

2.

Start! 

FUNDAMENTOS DE
DESARROLLO FRONT-END

Aplicación de estilos y
responsividad

Actividad N° 1: Incorporando CSS

En esta lección, aprenderemos sobre la importancia de las hojas de estilo CSS en el diseño y presentación de sitios web, así como cómo aplicar estilos a elementos específicos mediante selectores y reglas CSS para crear sitios visualmente atractivos y funcionales. Estos conocimientos son cruciales para construir sitios web efectivos y atractivos.

OBJETIVOS DE APRENDIZAJE

¿Qué aprenderemos?



Poder crear una hoja de estilos CSS para trabajar el diseño web



Aprender como utilizar selectores y reglas de CSS para aplicar estilos



Comprender la diferencia entre: Estilos en línea, embebidos, archivos externos





Rompehielo 🥶

En la escala del pato, cómo te sientes hoy?



› Qué es CSS, fundamentos y utilidad



Qué es CSS, fundamentos y utilidad

CSS, o Cascading Style Sheets (Hojas de Estilo en Cascada), es un lenguaje de estilo utilizado para controlar la presentación y el diseño de páginas web.

Su principal función es definir cómo se deben mostrar los elementos HTML en términos de diseño, formato, colores y disposición en una página web.





Qué es CSS, fundamentos y utilidad

Se utiliza para definir cómo se deben mostrar los elementos HTML en términos de diseño, formato, colores y disposición. CSS se basa en selectores, propiedades y valores para aplicar estilos a los elementos HTML, permitiendo una separación clara entre contenido y diseño.

Esto resulta en sitios web visualmente atractivos y bien estructurados.



Qué es CSS, fundamentos y utilidad

Puntos claves:

- ✕ **Lenguaje de Estilo:** CSS es un lenguaje de estilo utilizado para controlar la apariencia visual de una página web.

Definición de Estilos: Con CSS, puedes definir estilos para elementos HTML como colores, fuentes, tamaños de texto, márgenes y diseños.

Puntos claves:

Separación de Contenido y Presentación: CSS permite separar el contenido (HTML) de su presentación visual, lo que facilita la actualización y el mantenimiento del sitio web.

✕ **Flexibilidad en el Diseño:** CSS ofrece flexibilidad para crear diseños personalizados y atractivos, mejorando la experiencia del usuario.



Qué es CSS, fundamentos y utilidad

Puntos claves:

- ✕ **Reglas CSS:** Las reglas CSS consisten en un selector que elige elementos HTML y un conjunto de propiedades y valores que definen su estilo.

Selectores: Los selectores en CSS, como clases o identificadores, determinan a qué elementos se aplicarán las reglas de estilo.

Puntos claves:

Herencia y Cascada: CSS sigue un modelo de cascada, lo que significa que los estilos pueden heredarse de elementos padres a elementos hijos y pueden ser sobrescritos según la especificidad.

✕ **Enlace Externo:** En la práctica, se vinculan archivos CSS externos a documentos HTML utilizando la etiqueta `<link>` en la sección `<head>` del HTML.



› Estilos en línea, embebidos, archivos externos



Estilos en línea, embebidos, archivos externos



En CSS, existen tres formas principales de incluir estilos en un documento HTML: estilos en línea, estilos embebidos y estilos en archivos externos.

Los estilos en línea, embebidos y archivos externos son tres formas diferentes de aplicar estilos CSS a una página web, cada una con sus ventajas y desventajas.





Estilos en línea, embebidos, archivos externos

Estilos en línea (Inline Styles):

Definición: Los estilos en línea se aplican directamente a un elemento HTML utilizando el atributo style.

Esto significa que los estilos son específicos para ese elemento y se aplican sólo a él.



```
<p style="color: red; font-size: 16px;">Este es un párrafo con estilos en línea.</p>
```

Ventajas: Son simples de implementar y útiles para estilos específicos de un solo elemento.

Desventajas: Pueden resultar difíciles de mantener en páginas con muchos elementos, ya que los estilos están dispersos en todo el código HTML.



Estilos en línea, embebidos, archivos externos

Estilos embebidos (Embedded Styles):

Definición: Los estilos embebidos se definen dentro del elemento <style> en la sección <head> de un documento HTML. Los estilos se aplican a los elementos que coincidan con los selectores definidos en ese documento.



```
<!DOCTYPE html>
<html>
<head>
<style>
  p {
    color: blue;
    font-size: 18px;
  }
</style>
</head>
<body>
  <p>Este es un párrafo con estilos embebidos.</p>
</body>
</html>
```




Estilos en línea, embebidos, archivos externos

Estilos embebidos (Embedded Styles):

Ventajas: Permite definir estilos para múltiples elementos en una página y mantenerlos en un solo lugar.

Desventajas: Aún puede resultar complejo de mantener en páginas grandes, y los estilos están vinculados a una página específica.





Estilos en línea, embebidos, archivos externos

Archivos externos (External Stylesheets):

Definición: Los archivos externos CSS se crean como archivos separados con extensión .css y se vinculan a un documento HTML utilizando la etiqueta <link> en la sección <head>.

Los estilos se aplican a todos los elementos que coincidan con los selectores definidos en el archivo CSS. 

Ejemplo: (archivo estilos.css)

```
p {  
  color: green;  
  font-size: 20px;  
}
```



Estilos en línea, embebidos, archivos externos

Ejemplo: (archivo index.html)

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="styles.css">
  </head>
  <body>
    <p>Este es un párrafo con estilos desde un archivo externo de CSS.</p>
  </body>
</html>
```



Ventajas: Permite mantener los estilos de manera separada de la estructura HTML, lo que facilita la mantenibilidad y la reutilización de estilos en múltiples páginas.



Desventajas: Puede requerir una solicitud adicional al servidor para cargar el archivo CSS, aunque este se almacena en caché para futuras visitas.

➤ Referencias y selectores, por clase, por id



Referencias y selectores, por clase, por id



En CSS, los selectores son patrones que se utilizan para seleccionar elementos HTML a los cuales se aplicarán estilos.

Dos de los tipos más comunes de selectores son por clase y por ID. Aunque también podemos seleccionar por etiqueta, la elección entre selectores por clase y por ID depende de las necesidades de diseño y estructura.



Las clases suelen ser más flexibles y reutilizables, mientras que los IDs son más específicos y deben ser únicos.





Referencias y selectores, por clase, por id

Selección por etiqueta (selector de tipo):

Definición: es una forma de seleccionar todos los elementos HTML de un tipo específico y aplicar estilos a esos elementos. Este tipo de selector utiliza el nombre de la etiqueta HTML como criterio de selección.



El selector por etiqueta es útil cuando deseas aplicar un estilo general a un tipo específico de elemento en todo tu sitio web.

Sin embargo, es importante tener en cuenta que afectará a todos los elementos de esa etiqueta en la página, por lo que debes usarlo con precaución y considerar si es necesario o si es mejor utilizar selectores más específicos, como selectores por clase o por ID, para elementos particulares.





Referencias y selectores, por clase, por id

(Hoja css)

```
p{  
  color: blue;  
  font-weight: bold;  
}
```



(Hoja html)

```
<p>Este es un párrafo destacado.</p>  
<p>Este es otro párrafo destacado.</p>
```





Referencias y selectores, por clase, por id

Selección por Clase (Class Selectors):

Definición: Los selectores por clase permiten aplicar estilos a un grupo de elementos que comparten la misma clase.

Para seleccionar elementos por clase, se debe utilizar el símbolo de punto (.) seguido del nombre de la clase.

Por ejemplo, si tienes elementos con la clase ".destacado", el selector sería ".destacado".

Asegúrate de incluir el punto antes del nombre de la clase. “.puntoyNombreDelaClase”





Referencias y selectores, por clase, por id

(Hoja css)

```
.destacado{  
  color: blue;  
  font-weight: bold;  
}
```



(Hoja html)

```
<p class="destacado">Este es un párrafo destacado.</p>  
<p class="destacado">Este es otro párrafo destacado.</p>
```






Referencias y selectores, por clase, por id

Selección por ID (ID Selectors):

Definición: Los selectores por ID permiten aplicar estilos a un elemento HTML específico que tenga un atributo "id" único. Se definen con un numeral (#) seguido del nombre del ID.

Para seleccionar un elemento específico por su ID, se debe utilizar el símbolo de almohadilla (#) seguido del nombre del ID. 

Por ejemplo, si tienes un elemento con el ID "miElemento", el selector sería "#miElemento".

Asegúrate de incluir el almohadilla antes del nombre del ID.





Referencias y selectores, por clase, por id

(Hoja css)

```
.miElemento {  
  color: blue;  
  font-size: 16px;  
}
```



(Hoja html)

```
<h1 id="miElemento">Este es el título principal</h1>
```



LIVE CODING

Ejemplo en vivo

Migración de Estilos en Línea a Estilos Externos:

Contexto: Estás trabajando en la optimización de un sitio web y te has dado cuenta de que tiene demasiados estilos en línea (in-line) en su HTML. Esto dificulta el mantenimiento y la organización del código. Tu tarea es migrar estos estilos en línea a un archivo CSS externo para mejorar la estructura y el rendimiento del sitio.

Objetivo: Durante este live coding, mostrarás cómo migrar estilos en línea a un archivo CSS externo. Explicarás por qué es importante hacerlo y cómo esta práctica mejora la mantenibilidad y la organización del código.

Tiempo: 10 minutos

LIVE CODING

Ejemplo en vivo

```
<div style="color: red; font-size: 20px;">Este es un texto rojo con un tamaño de fuente de 20px.</div>
<p style="background-color: yellow; text-align: center;">Este es un párrafo con fondo amarillo y texto centrado.</p>
<a href="#" style="text-decoration: underline; background-color: lightblue;">Este es un enlace</a>
<h1 style="background-color: gray; color: white;">Este es un encabezado con fondo gris y texto blanco.</h1>
<p style="color: green; background-color: yellow;">Este es un párrafo con texto verde sobre fondo amarillo.</p>
<span style="color: red; background-color: pink;">Este es un texto rojo sobre fondo rosa.</span>
<div style="color: orange; background-color: gray;">Este es un texto naranja sobre fondo gris.</div>
<p style="color: white; background-color: black;">Este es un párrafo con texto blanco sobre fondo negro.</p>
```



LIVE CODING

Ejemplo en vivo

Solución Paso a Paso:

Paso 1: Identificación de Estilos en Línea

Abre el archivo HTML del sitio web y transcribir los estilos en línea de la página anterior

Ejemplo: <div style="color: red;">.

Paso 2: Creación de un Archivo CSS Externo

Crea un nuevo archivo CSS en la misma carpeta que tu archivo HTML.

Puedes llamarlo, por ejemplo, styles.css.

LIVE CODING

Ejemplo en vivo

Solución Paso a Paso:

Paso 3: Extracción de Estilos en Línea

Copia los estilos en línea que encontraste en el paso 1 y pégalos en el archivo CSS externo (styles.css).

Los estilos en línea se deben traducir a reglas CSS válidas.

Por ejemplo, si tienes `<div style="color: red;">`, en el archivo CSS sería `div { color: red; }`.

Paso 4: Vinculación del Archivo CSS

Abre el archivo HTML y elimina los estilos en línea que copiaste al archivo CSS.

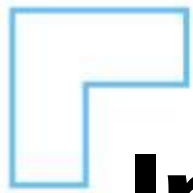
Vincula el archivo CSS externo a tu HTML utilizando la etiqueta `<link>` en el `<head>` del documento HTML. Debe verse así: `<link rel="stylesheet" href="styles.css">`.





Ejercicio

Incorporando CSS



Incorporando CSS

Breve descripción

Contexto: 🙌

Eres un desarrollador web contratado para mejorar el aspecto y la estructura de un sitio web existente. El cliente desea que el sitio sea más atractivo visualmente y siga las mejores prácticas de desarrollo web.



Consigna: 🛠️

Mejorar los estilos y la estructura del sitio web utilizando HTML y CSS. Utilizarás selectores CSS para aplicar estilos basados en ID y clases en el HTML existente.

Tiempo🕒: 10 minutos



Incorporando CSS

Breve descripción

Paso a paso:

Paso 1: Preparación

- Abre el proyecto web anterior existente y revisa su estructura actual.
- Crea un archivo CSS separado si aún no lo tienes y vincúlalo al archivo HTML.
- Asegúrate de que todas las imágenes y recursos necesarios estén disponibles y funcionando correctamente.

Paso 2: Mejora de la Estructura HTML

- Revisa el HTML y asegúrate de que esté bien estructurado con etiquetas semánticas como `<header>`, `<nav>`, `<section>`, `<article>`, `<footer>`, etc.
- Agrega IDs o clases a los elementos HTML que desees estilizar de manera específica, a partir de lo entendido en clase, dale nombres representativos a cada semántica.

○

¿Alguna consulta?

+



RESUMEN

¿Qué logramos en esta clase?

- ✓ **Comprender el uso de CSS para incorporar y trabajar el diseño web**
- ✓ **Aprender a diferenciar y utilizar los selectores como ID y Class**
- ✓ **Comprender y utilizar estilos en línea, embebidos, archivos externos**



#WorkingTime

Continuemos ejercitando

¡Antes de cerrar la clase! Te invitamos a: 📖 📖 📖

1. Repasar nuevamente la grabación de esta clase
2. Revisar el material compartido en la plataforma de Moodle (lo que se vio en clase y algún ejercicio adicional)
 - a. Material 1: *Lección 3: Aplicación de estilos y responsividad* página: 1-9
3. Traer al próximo encuentro, todas tus dudas y consultas para verlas antes de iniciar nuevo tema.

Momento: ✚

Time-out!

🕒 5 min.

