

Cuarta Sesión: Introducción a CSS, Selectores y Bootstrap – Estilos en Páginas Web

Duración: 3 horas

Objetivo de la clase:

Al finalizar esta sesión, los estudiantes comprenderán la estructura básica de CSS, sabrán cómo aplicar selectores de clase e ID desde la perspectiva de HTML y CSS, y recibirán una introducción a **Bootstrap** para mejorar el diseño y la responsividad de sus páginas web.

1. Revisión de la Sesión Anterior (15 minutos)

Vamos a revisar brevemente lo que aprendimos en la última clase sobre **formularios** y **JavaScript**. En esta sesión, daremos el siguiente paso: aplicar estilos a nuestras páginas con **CSS** y aprender a usar **Bootstrap**.

2. Estructura Básica de CSS (30 minutos)

2.1. ¿Qué es CSS?

CSS (Cascading Style Sheets) es el lenguaje que usamos para controlar la presentación de las páginas web. Mientras que **HTML** organiza el contenido, CSS se encarga de su apariencia: colores, tipografías, márgenes, espaciados, alineaciones, etc.

La estructura básica de una regla CSS es:

```
selector {  
    propiedad: valor;  
}
```

- **Selector:** Define a qué elementos de la página HTML se le aplicarán los estilos.
- **Propiedad y valor:** Especifican qué estilo se aplicará. Por ejemplo, color: blue; cambiaría el color del texto a azul.

2.2. Ejemplo básico de CSS

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8">
    <title>Página con CSS</title>
    <style>
      body {
        background-color: lightgray;
      }
      h1 {
        color: blue;
        font-size: 24px;
      }
      p {
        color: black;
        font-family: Arial, sans-serif;
      }
    </style>
  </head>
  <body>
    <h1>Bienvenidos a mi sitio</h1>
    <p>Este es un párrafo con estilo.</p>
  </body>
</html>
```

En este ejemplo:

- **background-color:** Cambia el color de fondo del elemento seleccionado.
- **color:** Cambia el color del texto del elemento seleccionado.
- **font-size:** Cambia el tamaño del texto.

3. Selectores de Clase e ID desde HTML (45 minutos)

3.1. Clase en HTML

El atributo `class` permite aplicar un conjunto de estilos a varios elementos. Una clase puede repetirse en múltiples elementos en una página HTML, lo que permite agruparlos y aplicar estilos comunes.

Ejemplo:

```
<p class="resaltado">Este es un párrafo con la clase  
"resaltado".</p>  
<p class="resaltado">Este es otro párrafo con la misma clase.</p>
```

En este caso, ambos párrafos usan la clase `resaltado`. Luego podemos definir estilos específicos para esta clase en CSS.

3.2. ID en HTML

El atributo `id` identifica un elemento único en la página. A diferencia de las clases, un **ID** no se debe repetir; es exclusivo para cada elemento.

Ejemplo:

```
<h1 id="titulo">Este es el título principal con un ID único.</h1>
```

Aquí estamos asignando el ID `titulo` al encabezado `<h1>`, lo que nos permitirá aplicarle un estilo único en CSS.

3. Tipos de Estilos en CSS (45 minutos)

Existen tres formas de agregar CSS a una página web:

3.1. CSS en línea (Inline CSS)

Se usa directamente dentro de las etiquetas HTML mediante el atributo style. No es recomendable usar CSS en línea para proyectos grandes, ya que hace el código difícil de mantener.

Ejemplo:

```
<p style="color: red;">Este es un párrafo con estilo en línea.</p>
```

3.2. CSS Interno (Internal CSS)

Se incluye dentro de la página HTML en la sección <head>, dentro de la etiqueta <style>. Esto es útil cuando queremos aplicar estilos solo a esa página.

Ejemplo:

```
<head>
  <style>
    h1 {
      color: green;
    }
  </style>
</head>
```

3.3. CSS Externo (External CSS)

Este es el método más recomendado, ya que permite escribir los estilos en un archivo separado (.css). La hoja de estilos externa se enlaza a la página HTML mediante la etiqueta <link> en el <head>.

Ejemplo:

```
<head>

  <link rel="stylesheet" href="estilos.css">

</head>
```

Contenido del archivo estilos.css:

```
h1 {
  color: purple;
  font-size: 28px;
}
p {
  color: gray;
}
```

4. Selectores de Clase e ID desde CSS (30 minutos)

Una vez que hemos definido las **clases** e **IDs** en HTML, podemos aplicarles estilos desde **CSS**. Estos selectores nos permiten apuntar a los elementos de la página a los que queremos aplicar estilos.

4.1. Selector de Clase en CSS

Para aplicar un estilo a todos los elementos que tienen una clase en HTML, utilizamos un selector de clase en CSS. Esto se hace colocando un punto (.) seguido del nombre de la clase.

Ejemplo en CSS:

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

Este código indica que cualquier elemento con la clase resaltado tendrá el texto en color rojo y en negrita.

4.2. Selector de ID en CSS

Los **IDs** en CSS se seleccionan con una almohadilla (#) seguida del nombre del ID. Los estilos aplicados a un ID solo afectarán al elemento que tenga ese ID en la página.

Ejemplo en CSS:

```
#titulo {  
    font-size: 30px;  
    color: green;  
}
```

Esto aplica un tamaño de texto de 30 píxeles y un color verde al elemento que tenga el ID titulo.

Diferencias entre clase e ID

- **Clase:** Se puede usar en múltiples elementos. Ideal para aplicar el mismo estilo a diferentes partes de una página.
- **ID:** Debe ser único para cada elemento. Se usa para aplicar un estilo único y específico a un solo elemento.

Propiedades Básicas de CSS (45 minutos)

Veamos ahora algunas de las propiedades más comunes de CSS, que puedes usar para dar estilo a tus páginas web.

4.1. Colores

La propiedad color define el color del texto, mientras que background-color define el color de fondo de un elemento.

```
h1 {  
    color: red;  
}  
  
body {  
    background-color: lightblue;  
}
```

4.2. Fuentes y Tamaño de Texto

La propiedad `font-family` define el tipo de letra que se va a usar, y `font-size` controla el tamaño del texto.

```
p {  
    font-family: "Arial", sans-serif;  
    font-size: 16px;  
}
```

4.3. Márgenes y Padding

- **margin:** Controla el espacio fuera de un elemento.
- **padding:** Controla el espacio dentro del elemento, entre el contenido y el borde.

```
h1 {  
    margin: 20px;  
    padding: 10px;  
}
```

4.4. Bordes

Puedes agregar bordes a cualquier elemento con la propiedad border.

```
p {  
  border: 2px solid black;  
}
```

5. Bootstrap – Introducción Básica (30 minutos)

Ahora que comprendemos cómo funciona CSS, vamos a ver cómo **Bootstrap** nos permite aplicar estilos rápidamente y crear páginas responsivas.

5.1. ¿Qué es Bootstrap?

Bootstrap es un framework CSS que facilita el diseño de páginas web. Proporciona estilos predefinidos y clases que podemos usar para crear sitios web atractivos sin necesidad de escribir CSS personalizado.

5.2. Incluir Bootstrap en tu página

Para usar Bootstrap, solo tenemos que enlazar su archivo CSS en el <head> de nuestro documento HTML:

```
<head>  
  <link  
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootst  
rap.min.css" rel="stylesheet">  
</head>
```

5.3. Usando clases de Bootstrap

Con Bootstrap, puedes utilizar clases como container, text-center, o btn para aplicar estilos y hacer que tu página sea **responsiva** (adaptada a diferentes tamaños de pantalla).

Ejemplo básico con Bootstrap:


```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8">
    <title>Página con Bootstrap</title>
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootst
rap.min.css" rel="stylesheet">
  </head>
  <body>
    <div class="container">
      <h1 class="text-center text-primary">Bienvenidos a mi
página</h1>
      <p class="text-muted">Este es un párrafo con estilos de
Bootstrap.</p>
      <button class="btn btn-primary">Haz clic aquí</button>
    </div>
  </body>
</html>
```

Aquí hemos utilizado las clases de Bootstrap:

- **container:** Centra el contenido y agrega márgenes automáticos.
- **text-center:** Centra el texto horizontalmente.
- **text-primary:** Aplica el color primario de Bootstrap.
- **btn btn-primary:** Aplica estilo de botón con el color primario de Bootstrap.

6. Ejercicio Práctico: Clases, IDs y Bootstrap (45 minutos)

Es hora de poner en práctica lo que hemos aprendido creando una página que combine **clases**, **IDs** y **Bootstrap**.

1. Crea un archivo HTML llamado index.html.
2. Aplica estilos personalizados usando clases e IDs desde CSS.
3. Usa Bootstrap para mejorar la presentación y hacer la página responsiva.

Ejemplo de estructura:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8">
    <title>Página con CSS y Bootstrap</title>
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootst
rap.min.css" rel="stylesheet">
    <style>
      .resaltado {
        color: red;
        font-weight: bold;
      }
      #titulo {
        font-size: 30px;
        color: green;
      }
    </style>
  </head>
```

```
<body>

  <div class="container">

    <h1 id="titulo" class="text-center">Bienvenidos a mi
página</h1>

    <p class="resaltado">Este párrafo tiene una clase
personalizada y está resaltado.</p>

    <button class="btn btn-success">Botón Verde</button>

  </div>

</body>

</html>
```

7. Cierre y Preguntas (15 minutos)

Hoy hemos aprendido cómo aplicar **clases** e **IDs** desde HTML y cómo usar CSS para estilizar esos elementos. También vimos cómo utilizar **Bootstrap** para mejorar rápidamente el diseño de nuestras páginas.