



ByteNova

Guia completa de HTML y CSS

Material descargable para estudiantes y principiantes

HTML

Estructura

CSS

Diseño

WEB

Resultado

bytenova - tecnologia explicada de forma clara

Esta guia fue creada para que estudiantes, jovenes y principiantes puedan estudiar HTML y CSS de forma clara, con ejemplos, diagramas y consejos practicos.

1. Que aprenderas en esta guia

HTML y CSS son la base de casi todos los sitios web. Aunque al principio pueden parecer complicados, en realidad se entienden mejor cuando separas dos ideas: HTML organiza el contenido y CSS le da apariencia visual.

En esta guia aprenderas los conceptos principales, la estructura de una pagina, etiquetas comunes, estilos CSS, buenas practicas y errores frecuentes que conviene evitar.

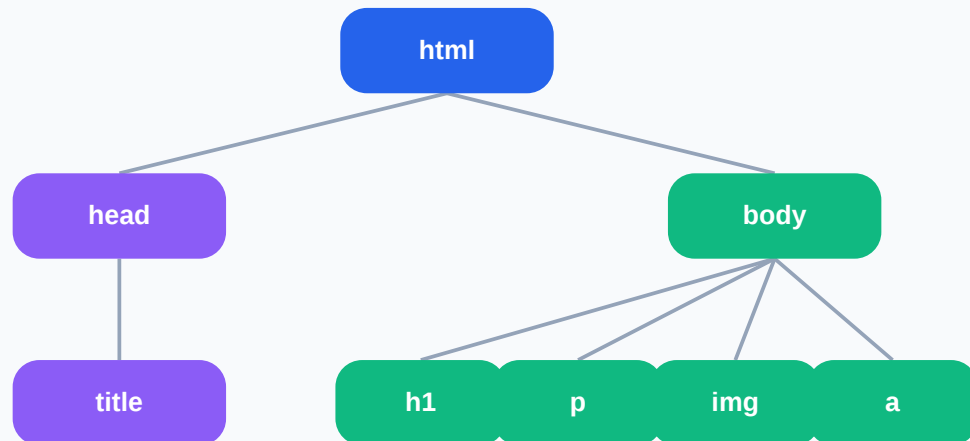
- Comprender la diferencia entre estructura y diseno.
- Identificar las partes principales de un documento HTML.
- Aplicar estilos basicos con CSS.
- Organizar archivos de un proyecto web sencillo.
- Crear una base de estudio para seguir aprendiendo desarrollo web.

Consejo ByteNova: no intentes memorizar todo al inicio. Primero entiende la idea general, luego practica escribiendo codigo poco a poco.

2. HTML: la estructura del contenido

HTML significa HyperText Markup Language. Es un lenguaje de marcado que usa etiquetas para indicar que tipo de contenido aparece en la pagina.

Ejemplo visual: estructura basica de una pagina HTML



head guarda informacion para el navegador. body contiene lo que ve el visitante.

Diagrama corregido: los elementos se muestran separados para evitar cruces y solapamientos visuales.

Partes principales

- **<html>**: contiene todo el documento.
- **<head>**: guarda informacion para el navegador, como titulo, metadatos y enlaces a CSS.
- **<body>**: contiene lo que el visitante ve en pantalla.
- **<h1>**, **<p>**, ****, **<a>**: representan titulos, textos, imagenes y enlaces.

3. Estructura minima de una pagina HTML

Una pagina HTML normalmente empieza con una estructura base. Esta estructura ayuda al navegador a interpretar correctamente el documento.

```
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Mi primera pagina</title>
<link rel="stylesheet" href="css/style.css">
</head>
<body>
<h1>Hola mundo</h1>
<p>Esta es mi primera pagina web.</p>
</body>
</html>
```

Que significa cada parte

- **DOCTYPE** indica que el documento usa HTML moderno.
- **lang="es"** ayuda a indicar que el idioma principal es espanol.
- **meta charset** permite mostrar correctamente tildes y caracteres especiales.
- **viewport** ayuda a que la pagina se adapte a celulares.
- **link rel="stylesheet"** conecta el archivo CSS.

4. Etiquetas HTML mas usadas

Etiqueta	Uso principal	Ejemplo
h1 - h6	Titulos y subtitulos	<code><h1>Bienvenido</h1></code>
p	Parrafos de texto	<code><p>Texto de ejemplo</p></code>
a	Enlaces	<code>Leer</code>
img	Imagenes	<code></code>
section	Agrupar secciones	<code><section>...</section></code>
div	Contenedor generico	<code><div class="tarjeta">...</div></code>
form	Formularios	<code><form>...</form></code>

Una etiqueta puede tener atributos. Por ejemplo, un enlace usa href para indicar a donde dirige, y una imagen usa src para indicar la ruta del archivo.

5. CSS: el diseño visual

CSS significa Cascading Style Sheets. Es el lenguaje que controla la apariencia de una página web. Con CSS puedes cambiar colores, fondos, tamaños, espaciados, bordes, sombras y animaciones.

Modelo de caja en CSS



Todo elemento HTML se puede imaginar como una caja con contenido, relleno, borde y margen.

Ejemplo básico

```
body {  
  background: #f8fafc;  
  font-family: Arial, sans-serif;  
}  
  
h1 {  
  color: #2563eb;  
  font-size: 40px;  
}  
  
.tarjeta {  
  background: white;  
  padding: 30px;  
  border-radius: 15px;  
  box-shadow: 0 10px 25px rgba(0,0,0,0.08);  
}
```

6. HTML y CSS juntos

HTML y CSS se complementan. HTML dice que elementos existen, y CSS dice como se ven esos elementos. Esta separacion permite que el proyecto sea mas limpio y facil de mantener.

HTML + CSS trabajan juntos



Ejemplo combinado

```
<!-- HTML -->
<a href="#biblioteca" class="btn-principal">Explorar Biblioteca</a>

/* CSS */
.btn-principal {
background: #2563eb;
color: white;
padding: 15px 30px;
border-radius: 8px;
text-decoration: none;
}
```

7. Organizacion de carpetas

Un proyecto web se entiende mejor cuando sus archivos estan organizados. Esta estructura separa contenido, estilos, scripts, imagenes, documentos y paginas internas.

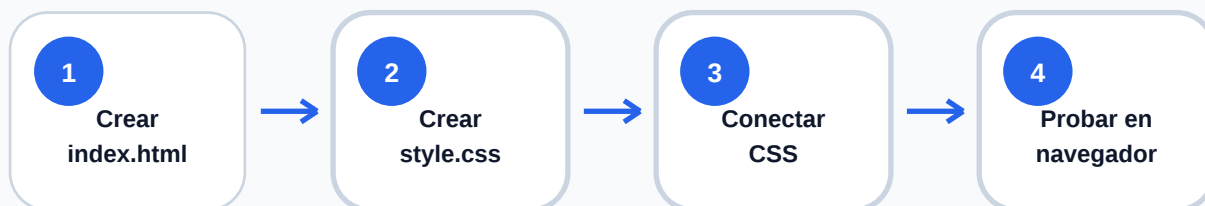
```
ByteNova/  
  index.html  
  css/  
    style.css  
  js/  
    script.js  
  images/  
    hero.png  
    favicon.svg  
  docs/  
    guia-html-css.pdf  
  pages/  
    sobre.html  
    privacidad.html  
    terminos.html
```

Por que organizar asi

- Evitas perder archivos importantes.
- Las rutas son mas faciles de comprender.
- El sitio queda listo para crecer con mas paginas.
- Mejora el orden cuando subes el proyecto a servicios como Netlify.

8. Diseno responsive

Una pagina responsive se adapta a diferentes tamanos de pantalla, como computadoras, tablets y celulares. Esto es muy importante porque muchas personas visitan sitios web desde el telefono.



Este flujo simple evita confusiones al empezar un proyecto web.

Media query basica

```
@media(max-width:768px){  
  header{  
    flex-direction:column;  
    padding:20px;  
  }  
  
  .hero{  
    flex-direction:column;  
    text-align:center;  
  }  
  
  .hero-texto,  
  .hero-imagen{  
    width:100%;  
  }  
}
```

9. Buenas practicas para principiantes

Las buenas practicas ayudan a escribir codigo mas claro y facil de corregir. No se trata de hacerlo perfecto al inicio, sino de evitar errores que causan confusion.

- Usa nombres de clases claros, por ejemplo: hero, tarjeta, biblioteca, footer.
- Guarda cada archivo con el nombre correcto y en la carpeta correcta.
- No mezcles todo el CSS sin comentarios. Divide por secciones.
- Revisa las llaves de CSS: cada bloque debe abrir con { y cerrar con }.
- Usa alt en las imagenes para mejorar accesibilidad y SEO.
- Prueba cada enlace despues de crearlo.
- Guarda cambios con Ctrl + S antes de revisar en el navegador.

Tip importante: si algo no cambia en el navegador, presiona Ctrl + F5 para recargar sin usar la version guardada en cache.

10. Errores frecuentes

Error	Por que pasa	Como corregirlo
No carga el CSS	Ruta incorrecta al archivo style.css	Revisar href="css/style.css" o ../css/style.css
Imagen no aparece	Nombre o carpeta incorrecta	Revisar images/hero.png y mayusculas
Boton no se ve bien	La clase no coincide con el CSS	Usar class="btn-explorar"
Modo oscuro falla	Falta el boton btn-modo	Agregar id="btn-modo" o proteger el JS
Enlace pages/pages	Ruta duplicada dentro de pages	Usar sobre.html y no pages/sobre.html

11. Checklist para estudiar HTML y CSS

Usa esta lista para verificar tu avance. Puedes marcar cada punto cuando lo entiendas o cuando lo practiques en tu propio proyecto.

Checklist de estudio

- ☐ Entiendo que HTML crea la estructura.
- ☐ Entiendo que CSS controla el diseño.
- ☐ Se crear un archivo index.html.
- ☐ Se conectar un archivo style.css.
- ☐ Puedo crear títulos, párrafos, enlaces e imágenes.
- ☐ Puedo cambiar colores, espacios y tamaños con CSS.
- ☐ Entiendo como funcionan las rutas de archivos.
- ☐ Puedo revisar errores simples en el navegador.
- ☐ Estoy listo para practicar creando una página sencilla.

12. Proyecto practico recomendado

La mejor forma de aprender es construir. Para practicar, crea una pagina sencilla con un encabezado, una seccion principal, tres tarjetas y un footer.

Reto ByteNova

- Crea una carpeta llamada mi-primer-sitio.
- Dentro crea index.html y css/style.css.
- Agrega un titulo principal, una descripcion y un boton.
- Crea tres tarjetas con temas que te interesen.
- Agrega colores, bordes redondeados y sombras.
- Prueba el resultado en computadora y celular.

Cuando logres completar este reto, ya tendras una base real para seguir aprendiendo JavaScript, disenio responsive, SEO y publicacion web.

ByteNova: la tecnologia del manana, explicada hoy.