

Elemen Semantik pada HTML

Apa itu Semantic HTML?

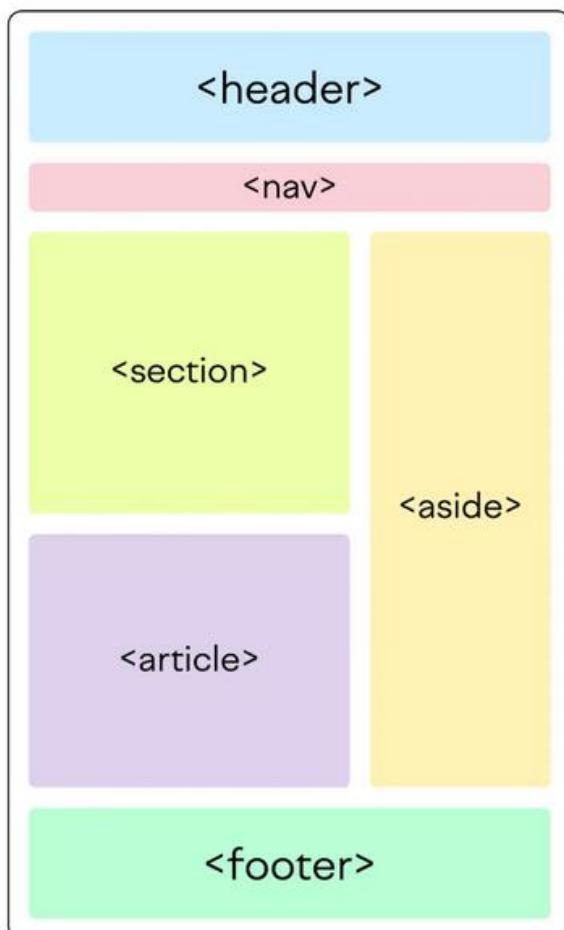
Semantic HTML adalah konsep dalam [pengembangan web](#) yang berfungsi menyediakan makna atau arti lebih dalam kode HTML.

Dalam *semantic* HTML, masing-masing elemen memiliki nama yang mencerminkan arti dari kontennya, bukan hanya presentasinya. *Semantic* membantu mesin pencari dan perangkat pembaca layar (*screen reader*) untuk memahami struktur dan konten dari halaman web dengan lebih maksimal.

Artinya, jika kamu ingin membuat kode yang mudah dipahami, tidak hanya oleh manusia tapi juga oleh mesin, *semantic* HTML adalah solusi yang tepat.

Contoh semantic

Semantic HTML



HTML Semantik vs. HTML Non-Semantik

Tag HTML semantik mendeskripsikan makna konten, sedangkan tag HTML non-semantik bersifat generik dan terutama digunakan untuk tujuan gaya dan tata letak.

Tag HTML non-semantik tidak menyampaikan makna khusus apa pun tentang konten itu sendiri.

Berikut perbandingan visual blok kode HTML semantik dan non-semantik:

Non-Semantic HTML Code	Semantic HTML Code
<pre><!DOCTYPE html> <body> <div> My Website <div> Home About Contact </div> </div> <div> <div>Welcome to My Website</div> </div> <div> <div>This is the main content of my website.</div> </div> <div> &copy; 2025 My Website </div> </body></pre>	<pre><header> <h1>My Website</h1> <nav> Home About Contact </nav> </header> <main> <h2>Welcome to My Website</h2> <p>This is the main content of my website.</p> </main> <footer> <p>&copy; 2025 My Website</p> </footer></pre>

Dengan membandingkan kedua blok kode, kita dapat dengan mudah melihat bahwa kode non-semantik yang hanya menggunakan tag `<div>` dan `` tidak menjelaskan konten atau bagian halaman.

Sedangkan kode HTML semantik memberikan deskripsi yang jelas yang bahkan dapat dipahami oleh orang yang bukan pembuat kode.

Misalnya, kode HTML semantik memberi tahu Anda bahwa “Situs Web Saya” adalah tajuk halaman.

Mengapa Tag HTML Semantik Penting?

Tag HTML semantik penting karena meningkatkan aksesibilitas dan pemahaman halaman Anda bagi manusia dan mesin pencari.

Berikut adalah manfaat utama penggunaan HTML semantik:

- **Pengalaman pengguna dan aksesibilitas yang lebih baik :** Pembaca layar dan alat lainnya dapat menavigasi tata letak semantik secara lebih efektif dan akurat mendeskripsikan konten Anda kepada pengguna
- **Performa SEO yang lebih baik :** Tag semantik dalam HTML membantu perayap web seperti Googlebot mengidentifikasi bagian konten Anda yang relevan secara akurat dari

HTML halaman Anda. Hal ini dapat menghasilkan pengindeksan yang lebih baik dan berpotensi meningkatkan peringkat untuk kata kunci yang relevan.

- **Peluang muncul di hasil kaya meningkat** : Elemen semantik dapat meningkatkan [markup skema](#) —kode yang Anda tambahkan ke halaman untuk membantu Google menampilkan hasil kaya. Hasil kaya berisi informasi tambahan seperti peringkat, harga, atau tanggal acara.
- **Kode yang tahan masa depan** : HTML semantik mengikuti standar web yang membuat kode Anda lebih kompatibel dengan teknologi masa depan
- **Komunikasi yang jelas** : Pengembang, desainer, dan SEO yang bekerja sama dapat dengan mudah memahami tujuan setiap bagian. Hal ini memudahkan pemeliharaan dan pembaruan situs web Anda.

Elemen HTML Semantik Umum

Ada banyak [elemen HTML semantik](#), tetapi mari kita lihat tag semantik yang paling umum digunakan berdasarkan kategori:

Tag HTML Semantik untuk Struktur

Elemen semantik HTML ini menentukan struktur dan tata letak halaman:

- **<header>** : Menunjukkan konten pengantar atau navigasi halaman web atau bagian. Sering kali berisi elemen seperti judul, menu navigasi, bilah pencarian, dan logo.
- **<nav>** : Menentukan blok tautan navigasi. Gunakan untuk navigasi di seluruh situs (misalnya, tautan ke "Beranda", "Tentang Kami", atau "Layanan") atau untuk mengelompokkan tautan ke dalam bagian-bagian tertentu dalam halaman (seperti "Fitur", "Harga", atau "Daftar Isi").
- **<main>** : Mewakili konten utama atau bagian utama halaman web. Gunakan tag ini hanya sekali per halaman.
- **<section>** : Mengelompokkan konten terkait yang memiliki tema atau tujuan yang sama. Misalnya, Anda dapat membagi halaman arahan menjadi beberapa bagian seperti "Fitur", "Manfaat", atau "Kasus Penggunaan".
- **<article>** : Menentukan konten mandiri yang dapat Anda gunakan kembali atau distribusikan secara terpisah, seperti postingan blog, ulasan produk, dan artikel berita.
- **<aside>** : Mewakili konten tambahan atau tangensial yang mendukung atau memberikan konteks tambahan pada konten utama. Beberapa contohnya meliputi bilah sisi, keterangan, atau blok tautan terkait.
- **<footer>** : Menunjukkan footer halaman atau bagian yang biasanya berisi tautan navigasi, informasi kontak, dan informasi hak cipta

Praktik membuat semantic

websaya.html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>web saya</title>
7  </head>
8  <body>
9  <header>
10 |  <h1>Belajar Elemen Semantik di HTML</h1>
11 </header>
12
13 <nav>
14 |  <a href="#">Home</a> |
15 |  <a href="#">About</a> |
16 |  <a href="#">Contact</a>
17 </nav>
18
19 <article>
20 |  <h1>Tutorial Semantik Elemen untuk Pemula</h1>
21 |  <p>Semantik elemen adalah elemen yang memiliki makna dan tujuan.
22 |  | Tujuannya agar kode HTML mudah dibaca dan tidak ada penyalahgunaan tag.
23 |  | Elemen semantik bagus untuk SEO dan juga dapat meningkatkan accessibility.
24 |  </p>
25 </article>
26
27 <footer>
28 |  Copyright © 2025 saya.com
29 </footer>
30
31 </body>
32 </html>
```

Style untuk Elemen Semantik

Cara memberikan style untuk elemen semantik sama saja seperti memberikan style pada elemen lainnya. Tinggal membuat atribut style, lalu mengisinya dengan kode style CSS.

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>web saya</title>
7  </head>
8  <body style="background-color: #silver;">
9      <header style="background-color: #white;">
10         <h1>Belajar Elemen Semantik di HTML</h1>
11     </header>
12
13     <nav>
14         <a href="#">Home</a> |
15         <a href="#">About</a> |
16         <a href="#">Contact</a>
17     </nav>
18
19     <article style="background-color: #white;">
20         <h1>Tutorial Semantik Elemen untuk Pemula</h1>
21         <p>Semantik elemen adalah elemen yang memiliki makna dan tujuan.
22             Tujuannya agar kode HTML mudah dibaca dan tidak ada penyalahgunaan tag.
23             Elemen semantik bagus untuk SEO dan juga dapat meningkatkan accessibility.
24         </p>
25     </article>
26
27     <footer style="background-color: #white;">
28         Copyright © 2025 saya.com
29     </footer>
30
31 </body>
32 </html>
```

TUGAS BUATLAH WEB dengan semantic element

HTML Entity

Apa itu HTML Entity

adalah kode yang mewakili karakter khusus, dimulai dengan & dan diakhiri dengan ;, untuk menampilkan karakter yang tidak dapat ditik di keyboard standar atau yang memiliki arti khusus di HTML (seperti <, >, dan &). Kode ini memastikan karakter ditampilkan dengan benar dan konsisten di semua browser. Contohnya, < digunakan untuk menampilkan simbol "<" dan digunakan untuk spasi tanpa putus.

HTML Entity dapat direpresentasikan sebagai potongan string. Dalam penggunaannya, entity menggunakan & atau ampersan untuk awalan sebuah entity, yang tidak jarang kita temukan dalam proses pembuatan sebuah website ataupun ketika sedang mengunjungi sebuah website entah itu simbol karakter seperti copyright (©), lebih besar (>), lebih kecil (<), dan sebagainya.

Dalam penggunaan **HTML Entity** terdapat 2 macam entity yaitu Entity Name dan juga Entity Number.

Entity Number

Penggunaan dibuat dengan desimal ASCII atau heksadesimal ASCII, contoh penggunaan :

```
<p>&#169; pplg SMK N 1 Kandeman </p>
```

Entity Name

Penggunaan lebih mudah karena menggunakan alfabet sehingga lebih mudah diingat oleh user, contoh penggunaan :

```
<p>&copy; pplg SMK N 1 Kandeman</p>
```

Terlihat perbedaan keduanya, Entity Name lebih mudah diingat karena menggunakan huruf alfabet namun dibalik kemudahan tersebut ternyata Entity Name masih belum mendukung semua browser sehingga penggunaan Entity Number lebih diunggulkan dari segi dukungan browser.

Beberapa karakter entity lain yang dapat digunakan :

Char	Number	Entity
©	©	©
®	®	®
™	™	™
@	@	@
¤	¶	¶
§	§	§
&	&	&
>	>	>
<	<	<

HTML Symbols

Symbol yang tidak ada di keyboard Anda dapat ditambahkan dengan menggunakan entitas.

Entitas HTML Symbols

Banyak Symbol matematika, teknis, dan mata uang, tidak ada pada keyboard normal. Untuk menambahkan simbol tersebut ke halaman HTML, Anda dapat menggunakan nama entitas atau nomor entitas (referensi desimal atau heksadesimal) untuk Symbol tersebut.

Contoh

Tampilkan tanda euro, €, dengan nama entitas, desimal, dan nilai heksadesimal:

```
<p>I will display &euro;</p>
<p>I will display &#8364;</p>
<p>I will display &#x20AC;</p>
```

Hasilnya menjadi:

```
<p>I will display &euro;</p>
<p>I will display &#8364;</p>
<p>I will display &#x20AC;</p>
```

Beberapa Symbol Matematika yang Didukung oleh HTML

Nomor	Char	Nomor	Entitas	Deskripsi
01	\forall	∀	∀	UNTUK SEMUA
02	∂	∂	∂	DIFERENSIAL PARSIAL
03	\exists	∃	∃	ADA
04	\emptyset	∅	∅	SET KOSONG
05	∇	∇	∇	NABLA
06	\in	∈	∈	ELEMENT DARI
07	\notin	∉	∉	BUKAN ELEMENT DARI
08	\ni	∋	∋	BERISI SEBAGAI ANGGOTA
09	\prod	∏	∏	N-ARY PRODUK
10	\sum	∑	∑	N-ARY JUMLAH

Beberapa Bahasa Yunani yang Didukung oleh HTML

Nomor	Char	Nomor	Entitas	Deskripsi
01	A	Α	Α	HURUF BESAR ALPHA YUNANI
02	B	Β	Β	HURUF BESAR BETA YUNANI
03	Γ	Γ	Γ	HURUF BESAR GAMMA YUNANI
04	Δ	Δ	Δ	HURUF BESAR DELTA YUNANI
05	E	Ε	Ε	HURUF BESAR EPSILON YUNANI
06	Z	Ζ	Ζ	HURUF BESAR ZETA YUNANI

Beberapa Entitas lain yang Didukung oleh HTML

Nomor	Char	Nomor	Entitas	Deskripsi
01	©	©	©	TANDA HAK CIPTA
02	®	®	®	TANDA TERDAFTAR
03	€	€	€	TANDA EURO
04	™	™	™	MEREK DAGANG
05	←	←	←	PANAH KIRI
06	↑	↑	↑	PANAH ATAS
07	→	→	→	PANAH KANAN
08	↓	↓	↓	PANAH BAWAH
09	♠	♠	♠	SPADE HITAM
10	♣	♣	♣	CLUB HITAM
11	♥	♥	♥	HEART HITAM
12	♦	♦	♦	DIAMOND HITAM

HTML Emojis

HTML dapat menampilkan sebuah emojis dengan menggunakan *decimal* atau *hexadecimal*, dan agar browser memahami kalau kita sedang menampilkan character kita harus mengawali dengan &# dan diakhiri dengan ;, sebagai contoh

```
<!-- Menggunakan Hexadecimal-->
<p>Akan menampilkan &#x1F981;</p> <!--Akan menampilkan 🐻 -->

<!-- Menggunakan Decimal -->
<p>Akan menampilkan &#129409;</p> <!--Akan menampilkan 🐻 -->
```

Daftar emojis dapat kamu lihat disini <https://unicode.org/emoji/charts/full-emoji-list.html>

Emojis dengan tone warna kulit

Beberapa emoji memiliki tone warna kulit yang berbeda beda, berikut contoh tone warna kulitnya,

Decimal	Warna	Preview
🏿	Dark skin tone	
🏾	Medium Dark skin tone	
🏽	Medium skin tone	
🏼	Medium Light skin tone	
🏻	Light skin tone	

Penggunaan warna kulit dapat ditambahkan setelah kode emoji, sebagai contoh

```
<p><span>&#129306;</span> No skin tone</p> <!--肤色无-->
<p><span>&#129306;&#127999;</span> Dark skin tone</p> <!--深色肤色-->
<p><span>&#129306;&#127998;</span> Medium Dark skin tone</p> <!--中深色肤色-->
<p><span>&#129306;&#127997;</span> Medium skin tone</p> <!--中色肤色-->
<p><span>&#129306;&#127996;</span> Medium Light skin tone</p> <!--浅色肤色-->
<p><span>&#129306;&#127995;</span> Light skin tone</p> <!--浅肤色-->
```

Praktik !

Entity

entity.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Contoh Entity</title>
</head>
<body>
    <h2>Contoh Entity</h2>
    <p>Tanda kurang dari: &lt;</p>
    <p>Tanda lebih dari: &gt;</p>
    <p>Simbol copyright: &copy;</p>
</body>
</html>
```

symbol.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Contoh Symbol</title>
</head>
<body>
    <h2>Simbol HTML</h2>
    <p>Euro: &euro;</p>
    <p>Pi: &pi;</p>
    <p>Panah ke kiri: &larr;</p>
</body>
</html>
```

emoji.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Emoji di HTML</title>
</head>
<body>
    <h2>Emoji</h2>
    <p>Emoji senyum: 😊 (&#128512;)</p>
    <p>Api: 🔥 (&#x1F525;)</p>
    <p>Jempol: 👍 (&#128077;)</p>
</body>
</html>
```

Latihan Praktik

Praktik 1 — Entity

Buat halaman yang menampilkan:

- Kalimat mengandung <html> yang ditulis menggunakan entity.
- Simbol ©, ®, dan ™.

Praktik 2 — Symbol

Buat halaman dengan daftar simbol:

- Mata uang (€, ¥, £)
- Tanda panah (← ↑ → ↓)
- Simbol matematika (∞, √, ±)

Praktik 3 — Emoji

Buat halaman yang menampilkan 5 emoji bertema:

- Hewan
- Wajah ekspresi
- Aktivitas

Tugas

Buat satu halaman HTML berjudul "**Entity, Symbol & Emoji**" yang memuat:

1. Penjelasan singkat HTML Entity.
2. Tabel berisi 10 entity beserta tampilan outputnya.
3. Daftar simbol matematika berikut entity-nya.
4. Galeri emoji menggunakan minimal 8 HTML emoji.
5. Tampilan halaman harus rapi menggunakan tag `<h1>`, `<h2>`, `<p>`, `` dan `<table>`.

HTML Canvas

HTML `<canvas>` adalah elemen yang digunakan untuk menggambar grafik secara dinamis menggunakan JavaScript. Dengan canvas, kita dapat membuat gambar, animasi, diagram, game, dan visual interaktif lainnya.

HTML Canvas?

`<canvas>` adalah elemen HTML yang menyediakan area menggambar tempat JavaScript dapat membuat grafik 2D maupun 3D. Elemen ini sendiri hanya berupa area kosong, dan JavaScript-lah yang mengontrol isi gambarnya.

Contoh deklarasi:

```
<canvas id="myCanvas" width="400" height="300"></canvas>
```

Konteks 2D

Untuk menggambar pada canvas, kita harus mengambil context.

```
const c = document.getElementById("myCanvas");
const ctx = c.getContext("2d");
```

Fungsi Dasar Menggambar

Beberapa fungsi dasar:

- **Menggambar garis:** `moveTo()`, `lineTo()`, `stroke()`
- **Menggambar persegi:** `fillRect()`, `strokeRect()`
- **Menggambar lingkaran:** `arc()`
- **Mengatur warna:** `fillStyle`, `strokeStyle`

Praktik canvas

[canvasPersegi.html](#)

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Canvas</title>
7  </head>
8  <body>
9      <canvas id="myCanvas" width="300" height="200" style="border:1px solid black"></canvas>
10 <script>
11     const c = document.getElementById("myCanvas");
12     const ctx = c.getContext("2d");
13     ctx.fillStyle = "blue";
14     ctx.fillRect(50, 50, 150, 100);
15 </script>
16
17 </body>
18 </html>
```

canvasGaris.html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Canvas</title>
7  </head>
8  <body>
9      <canvas id="myCanvas" width="300" height="200" style="border:1px solid black"></canvas>
10 <script>
11     const c = document.getElementById("myCanvas");
12     const ctx = c.getContext("2d");
13     ctx.moveTo(0, 0);
14     ctx.lineTo(300, 200);
15     ctx.stroke();
16 </script>
17
18 </body>
19 </html>
```

canvasLingkaran.html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>Canvas</title>
7  </head>
8  <body>
9  <canvas id="myCanvas" width="300" height="200" style="border:1px solid black"></canvas>
10 <script>
11 const c = document.getElementById("myCanvas");
12 const ctx = c.getContext("2d");
13 ctx.beginPath();
14 ctx.arc(150, 100, 50, 0, 2 * Math.PI);
15 ctx.stroke();
16 </script>
17
18 </body>
19 </html>
```

canvasTeks.html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>Canvas</title>
7  </head>
8  <body>
9  <canvas id="myCanvas" width="300" height="200" style="border:1px solid black"></canvas>
10 <script>
11 const c = document.getElementById("myCanvas");
12 const ctx = c.getContext("2d");
13 ctx.font = "20px Arial";
14 ctx.fillText("Hello Canvas!", 50, 100);
15 </script>
16
17 </body>
18 </html>
```

Latihan Praktik

Praktik 1 — Menggambar Bentuk Dasar

Buat halaman HTML dengan canvas berisi:

- Satu persegi berwarna.
- Satu lingkaran di tengah.
- Satu garis diagonal.

Praktik 2 — Mengatur Warna dan Garis

Buat 3 bentuk (lingkaran, persegi, garis) dengan warna yang berbeda. Setiap bentuk harus memiliki ketebalan garis yang berbeda menggunakan:

```
ctx.lineWidth = value;
```

Praktik 3 — Teks dan Bentuk

Buat halaman yang menampilkan teks menggunakan canvas dan tambahkan bentuk lain (lingkaran/persegi) sebagai dekorasi.

HTML Multimedia

Elemen <video> dalam HTML digunakan untuk menampilkan dan memutar video secara langsung di halaman web tanpa perlu plugin tambahan. HTML5 menyediakan kontrol bawaan seperti play, pause, volume, subtitle, dan fullscreen.

Apa itu Elemen <video>?

<video> adalah elemen HTML yang memungkinkan pemutaran file video dengan dukungan berbagai format seperti MP4, WebM, dan Ogg.

Contoh dasar:

```
<video src="video.mp4" controls></video>
```

Format video

Ada banyak sekali format video pada multimedia, hanya saja MP4, WebM, dan Ogg yang support pada HTML. Youtube merekomendasi format MP4.

Format	File	Deskripsi
MPEG	.mpg .mpeg	Dikembangkan oleh Moving Pictures Expert Group. Format video yang pertama populer di web. Tidak lagi mensupport HTML.
AVI	.avi	AVI (Audio Video Interleave). Dikembangkan oleh Microsoft. Biasa digunakan di kamera video dan perangkat keras TV. Dapat diputar dengan baik di komputer Windows, tetapi tidak di web browser.;
WMV	.wmv	WMV (Windows Media Video). Dikembangkan oleh Microsoft. Biasa digunakan di kamera video dan perangkat keras TV. Dapat diputar dengan baik di komputer Windows, tetapi tidak di web browser.
QuickTime	.mov	QuickTime. Dikembangkan oleh Apple. Biasa digunakan di kamera video dan perangkat keras TV. Dapat diputar dengan baik di komputer Apple, tetapi tidak di web browser.
RealVideo	.rm .ram	RealVideo. Dikembangkan oleh Real Media untuk memungkinkan streaming video dengan bandwidth rendah. Tidak dapat diputar di web browser.
Flash	.swf .flv	Flash. Dikembangkan oleh Macromedia. Seringkali membutuhkan komponen tambahan (plug-in) untuk dimainkan di browser web.
Ogg	.ogg	Theora Ogg. Dikembangkan oleh Xiph.Org Foundation. Didukung oleh HTML.
WebM	.webm	WebM. Dikembangkan oleh Mozilla, Opera, Adobe, dan Google. Didukung oleh HTML.
MPEG-4 atau MP4	.mp4	MP4. Dikembangkan oleh Moving Pictures Expert Group. Biasa digunakan di kamera video dan perangkat keras TV. Didukung oleh semua browser dan direkomendasikan oleh YouTube.

Atribut Penting pada <video>

- **controls** — menampilkan tombol kontrol.
- **autoplay** — memutar otomatis video.
- **loop** — mengulang video terus.
- **muted** — mematikan suara.
- **poster** — gambar awal sebelum video diputar.
- **width / height** — ukuran video.

Menambahkan Banyak Sumber Video (Fallback)

Digunakan agar browser memilih format video yang didukung.

```
<video controls>
  <source src="video.mp4" type="video/mp4">
  <source src="video.webm" type="video/webm">
  Browser Anda tidak mendukung elemen video.
</video>
```

Praktik video

video1.html

Video dengan Kontrol Dasar

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>video</title>
7  </head>
8  <body>
9  <h2>Video Dasar</h2>
10 <video src="sample.mp4" width="400" controls></video>
11 </body>
12 </html>
```

video2.html

Video Autoplay dan Loop

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>video</title>
7  </head>
8  <body>
9  <h2>Video Autoplay</h2>
10 <video src="sample.mp4" width="400" autoplay loop muted></video>
11 </body>
12 </html>
```

video3.html

Video dengan Poster

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>video</title>
7  </head>
8  <body>
9  <h2>Video dengan Poster</h2>
10 <video src="sample.mp4" width="400" controls poster="thumbnail.jpg"></video>
12
13 </body>
14 </html>
```

video4.html

Memainkan Video Memakai JavaScript

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>video</title>
7  </head>
8  <body>
9  <video id="myVid" width="400" controls>
10 <source src="sample.mp4" type="video/mp4">
11 </video>
12
13 <button onclick="playVideo()">Play</button>
14 <button onclick="pauseVideo()">Pause</button>
15
16 <script>
17 function playVideo() {
18 document.getElementById("myVid").play();
19 }
20 function pauseVideo() {
21 document.getElementById("myVid").pause();
22 }
23 </script>
24
25 </body>
26 </html>
```

Praktik 1 — Video Dasar

Buat halaman HTML berisi satu video yang memiliki:

- kontrol,
- ukuran 500px,

- teks judul di atas video.

4.2 Praktik 2 — Video dengan Poster dan Loop

Tambahkan video dengan poster gambar, dan atur agar video memutar secara otomatis dan berulang.

4.3 Praktik 3 — Tombol Play/Pause Manual

Buat dua tombol:

- Play
- Pause untuk mengontrol video menggunakan JavaScript.

HTML Audio

Elemen `<audio>` dalam HTML digunakan untuk memutar file suara di halaman web tanpa plugin tambahan. HTML5 menyediakan berbagai atribut untuk mengontrol audio seperti play, pause, loop, volume, dan autoplay.

Apa itu Elemen `<audio>`?

`<audio>` adalah elemen HTML untuk memutar audio dengan format umum seperti MP3, WAV, dan Ogg.

Contoh dasar:

```
<audio src="audio.mp3" controls></audio>
```

Atribut Penting pada `<audio>`

- **controls** — menampilkan kontrol pemutar audio.
- **autoplay** — memutar audio otomatis.
- **loop** — mengulang audio.
- **muted** — mematikan suara audio.
- **preload** — menentukan bagaimana browser memuat audio (auto, metadata, none).

Menambahkan Banyak Sumber Audio (Fallback)

Digunakan agar browser memilih format yang didukung.

```
<audio controls>
  <source src="audio.mp3" type="audio/mpeg">
  <source src="audio.ogg" type="audio/ogg">
  Browser Anda tidak mendukung elemen audio.
</audio>
```

Praktik html audio

audio1.html

Audio dengan Kontrol Dasar

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>video</title>
7  </head>
8  <body>
9  <h2>Audio Dasar</h2>
10 <audio src="lagu.mp3" controls></audio>
11
12 </body>
13 </html>
```

audio2.html

Audio Autoplay dan Loop

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>video</title>
7  </head>
8  <body>
9  <h2>Audio Autoplay</h2>
10 <audio src="musik.mp3" autoplay loop muted></audio>
11
12 </body>
13 </html>
```

audio3.html

Menggunakan Beberapa Sumber Audio

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>video</title>
7  </head>
8  <body>
9  <audio controls>
10 <source src="audio.mp3" type="audio/mpeg">
11 <source src="audio.wav" type="audio/wav">
12 </audio>
13
14 </body>
15 </html>
```

audio4.html

Mengontrol Audio dengan JavaScript

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |  <meta charset="UTF-8">
5  |  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  |  <title>video</title>
7  </head>
8  <body>
9  <audio id="myAudio" src="lagu.mp3" controls></audio>
10 <br>
11 <button onclick="playAudio()">Play</button>
12 <button onclick="pauseAudio()">Pause</button>
13 <button onclick="stopAudio()">Stop</button>
14
15
16 <script>
17 function playAudio() {
18 document.getElementById("myAudio").play();
19 }
20 function pauseAudio() {
21 document.getElementById("myAudio").pause();
22 }
23 function stopAudio() {
24 const audio = document.getElementById("myAudio");
25 audio.pause();
26 audio.currentTime = 0;
27 }
28 </script>
29
30 </body>
31 </html>
```

Latihan Praktik

Praktik 1 — Audio Dasar

Buat halaman HTML berisi audio dengan:

- kontrol,
- judul di atas audio,
- deskripsi singkat.

Praktik 2 — Audio Otomatis

Tambahkan satu audio dengan atribut autoplay, loop, dan muted.

Praktik 3 — Kontrol Manual

Buat tombol Play, Pause, dan Stop untuk mengontrol audio melalui JavaScript.

HTML Figure

Elemen <figure> dalam HTML digunakan untuk membungkus konten seperti gambar, ilustrasi, diagram, atau cuplikan kode yang memiliki hubungan dengan isi halaman. Biasanya digunakan bersama <figcaption> yang berfungsi sebagai teks keterangan.

Apa itu <figure>?

<figure> adalah elemen semantik HTML yang digunakan untuk menyajikan konten mandiri seperti:

- gambar
- grafik dan diagram
- kutipan panjang
- tabel atau ilustrasi
- cuplikan kode

Elemen ini biasanya digunakan dengan <figcaption> sebagai keterangan.

Apa itu <figcaption>?

<figcaption> memberikan deskripsi atau keterangan dari elemen yang ada di dalam <figure>.

Contoh struktur:

```
<figure>
  
  <figcaption>Pemandangan gunung saat pagi hari.</figcaption>
</figure>
```

Kelebihan Menggunakan <figure>

- Lebih semantik dan SEO-friendly.
- Memudahkan menambahkan keterangan gambar.
- Layout lebih fleksibel.

Praktik figure html

figure1.html

Figure dengan Gambar dan Keterangan

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>video</title>
7  </head>
8  <body>
9  <h2>Contoh Figure</h2>
10 <figure>
11     
12     <figcaption>Gambar pemandangan gunung.</figcaption>
13 </figure>
14
15 </body>
16 </html>
```

figure2.html

Figure untuk Ilustrasi Diagram

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>figure</title>
7  </head>
8  <body>
9  <h2>Diagram pada Figure</h2>
10 <figure>
11     
12     <figcaption>Diagram alur proses login aplikasi.</figcaption>
13 </figure>
14
15 </body>
16 </html>
```

figure3.html

Figure untuk Cuplikan Kode

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>figure</title>
7  </head>
8  <body>
9      <h2>Figure dengan Kode Program</h2>
10     <figure>
11         <pre>
12             function hello() {
13                 console.log("Hello World");
14             }
15         </pre>
16         <figcaption>Contoh fungsi JavaScript sederhana.</figcaption>
17     </figure>
18 </body>
19 </html>
```

Latihan Praktik

Praktik 1 — Figure Dasar

Buat halaman HTML yang menampilkan sebuah gambar dengan keterangan yang relevan menggunakan <figure> dan <figcaption>.

Praktik 2 — Tiga Figure

Buat tiga figure yang berisi:

- Gambar alam
- Gambar hewan
- Gambar objek Setiap figure harus memiliki keterangan.

Praktik 3 — Figure Cuplikan Kode

Buat sebuah figure yang menampilkan cuplikan kode HTML atau JavaScript lengkap dengan keterangan.