HTML

MAAK JE EERSTE WEBPAGINA

<HTML>

Studio Shodwe Project. 2024

WAT IS HTML?

Wat is HTML?

HTML is eigenlijk een afkorting voor **Hypertext Markup Language**. Het is de taal van webpagina's die een browser vertelt hoe bepaalde elementen, zoals tekst en afbeeldingen, moeten worden weergegeven, door middel van codes en symbolen.

HTML is de standaard voor het maken van webpagina's. Het World Wide Web Consortium, of W3C, beveelt het aan. Daarom implementeren de meeste browsers HTML om webpagina's min of meer uniform weer te geven.

WAAROM HTML LEREN?

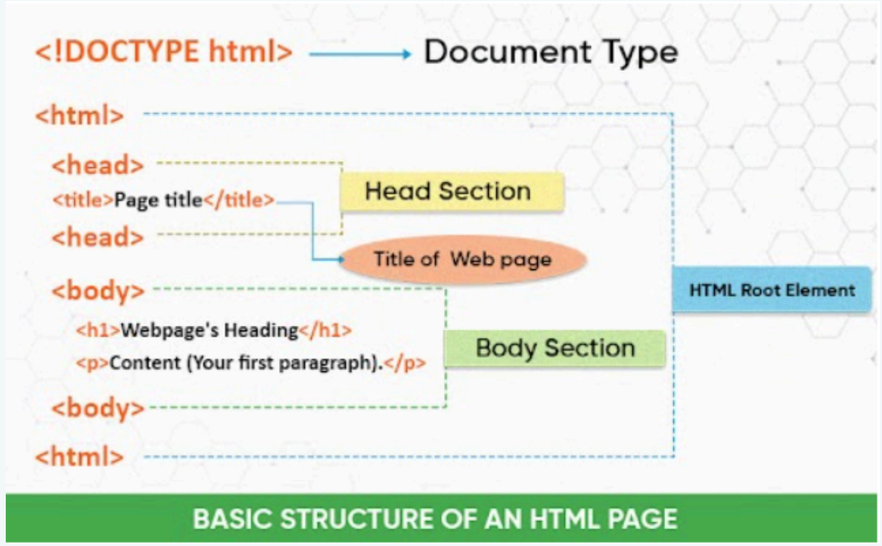
Basis van webontwikkeling: HTML is het startpunt voor iedereen die geïnteresseerd is in webontwikkeling.

Gemakkelijk te leren: Eenvoudige syntaxis die beginnersvriendelijk is.

Veelzijdig: Werkt samen met CSS en JavaScript om interactieve webpagina's te maken.

HTML PAGINASTRUCTUUR

De HTML-paginastructuur (ofwel de basisstructuur van HTML) bestaat uit de essentiële elementen die nodig zijn om een HTML-document te maken dat in de browser kan worden weergegeven.



STRUCTUUR

- De <!DOCTYPE html>-declaratie definieert dat dit document een HTML5-document is.
- Het <html>-element is het rootelement van een HTML-pagina.
- Het <head>-element bevat metagegevens over de HTML-pagina.
- Het <title>-element specificeert een titel voor de HTML-pagina (die wordt weergegeven in de titelbalk van de browser of op het tabblad van de pagina).
- Het <body>-element definieert de hoofdtekst van het document en is een container voor alle zichtbare inhoud, zoals koppen, alinea's, afbeeldingen, hyperlinks, tabellen, lijsten, enz.
- Het <h1>-element definieert de belangrijkste kop (hoofdtitel) van een webpagina.
- Het <p>-element definieert een alinea.

HTML Editors

What are HTML Editors?

HTML-editors zijn tools of software waarmee je HTML-documenten kunt maken, bewerken en beheren. Sommige HTML-editors kunnen HTML-bestanden ook uitvoeren of weergeven. Ze bieden handige functies zoals automatische inspringing, syntaxkleuren en automatisch aanvullen van code.

Populaire HTML-editors in 2025

Er zijn verschillende HTML-editors beschikbaar, zowel gratis als betalend. Hieronder volgt een lijst van enkele populaire editors:

1. **Notepad**
2. **TextEdit**
3. **Notepad++**
4. **Sublime**
5. **Visual Studio Code**
6. **Atom**
7. **Brackets**
8. **Adobe Dreamweaver CC**
9. **CoffeeCup**

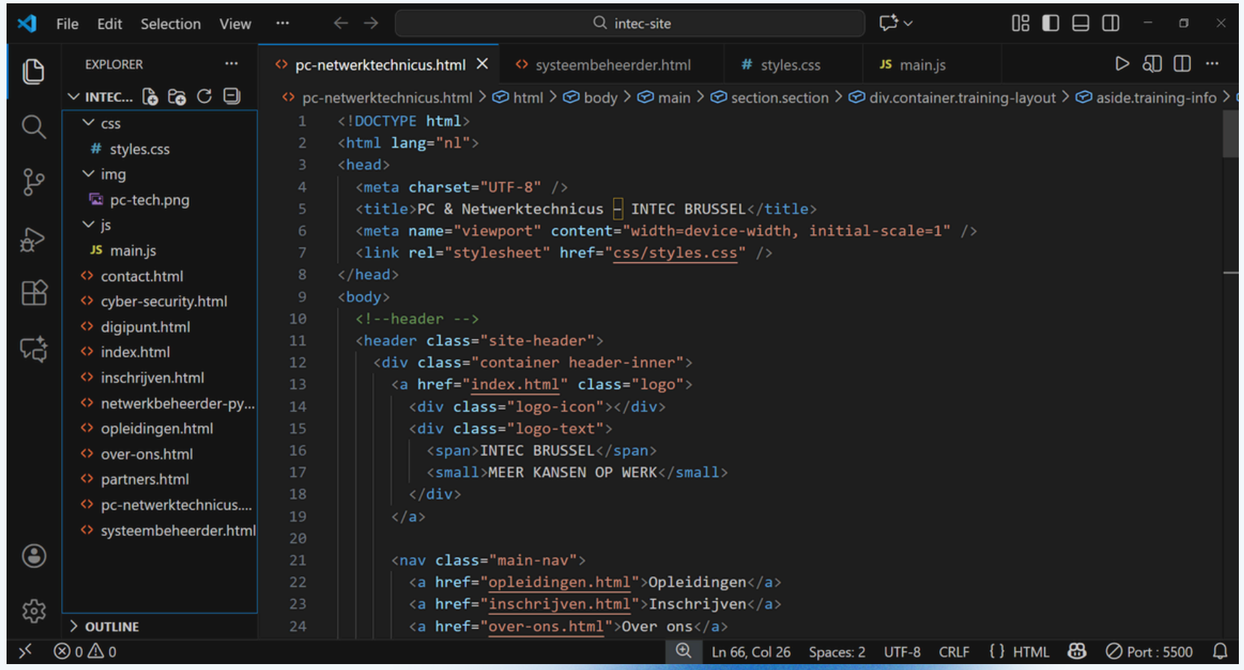
Visual Studio Code

Visual Studio Code (VS Code) is een broncode-editor die is ontwikkeld door Microsoft.

De editor ondersteunt Windows, Linux en macOS.

VS Code wordt veel gebruikt voor het maken en beheren van HTML-documenten dankzij handige functies zoals debugging, syntaxkleuren, codeherstructurering en intelligente code-aanvulling.

Visual Studio Code

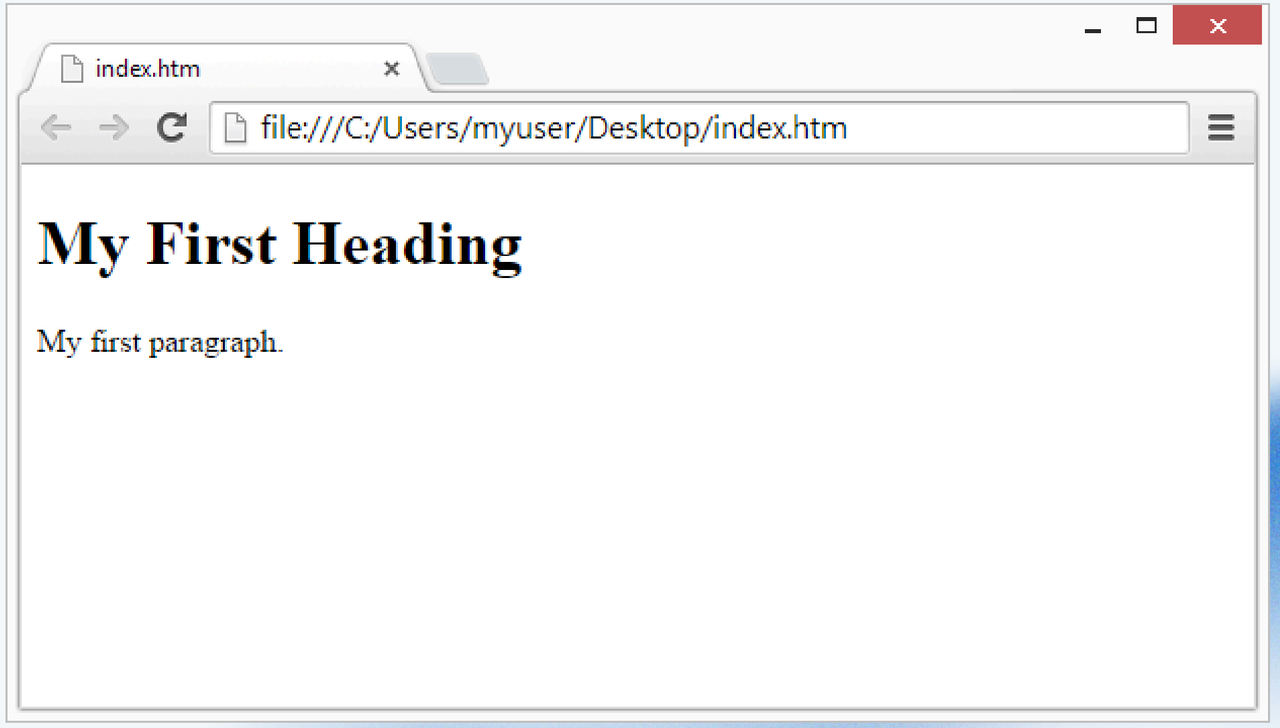


Web Browsers

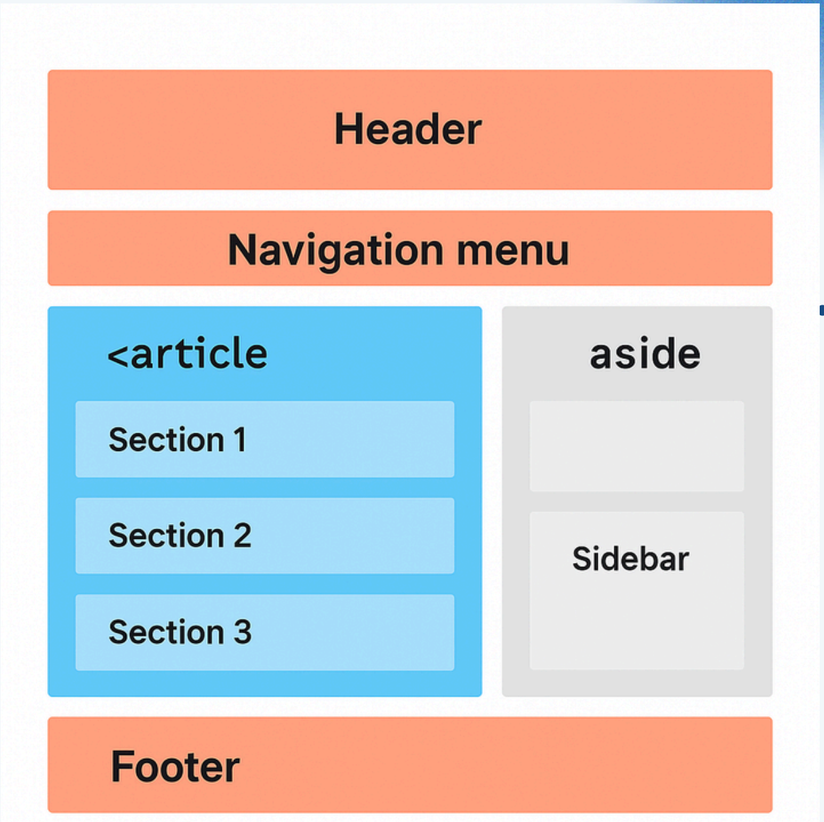
Het doel van een webbrowser (Chrome, Edge, Firefox, Safari) is om HTML-documenten te lezen en correct weer te geven.

Een browser geeft de HTML-tags niet weer, maar gebruikt ze om te bepalen hoe het document moet worden weergegeven:

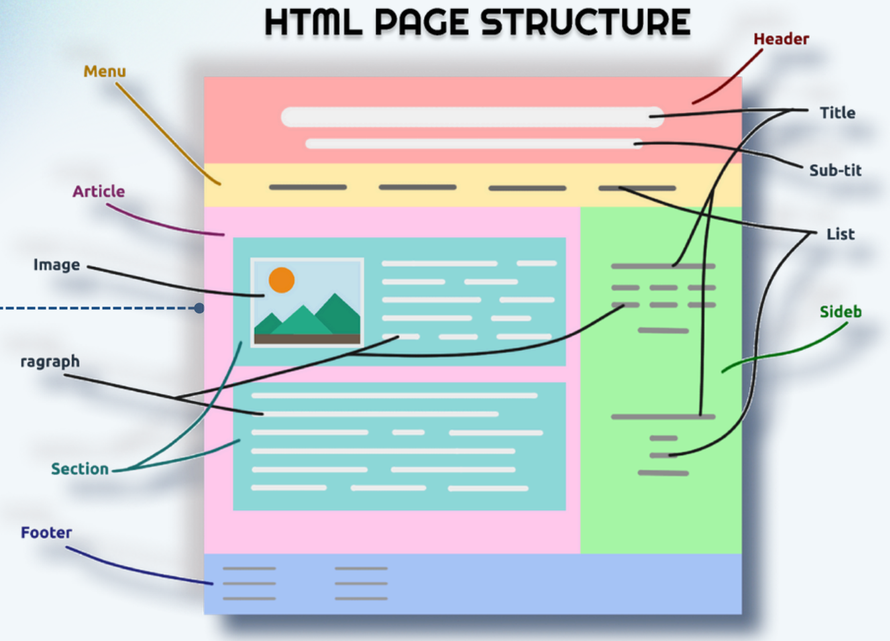
```
<h1>My First Heading</h1>
<p>My first paragraph.</p>
```



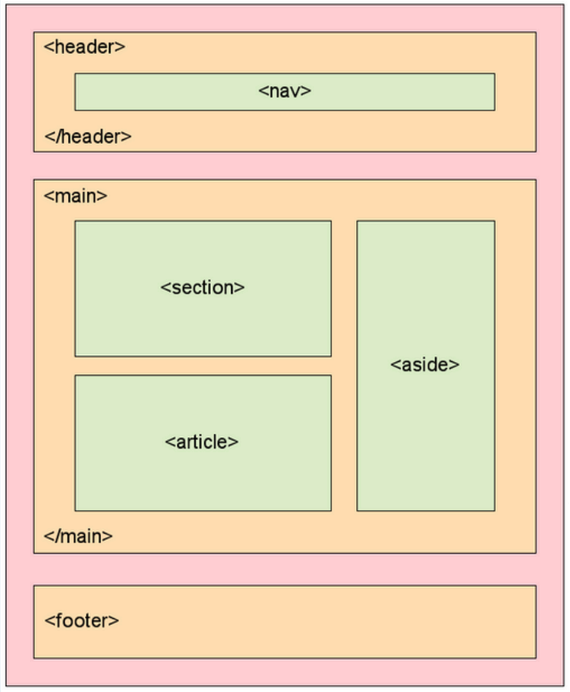
Studio Shodwe Project.



HTML PAGE STRUCTURE



visualisatie van een HTML-paginastructuur:



Hier zie je:

- Boven: Header met titel/logo
- Eronder: Navigatiebalk
- Linkerzijde: Article/Main content
- Rechterzijde: Aside (zijbalk)
- Onderaan: Footer

The <!DOCTYPE> Declaration

De <!DOCTYPE>-declaratie vertegenwoordigt het documenttype en helpt browsers webpagina's correct weer te geven.

Deze mag slechts één keer voorkomen, bovenaan de pagina (vóór eventuele HTML-tags).

De <!DOCTYPE>-declaratie is niet hoofdlettergevoelig.

De <!DOCTYPE>-declaratie voor HTML5 is:

<!DOCTYPPE html>

index.html > html

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
    initial-scale=1.0">
    <title>HTML - Tables</title>
</head>
<body>

  <p lang="en-GB">This paragraph is defined as British
  English.</p>

<p lang="fr">Ce paragraphe est défini en français.</p>

</body>
</html>
```

Bekijk de online voorbeelden:

https://codepen.io/Quinten-Work/pen/yLjYjyQ

https://codepen.io/Quinten-Work/pen/OJZyZMO

Dit zijn de tags die in het voorbeeld worden gebruikt:

Tags	Wat is het?
<!DOCTYPE html>	Dit is een moderne HTML5-pagina. Zodat de browser weet hoe hij de pagina moet begrijpen.
<html>	De hele webpagina zit binnen deze tag.
<head>	Dit is het hoofd van de pagina — bezoekers zien dit niet. Wat staat erin? titel van de pagina, instellingen, koppelingen naar CSS of scripts
<meta>	Geeft de browser metadata over het document. Het charset-attribuut declareert de tekencodering. Moderne HTML-documenten zouden altijd UTF-8 moeten gebruiken, ook al is dit geen vereiste. In HTML vereist de <meta>-tag geen afsluitende tag. Zie het onderwerp Meta voor meer informatie: https://riptutorial.com/html/topic/1264/meta-information
<title>	**De titel van de tab in de browser. Voorbeeld: <title>Mijn Eerste Website</title>**
<body>	Alles wat de gebruiker kan zien: tekst afbeeldingen knoppen menu's Alles binnen <body> is zichtbaar.
<h1>	De belangrijkste titel van de pagina. Tip: Slechts één h1 per pagina Grotere titels gebruik je met: <h2>, <h3> enz.
	Een paragraaf of gewoon een tekstblok. Voorbeeld: Deze website is gemaakt in HTML.

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>tutotial lang</title>
</head>
<body>

  <p lang="nl">
    Dit artikel is geschreven in het nederlands.
  </p>

  <p lang="en">
    This article is written in English.
  </p>

  <p lang="fr">
    Cet article est rédigé en français.
  </p>

</body>
</html>
```

HTML

```html
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>tutotial lang</title>
</head>

<body>

  <p lang="nl">
    Dit artikel is geschreven in het nederlands.
  </p>

  <p lang="en">
    This article is written in English.
  </p>

  <p lang="fr">
    Cet article est rédigé en français.
  </p>

</body>

</html>
```

HTML tags

HTML-tags zijn de basiselementen van HTML en worden gebruikt om de structuur van een document te bepalen.
Het zijn letters of woorden die tussen punthaken staan (**<** en **>**).

Meestal bestaan HTML-tags uit een **openingstag** en een **sluitingstag**.
Elke tag heeft een eigen betekenis. De browser leest deze tags en geeft de inhoud die ertussen staat op de juiste manier weer.

Voorbeeld:
Als je tekst plaatst binnen de paragraaf-tag ( ), zal de browser deze tekst als een aparte alinea weergeven.

In deze tutorial bespreken we alle **basis HTML-tags**.

Heading Tags	<h1>, <h2>, <h3>, <h4>, <h5>, and <h6>
Paragraph Tag	
Line Break Tag	 or
Center Tag	
Horizontal Rule Tag	<hr> or <hr />
Preserve Formatting Tag	<pre>
Non-breaking Space	
Listing Tags	<ul> and </ul> • Unordered Listing: <ul> • Ordered Listing: <ol> • List Item: <li>

HTML Basic Tags

HTML Heading Tags

Wat zijn heading tags?

- Heading tags zijn titels en koppen in HTML.
- Ze helpen de structuur van een webpagina duidelijk te maken.
- Er zijn 6 niveaus: van <h1> (belangrijkste) tot <h6> (minste belangrijk).

Waarom zijn headings belangrijk?

- Ze maken de inhoud overzichtelijk.
- Ze zijn goed voor SEO (Google begrijpt je pagina beter).
- Ze helpen schermlezers (toegankelijkheid).

Voorbeeld:

```
<h1>Welkom op mijn website</h1>
<h2>Over mij</h2>
<h3>Mijn hobby's</h3>
<h4>Favoriete boeken</h4>
<h5>Kleine titel</h5>
<h6>Zeer kleine titel</h6>
```

HTML Paragraphs

Wat is een HTML-paragraph?

Een paragraaf is een blok tekst in HTML.
Je maakt een paragraaf met het <p>-element.

✔ Kenmerken:

- Elke paragraaf begint op een nieuwe regel
- Browsers voegen automatisch witruimte boven en onder toe
- Ideaal voor gewone tekst, uitleg, content

```
<p>Dit is mijn eerste paragraaf.</p>

<p>HTML paragraphs worden gemaakt met het p-element.</p>

<p>Een paragraaf begint altijd op een nieuwe regel.</p>
```

HTML Line Breaks

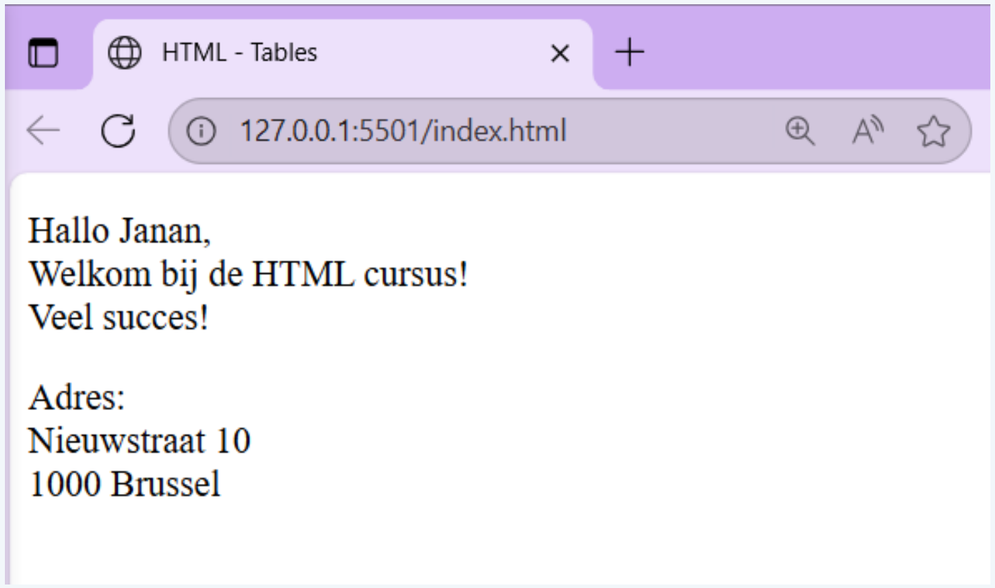
Wat is een line break?

Een line break breekt de tekst af en gaat direct naar een nieuwe regel, zónder een nieuwe paragraaf te starten.

✔ Dit gebruik je wanneer:

- Je binnen één paragraaf een nieuwe regel wilt
- Voor adressen, gedichten, of contactinfo
- Bij korte stukjes tekst die niet apart in <p> hoeven

```
<p>Hallo Janan,<br>
Welkom bij de HTML cursus!<br>
Veel succes!</p>
Adres:<br>
Nieuwstraat 10<br>
1000 Brussel
```



HTML Lists

Wat is een lijst in HTML?

Een lijst gebruik je om dingen overzichtelijk onder elkaar te tonen.

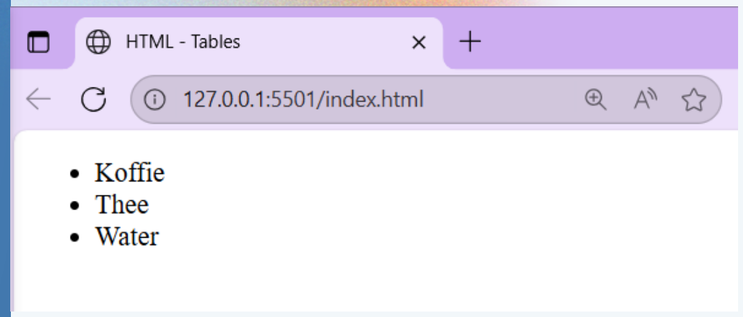
HTML heeft 3 soorten lijsten:

🔹 1. **Ongeordende lijst (unordered list)**

- Wordt gebruikt voor opsommingen
- Vorige tekens = bolletjes (•)

Tag: <ul>

```
<ul>
  <li>Koffie</li>
  <li>Thee</li>
  <li>Water</li>
</ul>
```

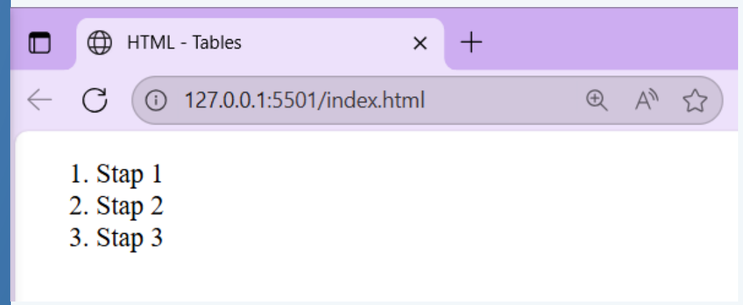


🔹 2. **Geordende lijst (ordered list)**

- Wordt gebruikt wanneer volgorde belangrijk is
- 1, 2, 3...

Tag: <ol>

```
<ol>
  <li>Stap 1</li>
  <li>Stap 2</li>
  <li>Stap 3</li>
</ol>
```



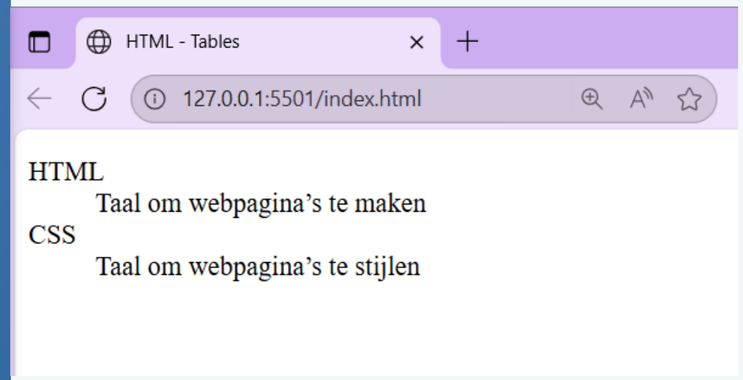
🔹 3. **Definitielijst (definition list)**

- Handig voor woordenboek, uitleg, definities
- Niet zo vaak gebruikt, maar goed om te kennen

Tags:

- <dl> → hele lijst
- <dt> → term
- <dd> → uitleg

```
<dl>
  <dt>HTML</dt>
  <dd>Taal om webpagina's te maken</dd>
  <dt>CSS</dt>
  <dd>Taal om webpagina's te stijlen</dd>
</dl>
```



HTML Tables

Wat is een HTML-tabel?

Een tabel gebruik je om gegevens in rijen en kolommen weer te geven.
Net als in Excel, maar dan in HTML.

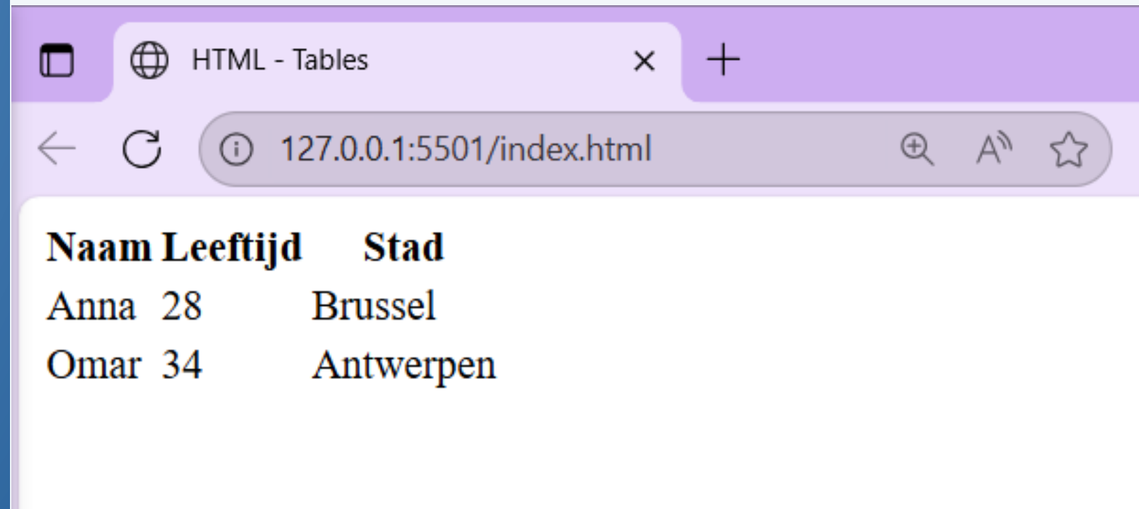
Tabel bestaat uit:

HTML	TagBetekenis
<table>	De hele tabel
<tr>	Een rij (table row)
<th>	Een kolomtitel (table header)
<td>	Een cel met data (table data)

```
<table>
 <tr>
  <th>Naam</th>
  <th>Leeftijd</th>
  <th>Stad</th>
 </tr>

 <tr>
  <td>Anna</td>
  <td>28</td>
  <td>Brussel</td>
 </tr>

 <tr>
  <td>Omar</td>
  <td>34</td>
  <td>Antwerpen</td>
 </tr>
</table>
```



HTML Links

HTML-koppelingen worden gedefinieerd met de <a>-tag:

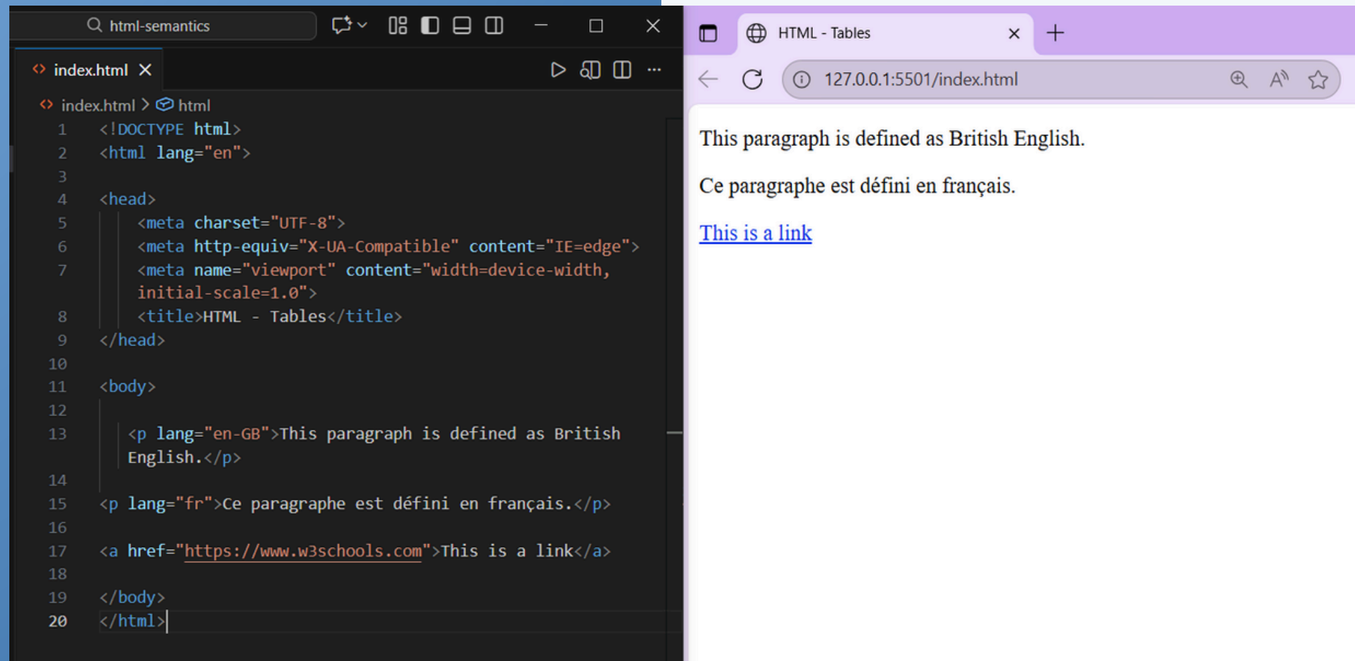
Voorbeeld:

```
<a href="https://www.w3schools.com">This is a link</a>
```

De bestemming van de link wordt gespecificeerd in het href-attribuut.

Attributen worden gebruikt om aanvullende informatie over HTML-elementen te verstrekken.

U leert meer over attributen in een later hoofdstuk.



HTML Media

Wat is HTML Media?

HTML Media zijn elementen die **beeld, geluid en video** toevoegen aan een webpagina.

De 3 belangrijkste media-elementen:

- Tag <img> → Afbeeldingen

```
<img src="images/foto.jpg" alt="Beschrijving van de foto">
```

- Tag <audio> → Geluidsbestanden

```
<audio controls>
  <source src="audio/music.mp3" type="audio/mpeg">
</audio>
```

- Tag <video> → Videobestanden

```
<video controls width="400">
  <source src="video/intro.mp4" type="video/mp4">
</video>
```

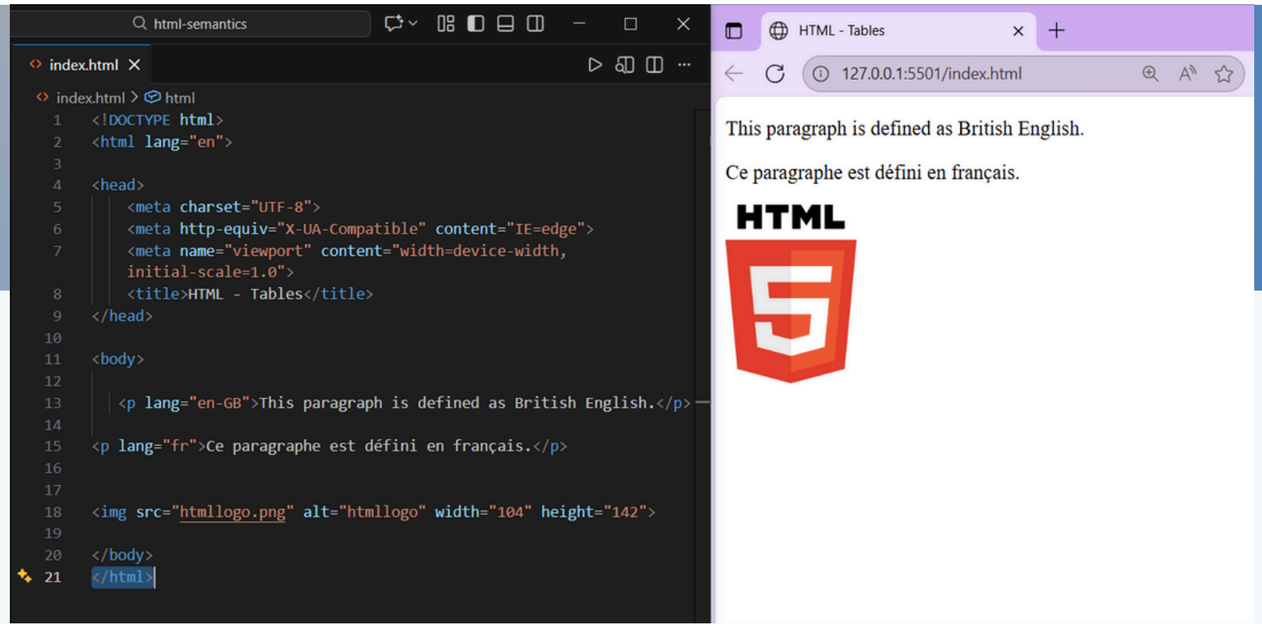
HTML Images

HTML-afbeeldingen worden gedefinieerd met de <img>-tag.

Het bronbestand (src), alternatieve tekst (alt), width en height worden als attributen verstrekt:

Voorbeeld:

```
<img src="w3schools.jpg" alt="W3Schools.com" width="104" height="142">
```

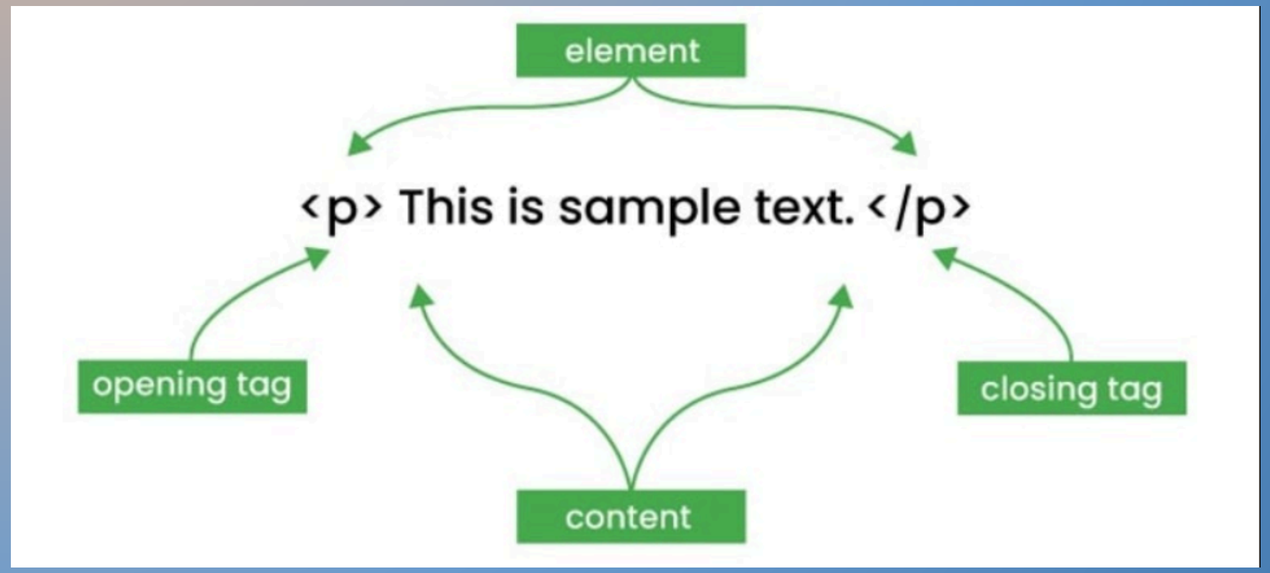


Semantic Elements in HTML

Wat is een HTML-element?

Een HTML-element is een basisonderdeel van een HTML-document en kan gegevens bevatten die op de webpagina worden weergegeven, zoals tekst, een afbeelding, een link of soms helemaal niets.
Een HTML-element bestaat uit een openingstag, inhoud en een sluitingstag. De openingstag kan ook attributen bevatten.

HTML Element Structure:



Start tag	Element content	End tag
<h1>	My First Heading	</h1>
	My first paragraph.	

```
<p>This is paragraph content.</p>
<h1>This is heading content.</h1>
<div>This is division content.</div>
```

HTML <header> Element

Wat is het <header> element?

Het <header>-element wordt gebruikt voor de bovenste sectie van een pagina of onderdeel (bijv. artikel).

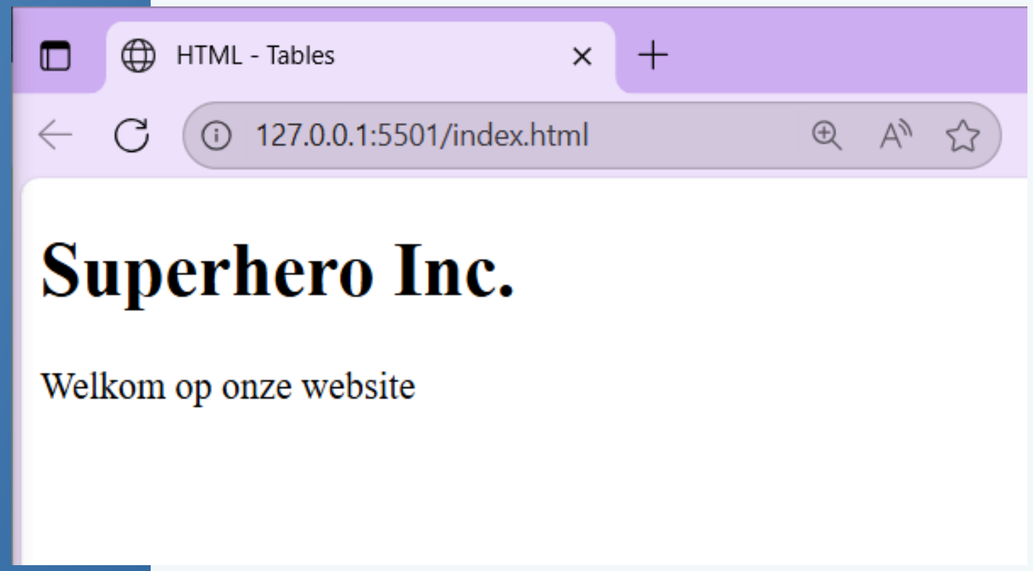
Een <header> bevat meestal:

- Titel van de pagina
- Logo
- Navigatie
- Introductietekst
- Branding

✓ **Waar kun je <header> gebruiken?**

- 1× voor de **hele pagina**
- Meerdere keren voor **artikels of secties**

```
<header>
 <h1>Superhero Inc.</h1>
 <p>Welkom op onze website</p>
</header>
```



HTML <nav> Element

Wat is het <nav> element?

Het <nav>-element wordt gebruikt voor **navigatie** op een website.

Het bevat meestal **links** naar:

- belangrijke pagina's
- secties van de site
- menu's
- navigatiebalken
- interne links

✓ **Waar gebruik je <nav>?**

- In de **header** (meest normaal)
- In een **sidebar**
- Als apart **hoofdmenu**
- Een pagina kan **meerdere** <nav> elementen hebben

```
<nav>
 <a href="index.html">Home</a>
 <a href="about.html">Over</a>
 <a href="contact.html">Contact</a>
 <a href="table.html">Tabel</a>
 <a href="form.html">Formulier</a>
 <a href="media.html">Media</a>
 <a href="links.html">Links</a>
</nav>
```

HTML <section> Element

Wat is het <section> element?

Het <section>-element wordt gebruikt om **inhoud te groeperen** in logische blokken.
Een “section” is een **zelfstandig thema** binnen een pagina.

Voorbeelden:

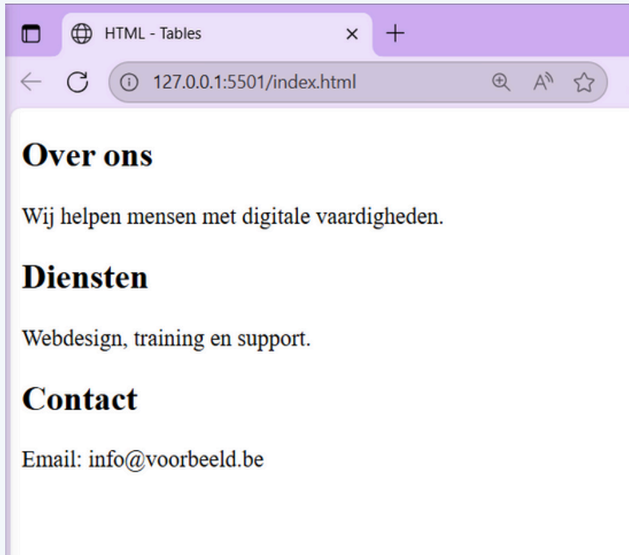
- Een “Over ons” sectie
- Een “Diensten” sectie
- Een “Contact” sectie
- Een hoofdstuk of onderdeel van een artikel

👉 Belangrijke tip:
Elke <section> bevat meestal een **heading** (<h2>, <h3>, ...).

```
<section>
 <h2>Over ons</h2>
 <p>Wij helpen mensen met digitale vaardigheden.</p>
</section>

<section>
 <h2>Diensten</h2>
 <p>Webdesign, training en support.</p>
</section>

<section>
 <h2>Contact</h2>
 <p>Email: info@voorbeeld.be</p>
</section>
```



HTML <article> Element

Wat is het <article> element?

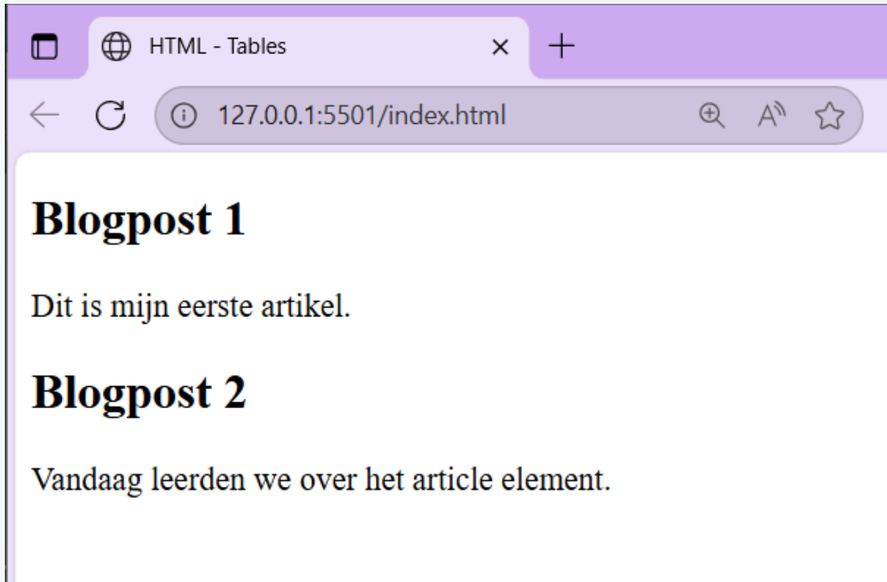
Het <article>-element wordt gebruikt voor zelfstandige, op zichzelf staande inhoud die apart gepubliceerd kan worden.

Een <article> bevat meestal:

- Titel (<h2> of <h3>)
- Tekst (<p>)
- Eventueel afbeelding, link, datum

```
<article>
  <h2>Blogpost 1</h2>
  <p>Dit is mijn eerste artikel.</p>
</article>

<article>
  <h2>Blogpost 2</h2>
  <p>Vandaag leerden we over het article
element.</p>
</article>
```



HTML <aside> Element

Wist je dat?

Het <aside> element geschikt is voor extra uitleg.

Goed gebruiken voor:

- Links die relevant zijn
- Extra informatie
- Tips, weetjes, quotes
- Navigatie aan de zijkant

```
<aside>
 <h3>Gerelateerde links</h3>
 <ul>
  <li>Wat is HTML?</li>
  <li>HTML Basisstructuur</li>
  <li>HTML Elementen en Tags</li>
  <li>HTML Oefeningen voor Beginners</li>
 </ul>
</aside>
```

The <footer> Element

Wat is het <footer> element?

Het <footer>-element wordt gebruikt voor de **onderkant van een pagina of onderdeel**.

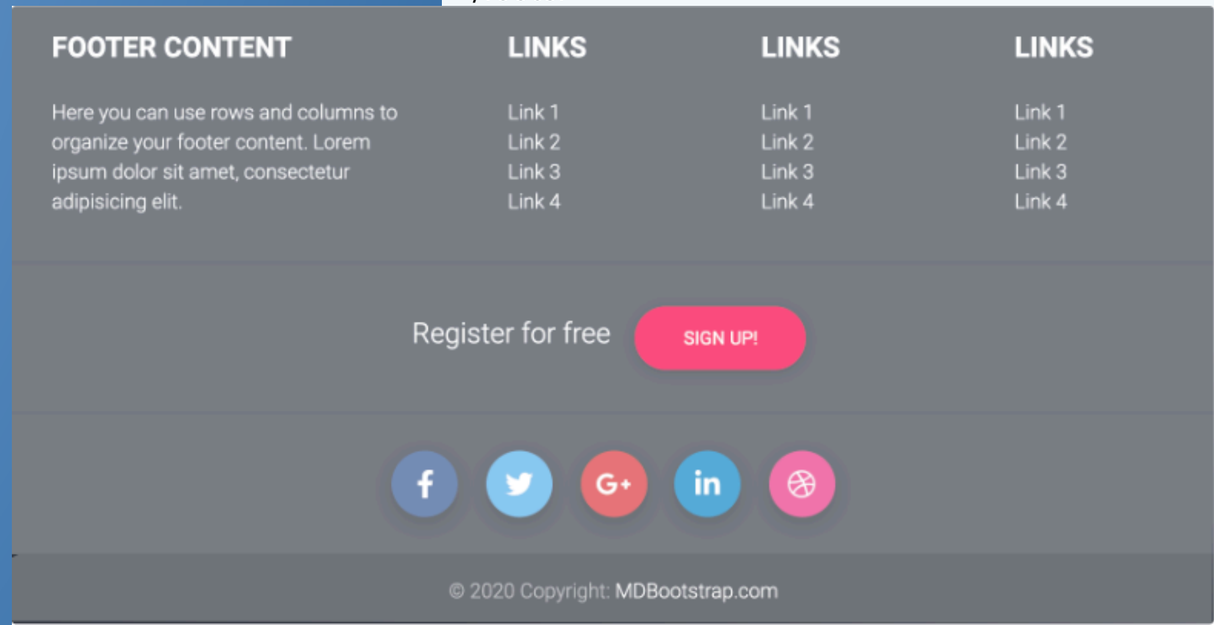
Het bevat meestal informatie zoals:

- 📌 copyright
- 📌 contactgegevens
- 📌 links
- 📌 kleine tekst
- 📌 auteur-informatie
- 📌 datum
- 📌 sociale media icoontjes

✓ **Waar wordt <footer> gebruikt?**

- 1× als **pagina-footer** (onderaan de website)
- Meermaals binnen **articles** of **sections**
- (bijv. onder een blogpost: datum + auteur)

```
<footer>
  <p>Gepubliceerd: 5 januari 2024 – Auteur: Janan</p>
</footer>
```



The Element

Wat is ?

Het element is een **algemene container** in HTML.

Het heeft **geen semantische betekenis** en wordt gebruikt om content te groeperen.

Met andere woorden:

👉 = een **lege doos**

👉 Je geeft het betekenis met **CSS** of door **inhoud erin te plaatsen**

Het is één van de meest gebruikte elementen in webdesign.

✔ **Waar gebruik je voor?**

- Groeperen van tekstblokken
- Maken van layout-structuren
- Groeperen van afbeeldingen
- CSS styling toepassen op een groep elementen
- Als onderdeel van een grid of flexbox layout

```
<div class="container">
  <div class="left">Links</div>
  <div class="right">Rechts</div>
</div>
```

The <main> Element

Wat is het <main> element?

Het <main>-element bevat de **belangrijkste inhoud** van een webpagina.
Dit is de inhoud die het **hoofdonderwerp** van de pagina vormt.

Het moet **uniek** zijn:

➡️ Je mag **maar één <main>** per pagina gebruiken.

Voorbeelden van hoofdinhoud:

- Artikelen
- Blogposts
- Productoverzicht
- Tekst van de pagina
- Foto's die de core content zijn

❌ **Wat hoort NIET in <main>?**

- Header
- Footer
- Navigatie
- Sidebar
- Logo's
- Copyright
- Reclame

```
<header>
  <h1>Mijn Website</h1>
</header>

<main>
  <h2>Diensten</h2>
  <p>Wij bieden webdesign, training en ondersteuning.</p>
</main>

<footer>
  <p>© 2024 DigiBrussel</p>
</footer>
```

The <input> Element

Wat is het <input> element?

Het <input> element wordt gebruikt om gegevens in te voeren in een formulier.

Gebruikers kunnen iets typen, kiezen of selecteren.

Het is één van de meest gebruikte HTML-elementen.

✔ Belangrijk:

- <input> heeft geen sluit-tag
- De functie hangt af van het type (via type="")

De meest gebruikte type="" waarden

Type

text
email
password
number
date
checkbox
radio
file
submite

```
<input type="email" placeholder="E-mailadres">

<input type="password" placeholder="Wachtwoord">

<input type="checkbox"> Ik ga akkoord
```

Wat zijn HTML Attributes?

HTML-attributen zijn speciale woorden die aanvullende informatie aan een HTML-element verstrekken. Attributen worden in de **openingstag** van het element geplaatst en worden gebruikt om het **gedrag van het element** te configureren of aan te passen. Alle attributen bestaan uit twee delen: **een naam en een waarde.**

N**aam**: De attribuutnaam is het sleutelwoord, ook wel **attribuut-ID** genoemd, dat een specifiek kenmerk definieert voor het element waarin het wordt gebruikt. Bijvoorbeeld: **href**(waarheen gaat een link) **src**(voor bestande die moet geladen worden).

Waarde: De attribuutwaarde is de data of informatie die je invult om dat attribuut in te stellen. De waarde wordt toegewezen tussen dubbele aanhalingstekens. Zo kunnen **"left", "center" of "right"** worden toegewezen aan het attribuut **"align"** met de alineatag (zoals weergegeven in het onderstaande voorbeeld).

Hieronder staat de syntaxis van een HTML-element met attribuut –

```
<tag_name attribute="Waarde">...</tag_name>
```

HTML Attributes

Alle HTML-elementen kunnen **attributen** hebben.
Attributen geven **aanvullende informatie** over elementen.
Attributen worden altijd gespecificeerd in de **starttag**.

The href Attribute

De <a>-tag definieert een hyperlink. Het href-attribuut bevat het webadres waar de link naartoe gaat:

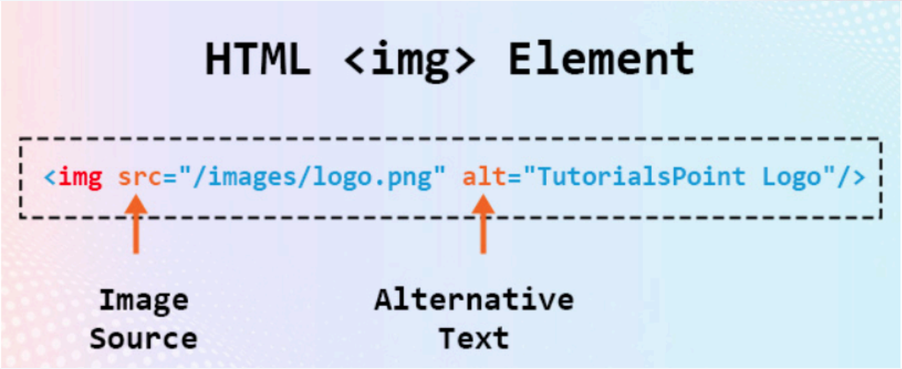
Voorbeeld:

```
<a href="https://www.w3schools.com">Visit W3Schools</a>
```

The src Attribute

De <img>-tag wordt gebruikt om een afbeelding in een HTML-pagina in te sluiten. Het src-attribuut specificeert het pad naar de weer te geven afbeelding:

voorbeeld:



Er zijn twee manieren om de URL in het src-attribuut op te geven:

1. Absolute URL - Linkt naar een externe afbeelding die op een andere website wordt gehost. Voorbeeld: src="https://www.w3schools.com/images/img_girl.jpg".

Opmerkingen: Externe afbeeldingen kunnen auteursrechtelijk beschermd zijn. Als u geen toestemming krijgt om ze te gebruiken, overtreedt u mogelijk de auteursrechtwetgeving. Bovendien hebt u geen controle over externe afbeeldingen; ze kunnen plotseling worden verwijderd of gewijzigd.

2. Relatieve URL - Is een pad naar een bestand op je eigen website. Je schrijft geen https://... of domeinnaam. De browser zoekt het bestand op basis van waar je huidige pagina staat. (/images/foto.jpg) : zoek vanaf de root (hoofdmappen) van je website.

Tip: HGebruik meestal relatieve URL's binnen je eigen website.
Ze zijn makkelijker bij testen (localhost) en blijven werken als je domein verandert, zolang je mapstructuur hetzelfde blijft.

The alt Attribute

Wat is het alt attribuut?

Het alt attribuut geeft een **tekstbeschrijving** van een afbeelding.

De tekst wordt zichtbaar wanneer:

- de afbeelding **niet kan laden**, of
- een gebruiker een **screenreader** gebruikt.

✓ **Waarom is het belangrijk?**

- **Toegankelijkheid** (voor blinden/slechtzienden)
- **SEO** (Google begrijpt de afbeelding)
- **Fallback** wanneer een afbeelding ontbreekt

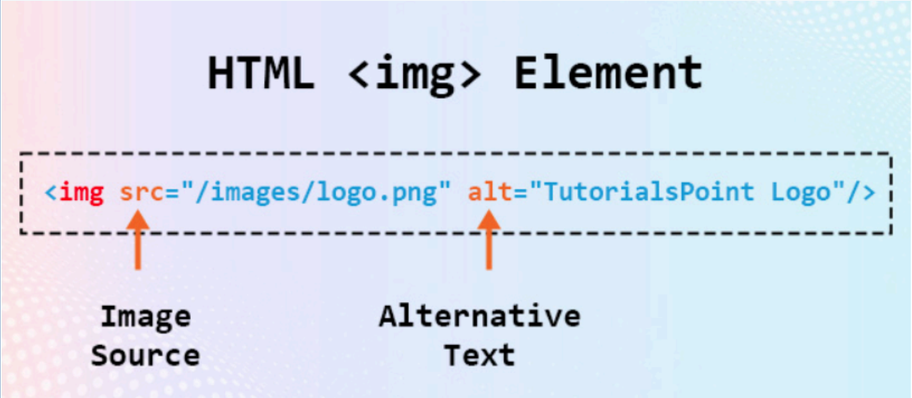
Wat moet er in een goede alt staan?

✓ Beschrijft wat er op de afbeelding staat

✓ Korte, duidelijke omschrijving

✓ Geen woorden zoals “foto van…” → dat weet de browser al

```
<img src="cat.png" alt="foto van een kat">
```



The style Attribute

Wat is het style attribuut?

Het style attribuut wordt gebruikt om **CSS rechtstreeks op één HTML-element** toe te passen.

Het wordt ook wel **inline CSS** genoemd.

✓ **Wanneer gebruiken?**

- Voor **kleine, éénmalige** stijlaanpassingen
- Voor **demo's** of **snelle tests**

✓ **Wanneer niet gebruiken?**

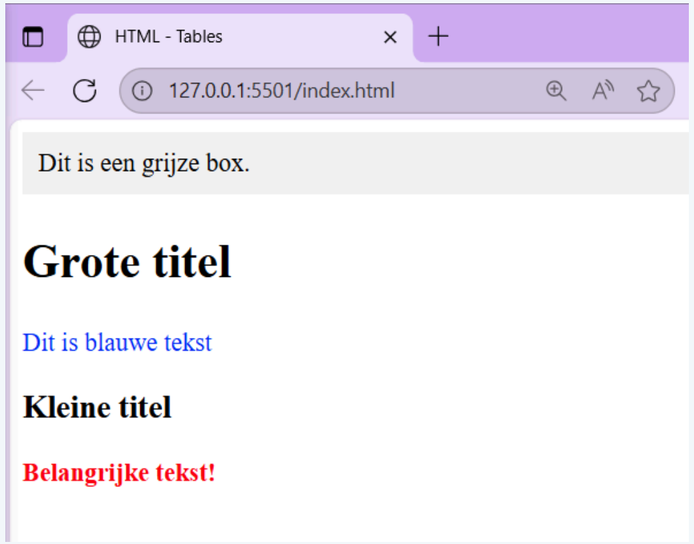
- Voor grote websites
- Voor herbruikbare stijlen
- Voor complete layouts

➡️ Dan moet je externe CSS gebruiken.

Met het style attribuut kun je:

color
background-color
font-size
font-weight
text-align
margin
padding
border

```
<div style="background-color: #f0f0f0; padding: 10px;">
 Dit is een grijze box.
</div>
<h2 style="font-size: 30px;">Grote titel</h2>
<p style="color: blue;">Dit is blauwe tekst</p>
<h3 style="font-size: 20px;">Kleine titel</h3>
<p style="color: red; font-weight: bold;">
 Belangrijke tekst!
</p>
```



The lang Attribute

Wat is het lang attribuut?

Het lang attribuut geeft aan **in welke taal een webpagina is geschreven.**

Het wordt geplaatst in de <html> **tag.**

✓ **Waarom is het belangrijk?**

- **Toegankelijkheid**: screenreaders lezen de correcte taal
- **SEO**: zoekmachines begrijpen je pagina beter
- **Vertaalsoftware**: browsers kunnen correcte automatische vertalingen maken
- **Internationale websites**: helpen bij taalherkenning

Waar zet je lang?

In de **openingstag van <html>**:

```
<html lang="en">
```

```
<!DOCTYPE html>
<html lang="nl">
<head>
  <meta charset="UTF-8">
  <title>Mijn Website</title>
</head>
<body>
  <h1>Welkom!</h1>
  <p>Dit is een voorbeeldpagina.</p>
</body>
</html>
```

```
<p lang="fr">Bonjour!</p>
<p lang="en">Hello!</p>
<p lang="nl">Hallo!</p>
```

The title Attribute

Wat is het title attribuut?

Het title attribuut geeft **extra informatie** over een HTML-element.
Deze informatie verschijnt als een **tooltip** wanneer je met de muis boven het element hangt.

✓ **Waar wordt het gebruikt?**

- Bij links
- Bij afbeeldingen
- Bij knoppen
- Bij tekst of iconen
- Bij elementen waar je extra uitleg wilt tonen

✓ **Hoe werkt het?**

Wanneer de gebruiker hovert met de muis → verschijnt kleine tekstballon.

Voorbeeld 1 — Link met title

```
<a href="#" title="Ga naar de homepage">Home</a>
```

Voorbeeld 2 — Afbeelding met title

```
<img src="cat.jpg" title="Een zwarte kat op de vensterbank">
```

Voorbeeld 3 — Button met title

```
<button title="Verzend het formulier">Submit</button>
```

Websites will realize that they do not need to do everything themselves. The Web is built on linking, and the Internet is ... well ... a network. These technologies are a perfect match for **letting other sites handle services that you don't want to do** yourself. Two examples that are already in place are outsourcing the acceptance of credit card payments and having a discussion forum hosted on another site. Currently, most large websites install their own search engines, but it would be easier to handle search through a link to an external search engine that was maintained by search exp display the search results on pages with the desired look and feel.

C|net article on discussion groups (and outsourcing of same)

As another example, all corporate websites need to give visitors directions to headquarters and other company facilities. There is no need for every site to design its own maps since there are sites that specialize in mapping services. Instead, give directions through an appropriate link to a preferred mapping service. Many of these services even provide customized directions from the individual user's starting point to the

The class Attribute

Wat is het class attribuut?

Het class attribuut wordt gebruikt om **HTML-elementen een naam te geven**, zodat ze:

- 📌 met **CSS** gestyled kunnen worden
- 📌 met **JavaScript** geselecteerd kunnen worden
- 📌 gegroepeerd kunnen worden (meerdere elementen met dezelfde class)

Het is één van de **belangrijkste** attributen in HTML.

Waarom een class gebruiken?

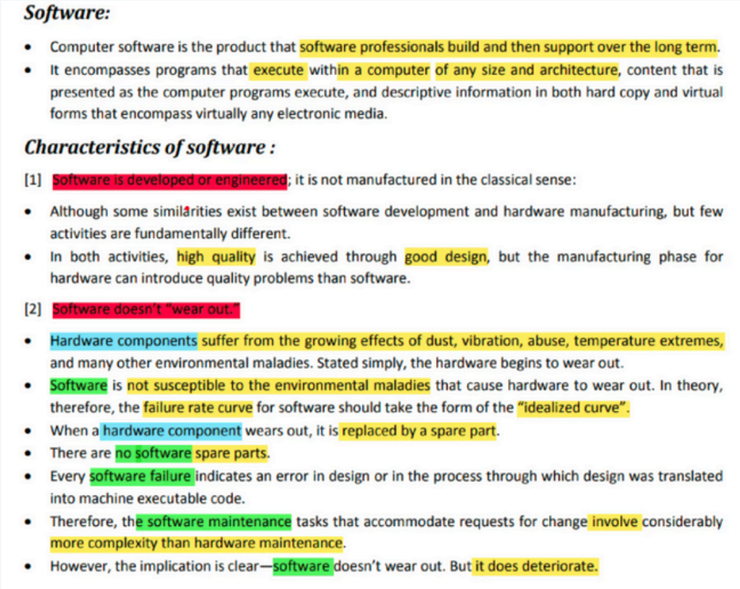
- Zelfde stijl toepassen op meerdere elementen
- Elementen groeperen (bijv. knoppen, kaarten, secties)
- Structuur maken in je layout
- Professioneel webdesign (CSS werkt met classes)

HTML

```html
<p class="highlight">Dit is belangrijke informatie.</p>
<h2 class="title">Over ons</h2>
<p class="title">Dit is een subtitel.</p>
<button class="primary-btn">Verzend</button>
```

CSS

```css
.highlight {
background-color: yellow;
}
.title {
color: blue;
font-size: 32px;
}
.primary-btn {
background: blue;
color: white;
padding: 10px;
}
```



Software:

- Computer software is the product that software professionals build and then support over the long term.
- It encompasses programs that execute within a computer of any size and architecture, content that is presented as the computer programs execute, and descriptive information in both hard copy and virtual forms that encompass virtually any electronic media.

Characteristics of software :

[1] Software is developed or engineered; it is not manufactured in the classical sense:

- Although some similarities exist between software development and hardware manufacturing, but few activities are fundamentally different.
- In both activities, high quality is achieved through good design, but the manufacturing phase for hardware can introduce quality problems than software.

[2] Software doesn't "wear out."

- Hardware components suffer from the growing effects of dust, vibration, abuse, temperature extremes, and many other environmental maladies. Stated simply, the hardware begins to wear out.
- Software is not susceptible to the environmental maladies that cause hardware to wear out. In theory, therefore, the failure rate curve for software should take the form of the "idealized curve".
- When a hardware component wears out, it is replaced by a spare part.
- There are no software spare parts.
- Every software failure indicates an error in design or in the process through which design was translated into machine executable code.
- Therefore, the software maintenance tasks that accommodate requests for change involve considerably more complexity than hardware maintenance.
- However, the implication is clear—software doesn't wear out. But it does deteriorate.

The id Attribute

Wat is het id attribuut?

Het id attribuut geeft **één element een unieke naam.**
Je mag dezelfde id **slechts één keer** per pagina gebruiken.

✓ **Waarvoor gebruik je id?**

- Voor unieke onderdelen op de pagina
- Voor CSS styling van één specifiek element
- Voor JavaScript (om iets te selecteren)
- Voor anker-links (bijv. naar een sectie scrollen)

✓ **Belangrijk:**

Een id moet **uniek** zijn → geen dubbele namen.

Voorbeeld:

- id="header" → één keer
- class="title" → meerdere keren

HTML

```
<h2 id="about">Over ons</h2>
```

CSS

```
#about {
  color: green;
  font-size: 28px;
}
```

JavaScript

```
document.getElementById("about").style.color = "red";
```

The Action Attribute

Wat is het action attribuut?

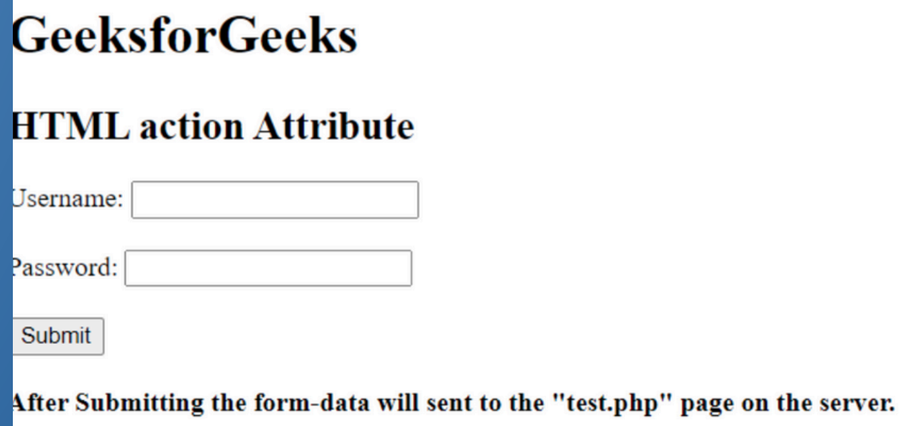
Het action attribuut bepaalt **waarheen** het formulier wordt **verstuurd** wanneer de gebruiker op Submit klikt.

Met andere woorden:
action = **de URL waar het formulier naartoe gaat**.

Deze URL kan zijn:

- een server-bestand (bijv. process.php)
- een API endpoint
- een pagina op dezelfde website
- een externe URL

```
<form action="/submit-form" method="post">
 <label for="username">Username:</label>
 <input type="text" id="username" name="username"
required>
 <br><br>
 <label for="email">Email:</label>
 <input type="email" id="email" name="email" required>
 <br><br>
 <input type="submit" value="Submit">
</form>
```



The Target Attribute

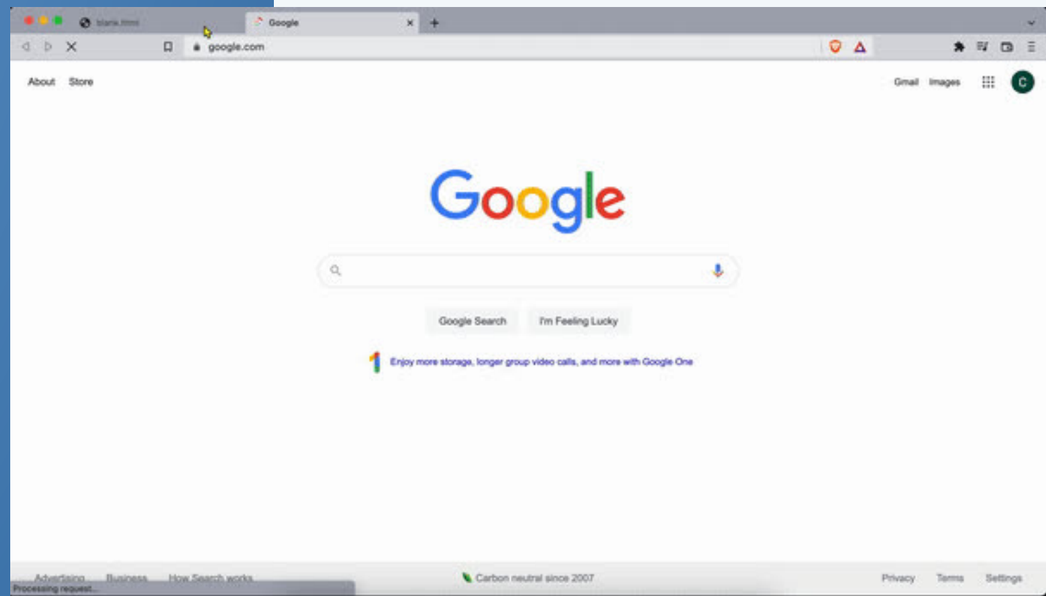
Wat doet het target attribuut?
Het target attribuut bepaalt waar een link geopend wordt wanneer iemand erop klikt.
Het wordt gebruikt in het <a> element:

```
<a href="..." target="...">Link</a>
```

De vier belangrijkste target-waarden:

_self Standaard (zelfde tab)
_blank Link openen in een nieuwe tab
_parent Voor frames
_top Openen in volledige pagina

```
<a href="about.html" target="_self">Over ons</a>
<a href="https://google.com" target="_blank">Bezoek Google</a>
<a href="page.html" target="_parent">Open bovenliggend frame</a>
<a href="index.html" target="_top">Home</a>
```



The Method Attribute

Wat doet het method attribuut?
Het method attribuut bepaalt **hoe** de gegevens van een formulier naar de server worden verstuurd.
Het wordt gebruikt in het <form> element:
<form method="post">

Er zijn 2 belangrijke methodes:

1. **method="GET"**
   - Stuurt de gegevens via de URL
   - URL wordt langer, zoals:
   - contact.html?naam=Janan&email=test@test.be
   - Sneller, simpel
   - Niet veilig voor gevoelige info

✓ **Wanneer gebruiken?**
- Zoeken
- Filters
- Niet-gevoelige gegevens
- Formulieren die geen privacy-informatie bevatten

2. **method="POST"**
   - Stuurt gegevens onzichtbaar (niet via de URL)
   - Geschikt voor gevoelige info
   - Grotere hoeveelheden data mogelijk
   - Beter voor echte formulieren

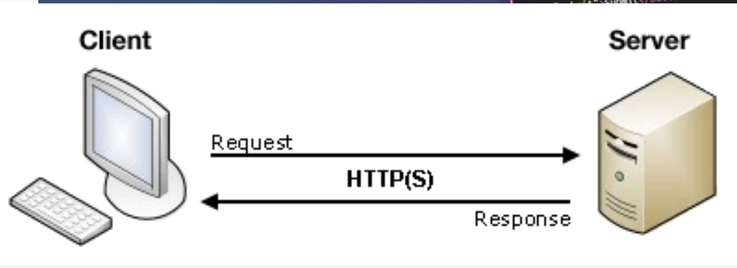
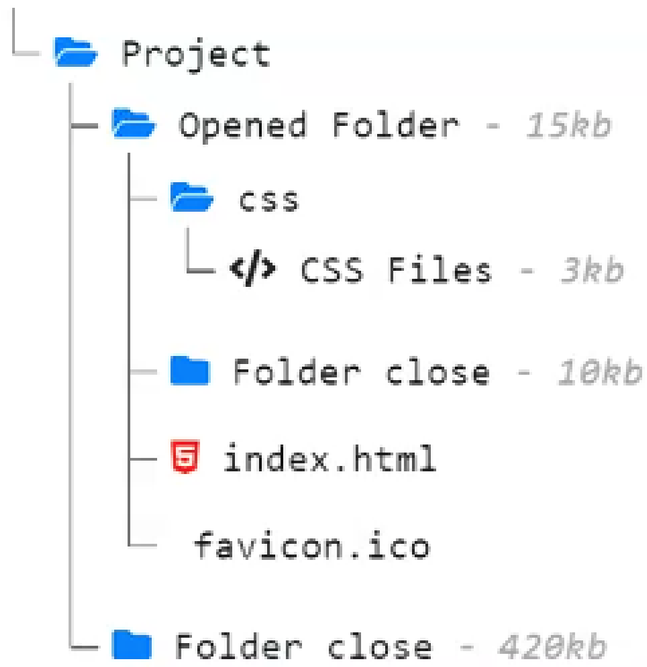
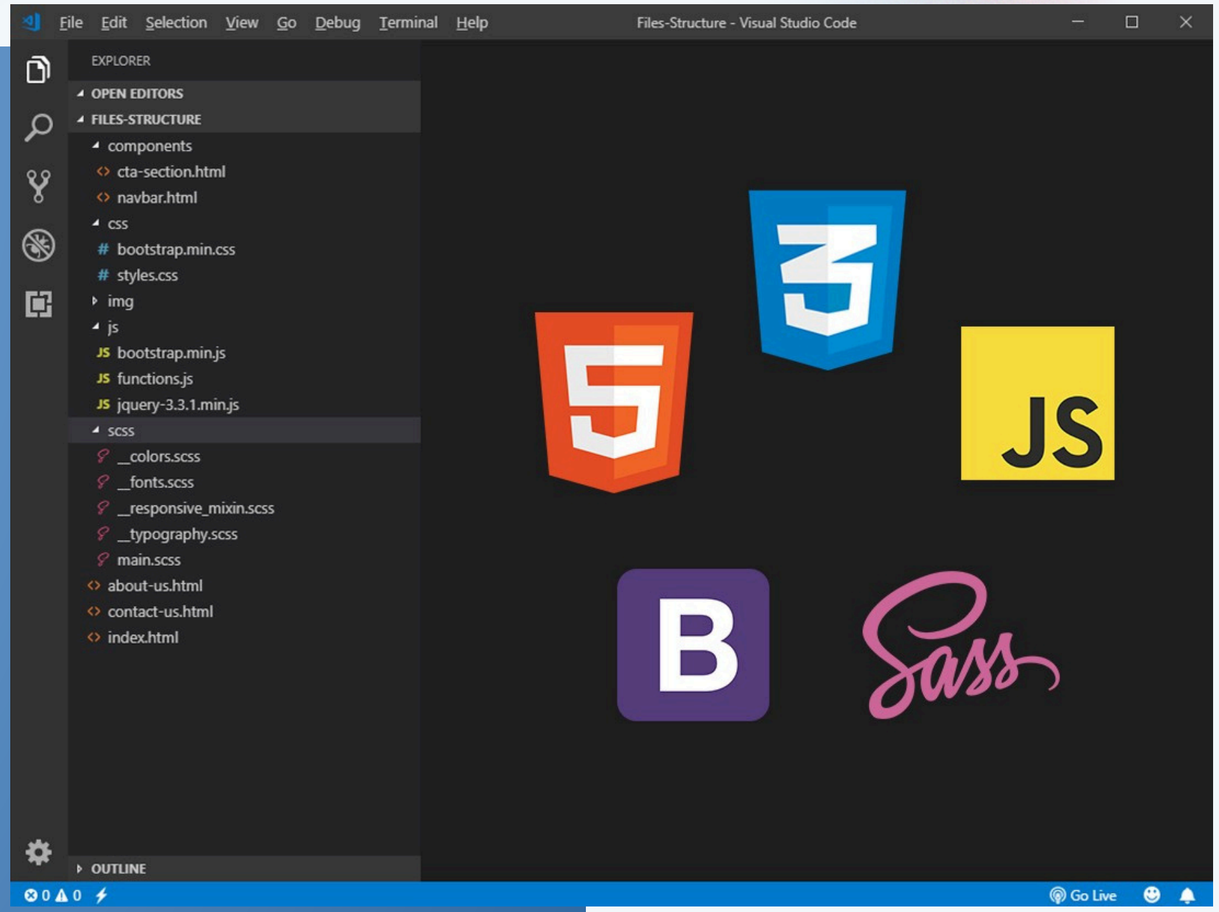
✓ **Wanneer gebruiken?**
- Contactformulieren
- Login / registreer formulieren
- Versturen van wachtwoorden
- Uploaden van bestanden
- Alles dat in een database terechtkomt

GET voorbeeld:

```html
<form action="zoek.html" method="get">
  <input type="text" name="zoekterm">
  <button type="submit">Zoek</button>
</form>
```

POST voorbeeld:

```html
<form action="verwerk.php" method="post">
  <input type="text" name="naam">
  <input type="email" name="email">
  <button type="submit">Verzend</button>
</form>
```



HTML File Structure / Bestandsstructuur