CSS - Phoenix

1 Wat is CSS?

datasource

In deze les leer je wat CSS is, waarom het belangrijk is en hoe je het gebruikt om de opmaak van een webpagina aan te passen. Je gaat zelf aan de slag met eenvoudige kleuren en lettertypes.

? Leerdoelen

- Je weet wat CSS doet en waarom we het gebruiken
- Je kunt een eenvoudige opmaakregel toevoegen via de HTML-tag (inline)
- Je kunt kleuren en lettertypes toepassen met CSS

? Wat is CSS?

- **CSS** staat voor **Cascading Style Sheets**
- Met CSS bepaal je hoe HTML eruitziet: kleuren, lettertypes, ruimte, achtergrond, enzovoort
- Zonder CSS is een website saai: zwart-wit, standaardletters, geen stijl

? Hoe voeg je CSS toe?

- Je kunt CSS op drie manieren gebruiken:
  1. **Inline** – direct in het HTML-element (in deze les)
  2. **In een <style>-blok** – in de `<head>` (volgende les)
  3. **In een extern .css-bestand** – voor grote projecten (komt later)

Voorbeeld: CSS inline gebruiken

```
<p style="color: red;">Deze tekst is rood</p>
<h1 style="font-family: Arial;">Welkom</h1>
```

De `style="..."` zit in het HTML-element en verandert de stijl van dat element.

?CSS Introduction

In deze korte video wordt uitgelegd wat CSS is. Kijk de video en als je nog niet alles begrijpt dan is dan prima, want we gaan alles ook nog een keer in deze lessen uitleggen.

https://www.youtube.com/watch?v=AGDDdsiZ0Ko&list=PLP9IO4UYNF0UCaUSF3XNZ1U9f01E5h5PM

https://www.youtube.com/embed/AGDDdsiZ0Ko?list=PLP9IO4UYNF0UCaUSF3XNZ1U9f01E5h5PM

?? Opdracht – Pas kleur en stijl toe

Let op: gebruik Phoenix Code om deze opdracht te maken.

1. Maak een nieuw HTML-bestand aan, noem het `css-les1.html`
2. Plak onderstaande code in het bestand:

```
<!DOCTYPE html>
<html>
<head>
  <title>Mijn eerste CSS</title>
</head>
<body>

  <h1>Welkom op mijn pagina</h1>
  <p>Ik ben een beginnende webdeveloper.</p>
  <p>Ik leer nu hoe ik CSS kan gebruiken.</p>

</body>
</html>
```

3. Pas de HTML aan met **inline CSS**, zodat:
   - De `<h1>` een blauwe kleur krijgt en een ander lettertype (bijv. Arial)

- De eerste `<p>` rood wordt en gecentreerd staat
- De tweede `<p>` een andere kleur krijgt die jij mooi vindt

4. Bekijk je werk in Phoenix Code via **Run**

? Inleveren

1. Maak een screenshot van jouw pagina met de aangepaste kleuren en lettertypes en zorg dat de code zichtbaar is.

2 CSS in het style-blok

In deze les leer je hoe je CSS los van HTML schrijft in het <style>-blok boven in het document. Je gebruikt selectoren zoals `body`, `h1` en `p` om elementen op te maken.

? Leerdoelen

- Je begrijpt waarom het slim is om CSS apart te schrijven
- Je kunt stijlen toevoegen in het <style>-blok
- Je kunt opmaak toepassen op meerdere elementen tegelijk met selectors

? Wat is het <style>-blok?

- Het `<style>`-blok zet je bovenin de pagina, tussen de `<head>`-tags
- Je schrijft daar CSS die geldt voor de hele pagina
- Je gebruikt een **selector** om aan te geven welk element je wilt opmaken. Hieronder vind je de selectors **body h1** en **p**

Voorbeeld:

```
<style>
  body {
    background-color: lightyellow;
  }

  h1 {
```

```
    color: navy;
    font-family: Verdana;
  }

  p {
    color: darkgreen;
    text-align: center;
  }
</style>
```

> In dit voorbeeld zijn body, h1 en p de **element**-selectors en die verwijzen naar de HTML-**elementen**.

?? Opdracht – Stijl je pagina met het <style>-blok

1. Maak een nieuw HTML-bestand in Phoenix Code, noem het `css-les2.html`
2. Plak deze basiscode in het bestand:

```
<!DOCTYPE html>
<html>
<head>
  <title>Stijl oefenen</title>
  <style>

  /* Voeg hier je CSS toe */

  </style>
</head>
<body>

  <h1>Over mij</h1>
  <p>Ik ben een student die leert over HTML en CSS.</p>
  <p>Ik vind het leuk om websites te maken.</p>

</body>
</html>
```

3. Voeg in het `<style>`-blok CSS toe zodat:
    - De `body` een lichtgrijze achtergrondkleur krijgt
    - Alle `<h1>`-koppen blauw worden en een ander lettertype krijgen (bijv. Georgia)
    - De `<p>`-tekst donkergroen wordt en gecentreerd staat

? Reflectie

- Wat is het voordeel van een `<style>`-blok boven inline stijl?
- Welke stijlen gelden voor de hele pagina, ook als je later extra tekst toevoegt?
- Wat zou je veranderen om je pagina nog mooier te maken?

Opdracht

1. Voer alle stappen uit zoals hierboven beschreven.

? Inleveren

1. Maak een screenshot van jouw pagina waarin je aangepaste stijlen duidelijk zichtbaar zijn

3 Tekstopmaak en achtergronden

In deze les leer je hoe je tekst kunt opmaken met kleur, lettertype, uitlijning en hoe je een achtergrondkleur toevoegt aan je pagina.

? Leerdoelen

- Je kunt tekst centreren of links/rechts uitlijnen
- Je kunt de kleur en het lettertype van tekst aanpassen
- Je kunt de achtergrondkleur van de hele pagina of een specifiek element aanpassen
- Je weet wat een selector is en wat het verschil is tussen een **class-selector** en een **element-selector**.

Selectors; element-selector en class selector

Zoals we eersder zagen beginnen we een **CSS blok** met een **selector**.

Tot nu toe hebben we **element-selectors** gebruikt. Deze verwezen naar de HTML-**elementen**.

We kennen ook **class-selectors** deze verwijzen naar **classes**. Classes zien er in HTML als volgt uit:

```
<div class="blok"> ... </div>
```

In dit geval is blok de naam van de selector.

In CSS verwijzen we naar een class op de volgende manier.

```
.blok {
  /* stijlregels */
}
```

> Een class selector in CSS begint met een . (punt)

? Belangrijke CSS-eigenschappen in deze les

- `color` – de kleur van de tekst
- `font-family` – het lettertype
- `text-align` – de uitlijning van tekst (bijv. `center`)
- `background-color` – de achtergrondkleur van een element

Voorbeeld:

```
.my-header {
  color: darkred;
  font-family: 'Comic Sans MS';
  text-align: center;
  background-color: lightyellow;
}
```

?? Opdracht – Maak een kleurrijke poster

1. Maak een nieuw HTML-bestand in Phoenix Code, noem het `poster.html`
2. Plak deze HTML-code in het bestand:

```
<!DOCTYPE html>
<html>
<head>
  <title>Mijn poster</title>
  <style>

  /* Voeg hier jouw stijlen toe */

  </style>
</head>
<body>

  <h1>Mijn favoriete onderwerp</h1>
  <p class="eerste">Dit is een stukje tekst over iets wat ik leuk vind.</p>
  <p>Bijvoorbeeld een sport, hobby, film of spel.</p>
  <p>Of een vak op College Amstelland dat je heel leuk vind :)</p>

</body>
</html>
```

3. Geef je `body` een leuke `background-color`
4. Geef je `h1` een opvallende kleur, een ander lettertype en `text-align: center`
   Gebruik hiervoor de **class** met **class-selector** `.header`
5. Geef de `p`-teksten elk een andere kleur en uitlijning en gebruik hiervoor **classes** met **class-selectors**. Geef elke <p> hiervoor een aparte class.
6. Experimenteer met verschillende kleuren en lettertypes zoals: `Arial`, `Georgia`, `Courier New`, `Comic Sans MS`

? Inleveren

1. Maak een screenshot van jouw kleurrijke poster waarin de CSS code ook duidelijk te lezen is.

4 Box-model: randen, padding en margins

In deze les leer je hoe je ruimte kunt maken rondom en binnenin een element met CSS. Je leert ook hoe je randen toevoegt en de afstand tussen blokken regelt.

? Leerdoelen

- Je begrijpt het verschil tussen **margin**, **padding** en **border**
- Je kunt ruimte tussen tekstblokken instellen met CSS
- Je kunt randen toevoegen aan blokken om ze beter zichtbaar te maken

? Wat is het box-model?

Elk HTML-element is als een ‘doos’ opgebouwd uit vier lagen:

- **Content:** de tekst of afbeelding binnenin
- **Padding:** ruimte tussen de content en de rand
- **Border:** de rand om het element heen
- **Margin:** de ruimte tussen dit element en andere elementen

image.png

Voorbeeld:

```
.blok {
  margin: 15px;
  border: 2px solid black;
  padding: 20px;
  background-color: lightblue;
}
```

?? Opdracht – Drie gekleurde tekstblokken

1. Maak een nieuw HTML-bestand in Phoenix Code, noem het `boxmodel.html`

2. Plak onderstaande code in je bestand:

```
<!DOCTYPE html>
<html>
<head>
  <title>Box model test</title>
  <style>

  .blok {
    border: 2px solid black;
    padding: 20px;
    margin: 20px;
    background-color: lightyellow;
  }

  </style>
</head>
<body>

  <div class="blok">
    <h2>Blok 1</h2>
    <p>Dit is het eerste tekstblok met padding en marge.</p>
  </div>

  <div class="blok">
    <h2>Blok 2</h2>
    <p>Ook dit blok heeft een rand en achtergrondkleur.</p>
  </div>

  <div class="blok">
    <h2>Blok 3</h2>
    <p>Pas de kleuren en afstanden aan zoals jij het mooi vindt.</p>
  </div>

</body>
</html>
```

3. Pas de CSS aan zodat:
    - Ieder blok een andere `background-color` heeft

- De `margin` tussen de blokken groter of kleiner is (proberen met `10px`, `30px`, enz.)
- De `padding` meer of minder ruimte geeft binnen het blok

4. Test verschillende `border`-instellingen, zoals `dashed`, `double` of `4px solid red`

? Reflectie

- Wat is het verschil tussen `margin` en `padding`?
- Waarom is het handig om borders tijdelijk toe te voegen tijdens het bouwen?
- Wat gebeurt er als je `padding` op 0 zet?

? Inleveren

1. Geef antwoord op de drie reflectievragen in eigen woorden en lever dat in, in een txt- of PDF bestand.
2. Voeg een schermafdruk van jouw boxen toe aan de PDF.

5 Classes gebruiken in CSS

In deze les leer je hoe je een `class` gebruikt om meerdere elementen dezelfde opmaak te geven. Zo kun je stijlen hergebruiken zonder telkens dezelfde regels te schrijven.

? Leerdoelen

- Je weet wat een `class` is en hoe je die toepast in HTML
- Je kunt CSS-stijlen koppelen aan een class
- Je kunt verschillende klassen maken en gebruiken op één pagina

? Wat is een class?

- Een `class` is een naam die je aan een element geeft zodat je het in CSS kunt opmaken
- Je zet in HTML: `<div class="menu">`
- In CSS gebruik je een punt (.) om een class aan te roepen: `.menu { ... }`
- Je kunt dezelfde class meerdere keren gebruiken.

- Een HTML element kan ook meerdere classes bevatten: `<div class="menu speciaal">`
  De stylen met de class-selector menu én speciaal zijn beiden van teopassing op dit element.

Voorbeeld:

```
<div class="menu">Pizza Margherita</div>
<div class="menu speciaal">Spaghetti Bolognese</div>
```

```
.menu {
  background-color: lightyellow;
  padding: 10px;
  border: 1px solid gray;
  margin-bottom: 10px;
}

.speciaal {
  color: red;
}
```

?? Opdracht – Bouw een gestyleerde menukaart

1. Maak een nieuw HTML-bestand in Phoenix Code, noem het `menukaart.html`
2. Plak onderstaande code als startpunt:

```
<!DOCTYPE html>
<html>
<head>
  <title>Menukaart</title>
  <style>

  .menu {
    background-color: lightyellow;
    border: 2px solid #999;
    padding: 15px;
    margin-bottom: 15px;
    font-family: Georgia;
  }
```

```
  </style>
</head>
<body>

  <h1>Restaurant De Codepan</h1>

  <div class="menu">Pizza Margherita</div>
  <div class="menu">Truffelpasta met parmezaan</div>
  <div class="menu">Ravioli met spinazie en ricotta</div>
  <div class="menu">Chef’s surprise</div>

</body>
</html>
```

3. Pas zelf de kleuren, marges of lettertypes aan zoals jij het mooier vindt
4. Voeg minstens **twee** eigen gerechten toe aan de lijst
5. Geef **twee** menu-items de **class** `speciaal` en zorg ervoor dat deze speciale gerechten duidelijk opvallen door hun stijl aan te passen.

? Reflectie

- Waarom is het handig om stijlen te herhalen met een class?
- Wat gebeurt er als je meerdere klassen aan één element geeft?
- Kun je bedenken wanneer je liever een class gebruikt dan een tag-selector (zoals `p`)?

? Inleveren

1. Maak een screenshot van jouw gestyleerde menukaart in de browser.

6 Layout en positionering

In deze les leer je hoe je met behulp van blokken en eenvoudige CSS een duidelijke indeling (layout) maakt. Denk aan een header bovenaan, een middenstuk (main) en een footer onderaan.

? Leerdoelen

- Je weet hoe je de layout van een pagina opbouwt met blokken (divs)
- Je kunt tekst en blokken centreren met `text-align` en `margin`
- Je kunt blokken op een vaste breedte zetten

? Wat is een layout?

- Een layout is de structuur van je pagina – hoe dingen verdeeld zijn over het scherm
- Voor layout gebruiken we vaak het HTML-element `<div>`
- Een `div` is een onopvallend blok dat je zelf kunt opmaken met CSS

Belangrijke CSS-eigenschappen:

- `text-align: center;` – centreren van tekst
- `margin: auto;` – centreren van blokken
- `width:` – bepaalt hoe breed een blok is (bijv. `80%` of `600px`)

Voorbeeld:

```
.blok {
  width: 80%;
  margin: 20px auto;
  padding: 20px;
  background-color: #f0f0f0;
  text-align: center;
}
```

?? Opdracht – Bouw een eenvoudige homepage-layout

1. Maak een nieuw HTML-bestand in Phoenix Code, noem het `layout.html`
2. Plak onderstaande basiscode in het bestand:

```
<!DOCTYPE html>
<html>
<head>
```

```
  <title>Mijn layout</title>
  <style>

    .blok {
      width: 80%;
      margin: 20px auto;
      padding: 20px;
      background-color: #eeeeee;
      text-align: center;
    }

    .header {
      background-color: lightblue;
    }

    .main {
      background-color: white;
    }

    .footer {
      background-color: lightgray;
    }

  </style>
</head>
<body>

  <div class="blok header">
    <h1>Welkom op mijn site</h1>
  </div>

  <div class="blok main">
    <p>Hier komt de inhoud van je website.</p>
  </div>

  <div class="blok footer">
    <p>Gemaakt door [jouw naam] - 2025</p>
  </div>

</body>
```

```
</html>
```

3. Pas de kleuren aan zodat het bij jouw stijl past
4. Voeg een afbeelding toe in de main-blok en plaats jouw naam onderaan in de footer.
5. Geef de blokken afgeronde hoeken (`border-radius`) van bijvoorbeeld `15px`

? Reflectie

- Wat gebeurt er als je de `width` verandert naar `60%` of `400px`?
- Hoe werkt `margin: auto` precies?
- Wat zou je toevoegen als je een echte website maakt?

? Inleveren

1. Maak een screenshot van jouw homepage met drie blokken die door jouw zijn vormgegeven.

7 CSS-selectors: element, class en id

In deze les leer je wat een `id`-selector is en wanneer je die gebruikt. Je herhaalt ook wat een `element`-selector en een `class`-selector doet. Daarna pas je deze drie selectors toe in één pagina.

? Leerdoelen

- Je herkent het verschil tussen element-, class- en id-selectors
- Je weet wanneer je een id gebruikt in plaats van een class
- Je kunt deze drie selectors toepassen in één pagina

? Wat zijn selectors in CSS?

Een **selector** in CSS bepaalt op welk HTML-element de stijlregel toegepast wordt.

Selector	Voorbeeld	Toepassing
Element	`p { color: blue; }`	Geldt voor **alle** paragrafen

Selector	Voorbeeld	Toepassing
Class	`.belangrijk { color: red; }`	Geldt voor **meerdere** elementen met class="belangrijk"
ID	`#titel { font-size: 30px; }`	Geldt voor **één uniek element** met id="titel"

Belangrijk verschil:

- `class` gebruik je voor meerdere elementen
- `id` gebruik je maar één keer op een pagina – het is uniek

?? Opdracht – Gebruik drie soorten selectors

1. Maak een nieuw HTML-bestand in Phoenix Code, noem het `selectors.html`
2. Plak deze code als basis:

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS Selectors</title>
  <style>

    /* Element-selector */
    p {
      color: green;
    }

    /* Class-selector */
    .special {
      font-weight: bold;
      background-color: lightyellow;
    }

    /* ID-selector */
    #titel {
      font-size: 30px;
      color: navy;
    }
```

```
  </style>
</head>
<body>

  <h1 id="titel">Welkom op mijn pagina</h1>

  <p>Dit is een normale paragraaf.</p>
  <p class="special">Dit is een speciale paragraaf.</p>
  <p>Nog een gewone paragraaf.</p>

</body>
</html>
```

3. Pas de stijlen aan zoals jij het mooi vindt:
   - Geef de `#titel` een andere kleur en een ander lettertype
   - Geef de class `.special` een rand of andere achtergrond
4. Voeg zelf nog een extra class toe voor een andere stijl (bijv. `.opvallend`)

? Reflectie

- Wanneer gebruik je een `class` en wanneer een `id`?
- Wanneer gebuik je een element-selector, bijvoorbeeld `body` of `h1`?
- Wat gebeurt er als je per ongeluk twee keer hetzelfde `id` gebruikt?

? Inleveren

1. De refectievragen en jouw eigen antwoorden (eigen woorden) in een txt- of PDF bestand.

8 Flexbox: moderne layouttechniek

In deze les leer je werken met `flexbox`. Dit is een moderne techniek waarmee je blokken makkelijk naast of onder elkaar zet, zonder ingewikkelde marges of floats.

? Leerdoelen

- Je weet wat `display: flex` doet
- Je kunt blokken naast elkaar zetten
- Je weet wat resonsive design is.
- Je kunt blokken centreren en de ruimte verdelen met `justify-content` en `align-items`

? Wat is flexbox?

Met flexbox kun je elementen binnen een container netjes ordenen, zowel horizontaal als verticaal.

- `display: flex` zet de container om in een flex-container
- `justify-content` bepaalt de horizontale uitlijning (zoals `center`, `space-between`)
- `align-items` bepaalt de verticale uitlijning (zoals `center`, `flex-start`)

Met felxboxes maak je een **responsive design**: dit betekent dat je web pagina op verschillende aparaten goed wordt getoond en dat de boxes worden aangepast als de breedte van het scherm wordt aangepast.

Voorbeeld CSS:

```
.container {
  display: flex;
  justify-content: space-between;
  align-items: center;
}

.blok {
  width: 150px;
  height: 150px;
  background-color: lightblue;
  text-align: center;
  padding: 20px;
  border-radius: 10px;
}
```

> Een complete uitleg over alle mogelijkheden vind je hier: https://css-tricks.com/snippets/css/a-guide-to-flexbox/

?? Opdracht – Maak een blokkenrij met Flexbox

1. Maak een nieuw bestand in Phoenix Code: `flex.html`
2. Gebruik deze HTML als basis:

```
<!DOCTYPE html>
<html>
<head>
  <title>Flexbox oefening</title>
  <style>

    .container {
      display: flex;
      justify-content: space-between;
      align-items: center;
      background-color: #f0f0f0;
      padding: 20px;
    }

    .blok {
      width: 150px;
      height: 150px;
      background-color: lightcoral;
      color: white;
      font-weight: bold;
      border-radius: 10px;
      display: flex;
      justify-content: center;
      align-items: center;
    }

  </style>
</head>
<body>

  <h1>Mijn Flexbox layout</h1>

  <div class="container">
    <div class="blok">Blok 1</div>
```

```
    <div class="blok">Blok 2</div>
    <div class="blok">Blok 3</div>
  </div>

</body>
</html>
```

3. Experimenteer met andere waarden voor `justify-content` zoals `center`, `space-around` of `flex-end`
4. Voeg eventueel nog een vierde blok toe
5. Pas de kleuren en tekst aan zodat het jouw stijl heeft

??Opdracht

Maak het volgende ontwerp zo goed mogelijk na en gebruik daarbij Flexboxen zoals je dat net hebt geleerd.:

image.png

De drie blokken staan op het midden van de webpagina en je design is **responisve**.

? Inleveren

1. Maak een screenshot van je resultaat (van de *laatste* opdracht).
2. Indien je AI hebt gebruikt, stuur dan de AI-Chat log mee en zorg ervoor dat je alleen zaken gebruikt die we hebben behandeld en die je kan uitleggen.

> Je kan worden gevraagd om jouw ontwerp te komen toe lichten / uitleggen.

--

Revision #19
Created 2025-06-04 11:52:07 UTC by Max
Updated 2026-07-08 17:23:15 UTC by Max