



Cursus Sass SCCS Basics

Web Advanced

Inhoudsopgave

1	Inleiding Sass	3
1.1	Eigenschappen van Sass	3
1.1.1	<i>Schaalbaar, herbruikbaar en slim</i>	3
1.1.2	Verschillende syntaxen	4
1.1.3	Compileerbaar	4
1.2	Extra benodigdheden.....	5
1.2.1	Node.js.....	5
1.2.2	Node-sass.....	5
2	Starten met een project	6
2.1	Node-sass gebruiken	6
3	Basisfunctionaliteiten van Sass.....	7
3.1	Variabelen.....	7
3.2	Nesting	8
3.3	Mixins	9
3.4	Partials en Imports	9
4	Meer informatie	10
5	Oefening	11

1 Inleiding Sass

Synthactically Awesome Style Sheets of **Sass** is een extensie op CSS, een zogenaamde CSS Preprocessor, waarmee je jouw code flexibeler en krachtiger kan maken. Het laat je toe om CSS op een programmeerbare manier te schrijven. Zo kan je onder andere variabelen, functies en conditionele statements gebruiken.



Het is niet de enige beschikbare CSS Processor, maar wel de meest populaire. Naast Sass bestaat er bijvoorbeeld nog *Less* en *Stylus*.

1.1 Eigenschappen van Sass

1.1.1 *Schaalbaar, herbruikbaar en slim*

CSS bestaat al sinds de jaren '90 en is qua structuur al wat achterhaald. Hierdoor kan je tegen de volgende problemen aanlopen:

- In een website worden vaak dezelfde kleuren gebruikt. Wanneer je op een gegeven moment één van deze kleuren wil wijzigen, moet je deze doorheen het volledige project gaan vervangen.
- In een standaard CSS-bestand ga je vaak dezelfde codeblokken zien terugkomen. Er is geen manier om deze herhaling op te vangen.
- Hoe groter het project wordt, hoe meer verschillende klasse-namen er dienen gebruikt te worden. Hierdoor krijg je vaak complexe en lange CSS-selectoren.

Sass lost niet alleen de hierboven beschreven problemen op, maar maakt CSS ook schaalbaar, herbruikbaar en slim. Zo kan je bijvoorbeeld de CSS-code opsplitsen in verschillende bestanden zodat het project overzichtelijker wordt om in te werken en bijgevolg ook beter leesbaar zal zijn.

1.1.2 Verschillende syntaxen

Om met Sass te werken zijn er twee mogelijke conventies:

- **.scss (Sassy CSS)**: deze conventie volgt de algemene CSS bracket syntax.

```
body #main .value {  
  color: #eee;  
  
  a.link {  
    color: #101010;  
    text-decoration: none;  
  }  
}
```

- **.sass**: deze conventie werkt aan de hand van indentatie.

```
body #main .value  
  color: #eee  
  
  a.link  
    color: #101010  
    text-decoration: none
```

We zullen in deze cursus gebruik maken van scss omdat dit beter aansluit bij de CSS syntax.

1.1.3 Compileerbaar

Browsers kunnen out of the box geen Sass interpreteren. De geschreven code gaat dus eerst moeten omgezet worden naar gewone CSS voordat we deze aan de browser kunnen doorgeven. Het omzetten van deze code wordt transpileren (transpile) of compileren (compile) genoemd.

1.2 Extra benodigdheden

1.2.1 Node.js

Node.js is een cross-platform run-time environment gebaseerd op de open-source JavaScript en WebAssembly V8-engine van Google.

Via de npm (Node Package Manager) kunnen we dependencies installeren.

Downloadlink: <https://nodejs.org/en/download/>



1.2.2 Node-sass

Om Sass code om te zetten naar CSS code hebben we iets nodig dat dit voor ons kan doen. We gaan hiervoor een compiler installeren: het npm pakket '*node-sass*'.

Downloadlink: <https://www.npmjs.com/package/node-sass>



2 Starten met een project

Via volgend stappenplan kan je eenvoudig een nieuw project starten met Sass SCSS.

1) Open JetBrains Webstorm en maak een nieuw project aan.

2) Voeg volgende items aan het project toe:

- de map 'assets'.
- de map 'css' met een CSS-bestand 'style.css'.
- de map 'scss' met een SCSS-bestand 'style.scss'.
- het HTML-bestand 'index.html' met een koppeling naar de 'style.css'.

3) Open de terminal en voer volgende commando's in:

- initialiseer de Node Package Manager: `npm init -y`
- installeer node-sass: `npm install node-sass -g`

2.1 Node-sass gebruiken

Het node-sass pakket is via bovenstaand stappenplan geïnstalleerd binnen het project, maar werkt nog niet. We gaan er altijd voor moeten zorgen dat de Sass-code effectief wordt omgezet naar bruikbare CSS-code. Dit doen we via volgend commando:

```
node-sass -o css scss/style.scss --watch
```

Wanneer we het commando ontleden, kunnen we volgende onderdelen onderscheiden:

- **node-sass:** het pakket dat we hebben geïnstalleerd en hiermee gaan aanspreken.
- **-o:** een argument dat we meegeven zodat we output kunnen zien in de terminal.
- **css:** de map waarin de gecompileerde code moet terecht komen.
- **scss/style.scss:** het bronbestand van de code.
- **--watch:** voert iedere wijziging van de code automatisch door.

Eens je dit commando hebt uitgevoerd zal je in het main.css bestand zien dat de code is omgezet en ga je de wijzigingen dus ook in de browser kunnen zien.

3 Basisfunctionaliteiten van Sass

3.1 Variabelen

Sass maakt het mogelijk om op een makkelijke manier met variabelen te werken. Op die manier kan je een waarde aan een eigenschap koppelen. Een variabele kunnen we in Sass declareren via het dollarteken.

```
$primaire-kleur: #FF0000;
$secundaire-kleur: #CCCCCC;
$zwarte-kleur: #000000;
$witte-kleur: #FFFFFF;
$font-xl: 6rem;
$font-lg: 4rem;
$font-md: 2rem;
$font-sm: 1rem;
```

Het gebruik van variabelen gaat vooral interessant zijn wanneer we bepaalde waardes, zoals bijvoorbeeld kleuren, meerdere keren gaan gebruiken. Dien je na verloop van tijd de kleur aan te passen, dan moet je dat slechts op één plaats doen in plaats van doorheen heel het project de kleur te moeten zoeken en te vervangen.

```
body {
  background-color: $secundaire-kleur;
}

h1 {
  color: $primaire-kleur;
  font-size: $font-lg;
  font-weight: extra bold;
}

h2 {
  font-size: $font-md;
}
```

3.2 Nesting

Nesting of het in elkaar nesten van CSS selectoren is een goede en bruikbare manier die door Sass voorzien wordt. Het zorgt er namelijk voor dat we veel minder lijnen CSS code moeten schrijven en het is ook makkelijk in gebruik.

Het doel van nesting is dat vaak terugkomende niveaus maar één keer moeten gedefinieerd worden.

```
nav {  
  ul {  
    margin: 0;  
    padding: 0;  
    list-style: none;  
    li {  
      display: inline-block;  
      a {  
        display: block;  
        padding: 0.5rem 1rem;  
        text-decoration: none;  
        &:hover {  
          text-decoration: underline;  
        }  
      }  
    }  
  }  
}
```

Pseudo-klassen kunnen we in Sass ook makkelijk genest toevoegen. In bovenstaand voorbeeld hebben we een hover-effect toegevoegd aan ons a-element met behulp van het &-teken.

3.3 Mixins

Wanneer je gewone CSS schrijft, ga je zien dat bepaalde stukken code vaak gaan terugkomen. In Sass kunnen we die herhalingen opvangen in zogenaamde Mixins. Concreet zijn het dus codeblokken die herbruikbare CSS stijlen bundelen. Je kan het vergelijken met variabelen, maar dan krachtiger omdat je meerdere stijlen samen kan nemen.

Een mixin creëren doe je door het keyword '@mixin' te gebruiken gevolgd door de naam die je eraan wil geven. De stijlen zet je dan binnen accolades.

```
@mixin overlay() {  
    bottom: 0;  
    left: 0;  
    right: 0;  
    top: 0;  
    position: absolute;  
}
```

Eens je de mixin hebt aangemaakt, kan je die binnen een CSS selector gebruiken met het keyword '@include' gevolgd door de naam van de mixin.

```
.modal-background {  
    @include overlay();  
    background-color: $zwarte-kleur;  
    opacity: 0.9;  
}
```

3.4 Partials en Imports

Tot nu toe zijn we echter blijven werken in één bestand. In grote projecten is dit niet efficiënt om een bestand te hebben waar alle code in verzameld zit. Ook hier voorziet Sass in een oplossing om de code op te splitsen in kleinere zelf beschrijvende bestanden.

Partials zijn bestanden waarin we code gaan schrijven die van toepassing zijn op een bepaald gedeelte van de code. De naam van dergelijke bestanden laat je voorafgaan door een '_'. Dit zal er namelijk voor zorgen dat tijdens het compileren geen apart CSS bestand zal aangemaakt worden voor deze partial.

Op zich gaat deze code in eerste instantie niets doen omdat we het nergens linken of koppelen. Om er voor zorgen dat we deze code in het project kunnen gebruiken gaan we werken met imports. Om een partial in te laden in onze 'style.scss' gebruiken we het keyword '@import' en geven we hieraan het pad mee naar het bestand.

```
@import "reset";  
@import "navigation";  
@import "header";  
@import "main";  
@import "footer";  
@import "buttons";
```

4 Meer informatie

Deze basiscursus voorziet slechts inzicht in enkele eenvoudige functionaliteiten van Sass SCSS, maar met Sass SCSS kunnen we nog veel meer! Via onderstaande linken krijgen jullie meer informatie over het gebruik en de functionaliteiten van Sass SCSS.

<https://sass-lang.com/guide/>

<https://www.w3schools.com/sass/>

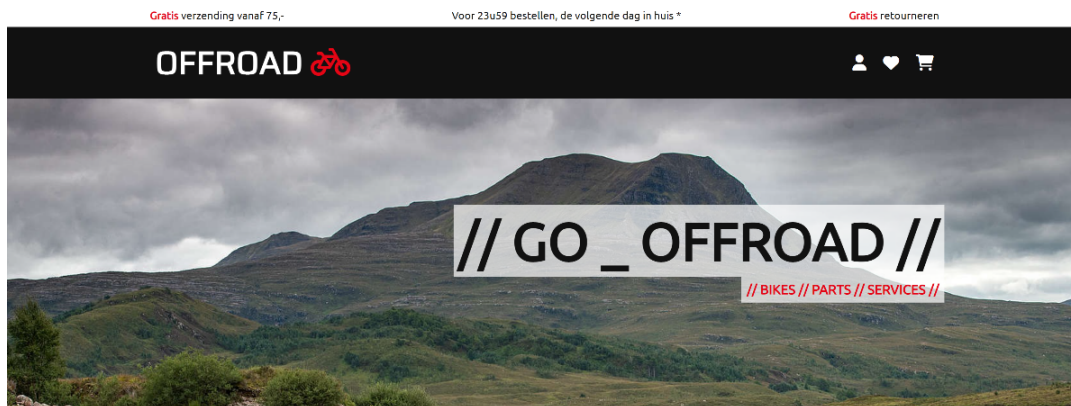
<https://www.freecodecamp.org/news/the-beginners-guide-to-sass/>

<https://www.youtube.com/watch?v=a5j7KoflTs>

<https://www.javatpoint.com/sass-tutorial>

5 Oefening

Maak de oefening van OFFROAD fietsenwinkel uit het opleidingsonderdeel 'Web Essentials' opnieuw. Schrijf de volledige opmaak via Sass SCSS en gebruik alle basisfunctionaliteiten die aan bod komen in deze mini-cursus.



// NEW IN //



LANA 27.5

Dé beste hardtail beginners-mountainbike voor vrouwen op de markt. Deze mountainbike is veilig, wendbaar en speels op de weg dankzij de lage bovenbuis en de 27.5" wielen. Een echte eyecatcher verkrijgbaar in diverse kleuren en met verschillende opties.

- LANA AL
- Rock Shox Judy TK R 100MM
- Shimano Deore RD-M1600 12-S

KATO 29

De KATO is een snelle, wendbare hardtail mountainbike die de bergen efficiënt opklimt. Het frame is volledig opnieuw ontworpen met interne bekabeling en een fris design. Slechts één ding is hetzelfde gebleven; de fun om ermee te rijden!

- KATO AL
- SR SUNTOUR XCR RL AIR 100MM
- Shimano Deore RD-M4120 10-S



// SALES + OUTLET //



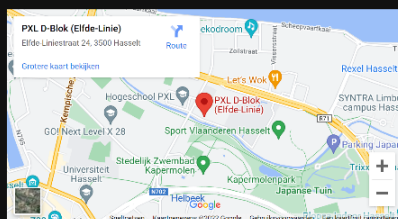
NIRVANA TOUR



LECTOR



LECTOR SF



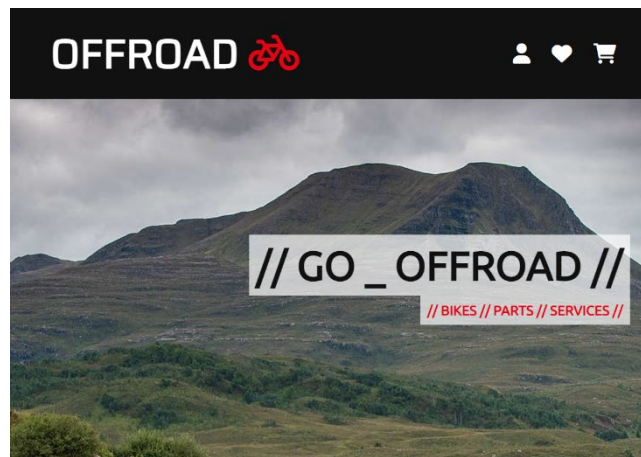
OFFROAD BIKE SHOP

Elfdre-Linestraat 24
3500 HASSELT
BELGIE
0452 / 63 51 42
011 / 32 65 87
info@offroad-bikes.be



Enkele tips

- Haal de nodige startbestanden (assets en tekst) af via Blackboard.
- Gebruik de Google Font lettertypes 'Ubuntu' en 'Saira'.
- Gebruik de kleuren #000000, #111111, #FFFFFF en #E40515.
- Werk met de meeteenheden 'rem', 'px' en '%'.- Voorzie een responsive design voor schermbreedtes tussen 2500px en 500px.
- Geef id's en classes namen via de 'BEM naming convention'.
- Gebruik de iconenbibliotheek 'Font Awesome'.
- Zorg voor hover-effecten waar nodig.
- Voeg Google maps toe in de footer via een iframe.
- *Je mag zelf kiezen om volledig opnieuw te beginnen aan deze oefening of om de verkregen CSS-code in het ZIP-bestand aan te passen naar SCSS-code.*



// NEW IN //

LANAO 27.5

Dé beste hardtail beginners-mountainbike voor vrouwen op de markt. Deze mountainbike is veilig, wendbaar en speels op de weg dankzij de lage bovenbuis en de 27.5" wielen. Een echte eyecatcher verkrijgbaar in diverse kleuren en met verschillende opties.

- LANA O AL
- Rock Shox Judy TK R 100MM
- Shimano Deore RD-M1600 12-5



KATO 29

De KATO is een snelle, wendbare hardtail mountainbike die de bergen efficiënt opklimt. Het frame is volledig opnieuw ontworpen met interne bekabeling en een fris design. Slechts één ding is hetzelfde gebleven; de fun om ermee te rijden!

- KATO AL
- SR SUNTOUR XCR RL AIR 100MM
- Shimano Deore RD-M4120 10-S



// SALES + OUTLET //



NIRVANA TOUR



LECTOR



LECTOR SF