# Oefeningen Onderdeel 3

# DE PIJLERS – DEEL 2

## OEFENING 1 – JavaScript bundling

Tracht de pipeline zelf eens op te zetten aan de hand van de slides. Maak een 3-tal javascript bestanden die elk iets loggen op de html pagina (gebruik misschien eens een <textarea>). Bundle deze 3 bestanden door middel van webpack.

Probeer je code eens te debuggen, wat valt je op?

## OEFENING 2 – SASS

Tracht de pipeline zelf eens op te zetten aan de hand van de slides. Maak een .scss bestand en zorg ervoor dat de pipeline dit omzet naar .css.

## OEFENING 3 – PostCSS

In deze oefening gebruik je geen preprocessor zoals SASS maar wel een postprocessor (PostCSS)​. We gaan eens een nieuwe CSS feature gebruiken: een ‘custom selector’ ([https://drafts.csswg.org/css-extensions/#custom-selectors](https://drafts.csswg.org/css-extensions/" \l "custom-selectors)). Dat is een uitbreiding op CSS die nog niet ondersteund wordt door browsers. Je hebt dus een CSS processor nodig die dat omzet naar CSS3.

Concreet zou ik een custom selector --heading willen maken die één van de zes headers voorstelt. Zo is het makkelijk om een bepaalde stijl toe te passen op alle mogelijke headers.

@custom-selector :--heading h1, h2, h3, h4, h5, h6;

:--heading:hover {

color: red;

}

Je zal de webpack pipeline dus moeten uitbreiden zodat CSS bestanden met bovenstaande syntax omgezet worden naar een geldig CSS3 formaat. Hier zal je wel wat Google-speurwerk moeten voor verrichten :)

## Oefening4 – Bootstrap via CDN

Maak een ‘BootstrapViaCDN’ Visual Studio solution (ASP.NET Core 5.0 project, Empty template) of een NodeJS Express server aan.

Voorzie een index.html met de [html template](https://getbootstrap.com/docs/5.0/getting-started/introduction/) die door Bootstrap wordt aanbevolen.

Voeg wat inhoud toe aan index.html zodat je de styling kan zien.​

## Oefening 5 – Bootstrap via NPM en Webpack

Maak een ‘BootstrapViaNPM’ Visual Studio solution (ASP.NET Core 5.0 project, Empty template) of een NodeJS Express server aan.

Voeg een package.json bestand toe​

Voorzie een index.html met de [html template](https://getbootstrap.com/docs/4.3/getting-started/introduction/) die door Bootstrap wordt aangeraden, **maar haal de <link> en <script> tags weg**.​

Zie <https://getbootstrap.com/docs/5.0/getting-started/webpack/>

* Configureer je project voor webpack (cfr. Workshop)​
* Voeg een src\index.js bestand toe​
* Volg de procedure in Importing compiled CSS​

Voeg de nodige link en script tags toe in index.html​.

## Oefening 6 – Stylen van site met Bootstrap

In onderdeel 2 werd als oefening een URL checker gemaakt.

Voeg hier nu Bootstrap 5 en FontAwesome aan toe​

* Neem de documentatie door op <https://getbootstrap.com/docs/5.0/getting-started/introduction/> ​
* En eventueel ook op <https://www.w3schools.com/bootstrap5/index.php> ​
* “Extra”: kan je de NPM/webpack werkwijze gebruiken?

## Oefening 7 – Omzetten van ES6 naar ES5

Je hebt keihard gewerkt aan een webapplicatie, gebruik makende van de modernste technieken. Uiteraard heb je ook gebruik gemaakt van ES6…

Maar nu komt je manager/klant langs met de opmerking ‘Ohja, nog vergeten te zeggen … onze webapplicatie moet op oude terminals draaien met IE11. Dat gaat toch lukken he ?’

Je neemt je code snel door en komt het volgende[[1]](#footnote-2) tegen :

|  |
| --- |
| const helloWorld = "Hello World";  class Greeter {  constructor(userName) {  this.\_userName = userName;  }  greet() {  return `${helloWorld} and ${this.\_userName}!`;  }  }  const span = document.getElementById("greeting");  document.getElementById("name").addEventListener("input", (e) => {  let greeter = new Greeter(e.target.value);  span.innerText = greeter.greet();  }); |

Is dit geldige ES5 code? Hoe weet je wat er ondersteund wordt en wat niet?

Wat moet je doen om dit om te zetten naar ES5 zonder aan je broncode sleutelen?

1. Een aantal JS concepten zal je mogelijks nog niet kennen (zoals vb. class). We komen er nog op terug in de volgende onderdelen. [↑](#footnote-ref-2)