

02.03-external-css

In de twee voorgaande onderdelen hebben we bekeken hoe inline CSS en style block werkt en dat deze niet de beste oplossingen zijn voor de volgende redenen.

inline css cons:

- niet onderhoudsvriendelijk
- altijd hoogste prioriteit dus minder flexibel
- veel herhaling
- maakt opmaak (HTML) minder leesbaar

style block css cons:

- minder gebruiksvriendelijk dan external CSS
- HTML pagina specifiek

Voorheen zag de structuur van ons project er als volgt uit.

```
.
├── index.html
```

Eerst gaan we een extra folder maken *src* met daarin nog een folder genaamd *css* in deze folder plaatsen we al onze css bestanden die we dan gaan linken aan onze HTML pagina's.

```
.
├── src
│   ├── css
│   │   └── main.css
└── index.html
```

De structuur van ons project zou er dan als volgt moeten uitzien.

In onze css bestanden kunnen we direct onze css classes en identifiers schrijven zonder deze in een style block te moeten plaatsen.

Voorbeeld:

```
.c-red{
  color:red;
}
```

Link CSS to HTML page

we hebben nu een extern css bestand in de volgende plaats "src/css" genaamd "main.css".

om nu de link tussen onze HTML en CSS te maken gaan we gebruik maken van de tag **<link>** deze plaatsen we ook weer in de **<head>** van onze HTML pagina.

```
<!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">

  <!-- link css -->
  <link rel="stylesheet" href="src/css/main.css">

  <title>Demo external css</title>
</head>
<body>

</body>
</html>
```

rel: wordt gebruikt om de relatie tussen het huidige en het gekoppelde document op te geven.

href: dit staat voor "hyperlink reference" en het identificeert een plaats (een hyperlink of gewoonlijk een link genoemd) in een hypertext die de gebruiker naar een andere plaats linkt. Of een koppeling maakt met een ander bestand.

Link to external css

We hebben nu bekeken hoe we onze css in een apart bestand kunnen plaatsen, maar we kunnen ook naar css bestanden verwijzen die we niet in ons project zitten.

```
<!-- CSS only -->
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
gH2yIJqKdNHPEq0n4Mqa/HGKIhSkIHeL5AyhkYV8i59U5AR6csBvApHHNl/vI1Bx"
crossorigin="anonymous">
```

met deze link gebruiken we een css framework in dit geval gaat het over **bootstrap**. Om bootstrap te gebruiken kan je de source bestanden downloaden en toevoegen aan je project of via de **CDN** werken zoals het voorbeeld hierboven.

De code voor bootstrap te gebruiken met CDN kan je terugvinden op de website van [Bootstrap](#) met de CDN kunnen we ook de javascript van bootstrap gebruiken.

override thirdparty css classes

Wanneer we een css framework gebruiken maar we willen de stijl van de voor gedefinieerde klassen aanpassen dan zullen we in onze eigen css bestand dezelfde naam gebruiken zoals bij het framework, maar dan moeten we onze css bestand later inlezen dan het css bestand van het framework.

Voorbeeld:

```
<!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">

  <!-- link css -->
  <!-- framework css first -->
  <link rel="stylesheet" href="src/css/bootstrap.css">

  <!-- our main css -->
  <link rel="stylesheet" href="src/css/main.css">

  <title>Demo reset css</title>
</head>
<body>

</body>
</html>
```

Reset css

Omdat onze website met verschillende browsers kan bezocht worden kunnen we het probleem tegen komen dat onze website er verschillend kan uitzien, het gaat dan om kleine verschillen.

Door een reset css te gebruiken zal onze front-end niet onderhevig zijn aan de verschillen onder de browsers.

```
<!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">

  <!-- link css -->
  <!-- reset css first -->
  <link rel="stylesheet" href="src/css/reset.css">

  <!-- our main css -->
  <link rel="stylesheet" href="src/css/main.css">

  <title>Demo reset css</title>
</head>
<body>

</body>
</html>
```

Note: de reset css moet altijd als eerste ingeladen worden!