



Voor mijn Website Layout Poc heb ik eerst gekeken naar hoe een website opgebouwd is. De website hierboven bestaat uit veel onderdelen zoals een header met daarin verschillende divs. Sections met daarin ook divs en een footer. Ik heb door middel van html en css geëxperimenteerd en met deze content blokken gespeeld. Dit zie je in de code hieronder.

```

<body>

<h1>Let's build a layout!</h1>

<div>A</div> <!-- div staat voor divider, het is simpel gezegd een blokje met content -->
<br> <!-- Hier zie je dat de div het hele scherm van links naar rechts in beslag neemt aan de achtergrond kleur -->
<div>B</div>
<div>C</div>
<div>D</div>

<div class="kip">A</div> <!-- Door class="kip" geven we een classnaam genaamd kip mee aan het element. Deze kun je later specifiek targeten in de CSS -->
<div class="kip">B</div>
<div class="kip">C</div>
<div class="kip">D</div>

<header></header> <!-- Omdat de code makkelijker te overzien zijn er sementic tags -->
<nav></nav> <!-- header, navigation, section en footer zijn sementic tags -->
<section></section> <!-- Deze zijn precies hetzelfde als een div maar overzichtelijker omdat de naam zegt wat het betekent -->
<footer></footer>

<header>header</header>
<section>first-section</section>
<section>second-section</section>
<footer>footer</footer>
<p>Hieronder zie je een voorbeeld van een website layout en welke tags er gebruikt zijn</p>

 <!-- Met img roep je een afbeelding op -->
<!-- Zorg ervoor dat bij src de pad naar de afbeelding staat, als deze in een map staat moet dit erbij staan -->
<!-- Achter alt komt de tekst die weergegeven word als de afbeelding niet goed zou laden op de website -->

</body>
</html>

```

Dit is de html code voor mijn Website Layout Poc. Ik heb hierin gespeeld met divs, headers, sections en footers. Ook heb ik hier voor het eerst een afbeelding opgeroepen en met classnamen gewerkt.

Ik heb de uitleg gecoment bij elke regel code. Kort samengevat staat er wat een divider is en hoe deze werken, wat een classnaam doet en dat sementic tags eigenlijk niet veel anders zijn. Deze namen worden vooral gebruikt om het overzichtelijker te maken. Hieronder zie je de CSS code die ik heb gebruikt om de verschillende divs een andere kleur, positie en formaat te geven.

```

body {
  background: rgb(58, 58, 58);
  color: white;
  font-family: Helvetica Neue, Helvetica, Arial;
  margin: 0;
  padding: 0; /* Margin en padding zorgt er hier voor dat je geen ruimtes tussen je divs en de randen van je scherm hebt */
} /* Dalijk word uitgelegd hoe margin en padding precies werken */
div {
  background: blue; /* Hierdoor krijgen alle divs een blauwe achtergrond */
}

header {
  background: black;
  margin: 20px; /* Margin zorgt ervoor dat er ruimte omheen komt */
} /* Nu zie je dat er boven, onder, links en rechts van de header een ruimte van 20px is */
section {
  background: orange;
  padding: 20px; /* Padding zorgt ervoor dat die ruimte in de box komt ipv eromheen */
} /* Nu zie je dat de section 20px groter is aan de bovenkant, onderkant, links en rechts */
footer {
  background: black;
  margin: 5px 600px; /* Je kunt meerdere waardes invullen bij padding/margin. Bij 1 waarde word deze aan alle kanten gebruikt */
} /* Bij twee waardes staat de 1e waarde voor boven en onderen en de 2e voor links en rechts */
p {
  background: black;
  padding: 20px 30px 15px 300px; /* Hier zie je 4 waardes. Je kunt ipv px nog vele andere units gebruiken */
} /* Een voorbeeld hiervan is %, hiermee pak je bijv 20% van het scherm als ruimte*/
.kip {
  background: red;
  display: inline-block; /* Inline-block zorgt ervoor dat de divs niet onder elkaar maar naast elkaar komen te staan */
  width: 100px; /* Met width en height kun je een hoogte en breedte aan de divs geven */
  height: 40px; /* Je kunt ook display: flex; bij de section zetten als je gebruik wilt maken van flexbox */
}

```

Hierboven zie je de Css code voor mijn website layout page. Ik vertel hier hoe margin en padding werken, wat de background property doet en hoe je door middel van display de manier waarop je divs worden getoond kan veranderen. Hieronder zie je het resultaat van alle code hierboven.

## Let's build a layout!

