

```
<body>
  <header>
    <nav>      <!-- De nav blijft een sementic tag, dit is dus gewoon een div maar met een duidelijke naam -->
      <ul>      <!-- Voor een navigatie balk gebruiken we een unordered list met daarin de pagina's die we willen -->
        <li>Home</li>
        <li>Locations</li>  <!-- Door ul>li*3 maak je een <ul> aan met daarin 3 * <li> -->
        <li>Contact</li>
      </ul>
    </nav>
    <h1>Title</h1>
  </header>

  <section class="een">
    <div>a</div>
    <div>b</div>
    <div>c</div>
  </section>
  <section class="twee">
    <div>a</div>
    <div>b</div>
    <div>c</div>
  </section>
  <section class="drie">
    <div>a</div>
    <div>b</div>
    <div>c</div>
  </section>

  </section>
  <section>second-section</section>
  <footer>footer</footer>

  <div>
    <div class="center">1</div>
    <div class="end">2</div>
    <div>3</div>
  </div>

```

Dit is het html gedeelte van mijn Flexbox Poc. Bovenin vertel ik wat een sementic tag. Ook kun je makkelijk grotere stukken code typen door `ul>li*3` en dan enter te typen. Dit betekent een unordered list met daarin 3 list items. Dit werkt zo voor meerdere tags en kan code schrijven een stuk sneller maken. Ik heb verschillende class namen gebruikt voor de sections en divs zodat ik per classnaam een andere flexbox regel kan toepassen.

Verder heb ik een paar sections met divs gemaakt zodat ik content heb om flexbox op te oefenen. Ik heb overal comments bij gezet die de code nog eens extra verduidelijken.

Scroll verder voor het css gedeelte.

```

body {
  background: □rgb(58, 58, 58);
  color: ■white;
  font-family: Helvetica Neue, Helvetica, Arial;
  margin: 0;
  padding: 0;
}
header {
  background: □black;
  padding: 20px;
}
section {
  background: ■white;
  color: ■white;
  padding: 20px;
}
div {
  background: □red;
  width: 10vw;
}
ul {
  margin: 0;
  padding: 0; /* Hiermee halen we de standaard margin/padding weg omdat dat er nu iets netter uit ziet */
  list-style-type: none; /* Hiermee halen we de bolletjes voor de list items weg */
}
li {
  display: inline-block; /* Hiermee zetten we de list items achter elkaar */
  margin: 0 20px 0 0; /* Hiermee zetten we wat margin rechts van de list items, margin-right: 20px; werkt ook */
}
footer {
  background: □black;
  padding: 10px 20px;
}

```

Dit zijn wat basic Css regels waarmee ik de opzet van mijn Poc gestyled hebt. Zo heb ik wat achtergrond kleuren toegevoegd zodat je goed kan zien wat de divs en de inhoud van de divs doen over het scherm. Hieronder zie je de flexbox code waarmee ik gespeeld heb.

```

section.een{
  display: flex; /* Display flex moet op de parent container van de divs en maakt er standaard row van */
  flex-direction: row; /* Hier kun je de richting van de divs veranderen. */
  /* row voor links naar rechts, row-reverse voor rechts naar links */
  /* column voor boven naar onderen, column reverse voor onderen naar boven*/
}
.twee{
  display: flex;
  flex-direction: column;
}
.drie{
  display: flex;
  flex-direction: row-reverse;
}
.drie div{
  margin: auto; /* Hiermee zorg je ervoor dat je divs gelijk over het scherm worden verdeeld, zelfs als je je window verkleint */
}
.center{
  display: flex;
  justify-content: center; /*Met justify-content kun je de content van een container veranderen, Door center word alles in het midden gezet*/
}
.end{
  display: flex;
  justify-content: flex-end; /* Door flex-end word alles aan het einde gezet. Door space-between word de ruimte tussen de content blokken even groot*/
}

```

Je ziet dat je overal eerst display: flex; moet neerzetten om met flexbox te kunnen werken. Door display: flex; word de inhoud automatisch al in een row geplaatst in plaats van een column.

Door een flex-direction te geven kan je kiezen of de inhoud in een row, reversed row, column of reversed column komt te staan. In het resultaat hieronder zie je het verschil.

Ook kun je justify-content gebruiken om de inhoud te positioneren. Door bijvoorbeeld Justify-content: center; te gebruiken word alle content in het midden geplaatst.

Title

a b c

a
b
c

c

b

a

footer

1
2
3

Je ziet dat ik door middel van flexbox de divs in een rij en column kan zetten en dat ik de positie van de divs én de inhoud kan positioneren.