

1. CSS3: TRANSITIONS EN ANIMATIONS

Om transitions en animations te gebruiken in CSS3 raden we aan om een preprocessor (vb. prepros) te gebruiken zodat alle geschreven code kan werken in iedere browser. De transition property werkt niet in IE9 en vorige versies! Om dezelfde functionaliteiten te bouwen heb je voor deze oudere browsers nog javascript/jquery nodig!

1.1. TRANSITIONS

1.1.1. 1 PROPERTY

TRANSITION 01: transition-property

De transition-property activeert een css property en voegt er een transition effect aan toe.

Een transition effect wordt meestal gebruikt in combinatie met een hover.

De transition duration kan zowel in ms (milliseconden) als s (seconden) worden gebruikt.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Transitions 01</title>
  <style>
    body{
      background-color: thistle;
    }
    button{
      display:block;
      width:60%;
      padding:10px;
      margin: 50px auto;
      border: none;
      cursor: pointer;
      color:#fff;
      background-color:seagreen;
      /*met transition*/
      transition-property: background-color;
      transition-duration: 400ms;
    }
    button:hover{
      background-color: aquamarine;
    }
  </style>
</head>
<body>
  <button>Transition 01</button>
</body>
</html>
```



TRANSITION 02: transition-delay

Zoals we in het eerste voorbeeld gezien hebben kunnen we gelijk welke property van CSS3 aan de transition-property koppelen.

In onderstaand voorbeeld voeren we dit uit met **width**.

Daarnaast voegen we een nieuwe property toe, nl. **transition-delay**. Transition-delay zorgt ervoor dat er een **vertraging** optreedt vooraleer de transition start.

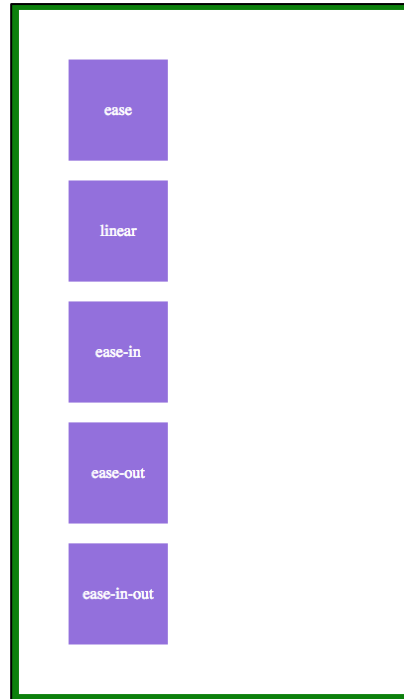
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Transitions 02</title>
  <style>
    body{
      background-color: thistle;
    }
    div{
      margin-top: 50px;
      background-color: gray;
      height: 300px;
      width: 50%;
      transition-property: width;
      transition-duration: .5s;
      transition-delay: 2s;
    }
    div:hover{
      width: 90%;
    }
  </style>
</head>
<body>
  <div>
    </div>
</body>
</html>
```



TRANSITION 03: transition-timing-function

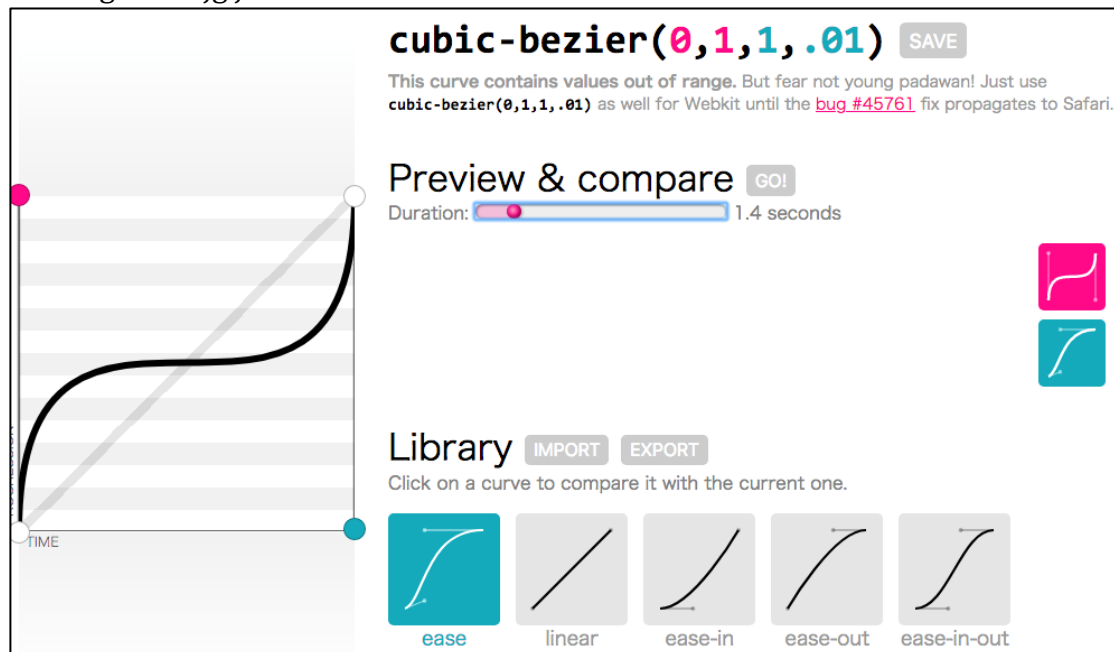
Naast de transition-duration hebben we nog een bijkomende mogelijkheid die geen duration behoeft, nl. de **transition-timing-function**. De opties die we hier gebruiken zijn ease, linear, ease-in, ease-out en ease-in-out;

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: green;
    }
    .container{
      margin: 10px;
      padding: 30px;
      background-color: #fff;
    }
    .box{
      padding: 10px;
      background-color: mediumpurple;
      color: #fff;
      width: 80px;
      margin: 20px;
      position: relative;
      left: 0;
      transition-property: left;
      transition-duration: 1s;
    }
    .box p{
      text-align: center;
      line-height: 50px;
    }
    .container:hover .box{
      left: 80%;
    }
    #ease{
      transition-timing-function: ease;
    }
    #linear{
      transition-timing-function: linear;
    }
    #ease-in{
      transition-timing-function: ease-in;
    }
    #ease-out{
      transition-timing-function: ease-out;
    }
    #ease-in-out{
      transition-timing-function: ease-in-out;
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="box" id="ease">
      <p>ease</p>
    </div>
    <div class="box" id="linear">
      <p>linear</p>
    </div>
    <div class="box" id="ease-in">
      <p>ease-in</p>
    </div>
    <div class="box" id="ease-out">
      <p>ease-out</p>
    </div>
    <div class="box" id="ease-in-out">
      <p>ease-in-out</p>
    </div>
  </div>
</body>
</html>
```



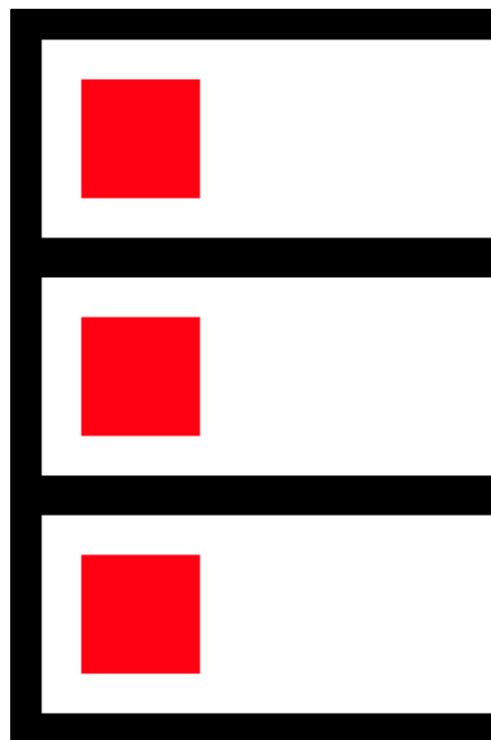
TRANSITION 04: transition-timing-function: cubic-bezier

Op de site van cubic-bezier.com zie je duidelijk wat de verschillen zijn wanneer we de property cubic-bezier gebruiken. Alle properties gaan tot **1**. Wanneer je over 1 gaat krijg je een bounce back effect.



We passen bovenstaande toe in een voorbeeld.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: black;
    }
    .container{
      background-color: #fff;
      margin: 20px;
      padding: 20px;
    }
    .box{
      background-color: red;
      height: 60px;
      width: 60px;
      position: relative;
      left: 0;
      transition-property: left;
      transition-duration: 1s;
    }
    .container:hover .box{
      left: 90%;
    }
    #one .box{
      transition-timing-function: cubic-bezier(.03,.9,.97,-0.08);
    }
    #two .box{
      transition-timing-function: cubic-bezier(.69,.03,.97,-0.08);
    }
    #three .box{
      transition-timing-function: cubic-bezier(.8,-0.43,.6,1.55);
    }
  </style>
</head>
<body>
  <div class="container" id="one">
    <div class="box">
    </div>
  </div>
  <div class="container" id="two">
    <div class="box">
    </div>
  </div>
  <div class="container" id="three">
    <div class="box">
    </div>
  </div>
</body>
</html>
```



1.1.2. MEERDERE PROPERTIES

TRANSITION 05: transition-property

Het is mogelijk om meerdere properties toe te voegen aan transition-property. Deze kunnen dan met dezelfde andere settings zoals transition-duration, ... werken.

Je zal zien wanneer je over de onderstaande button zal hoveren, dat zowel de background-color als de border-color de volgorde van de ingegeven transition-duration en transition-delay zullen volgen.

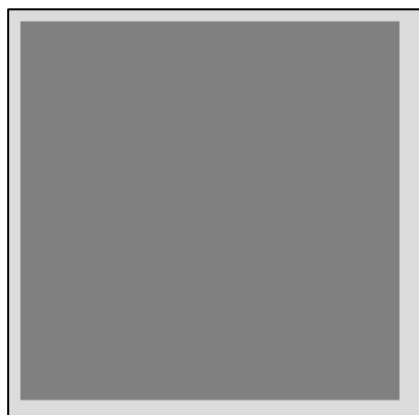
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: gray;
    }
    button{
      padding: 20px;
      display: block;
      width: 50%;
      cursor: pointer;
      color: #000;
      text-align: center;
      margin: 20px auto;
      border: 7px solid yellow;
      background-color: darkturquoise;
      transition-property: background-color,
        border-color;
      transition-duration: 1s, 2s;
      transition-delay: 0s, 1s;
    }
    button:hover{
      background-color: white;
      border-color: green;
    }
  </style>
</head>
<body>
  <button>Meerdere properties</button>
</body>
</html>
```



TRANSITION 06: transition-property

Hier is het de bedoeling dat de box een hoogte van 400px en een breedte van 500px krijgt rekening houdend met de transition-timing-fuction.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: gainsboro;
    }
    .box{
      height: 200px;
      width: 200px;
      background-color: gray;
      transition-property: height, width;
      transition-duration: 1s;
      transition-timing-function: ease-out,
        cubic-bezier(1,-0.29,.25,0.77);
    }
    .box:hover{
      height: 400px;
      width: 600px;
    }
  </style>
</head>
<body>
  <div class="box">
  </div>
</body>
</html>
```



1.1.3. SHORTHAND NOTATIE

TRANSITION 07: transition shorthand notatie

Kopieer de vorige oefening en pas als volgt aan:

Je kan in deze volgorde de volledige transition schrijven = **transition-property**, **transition-duration**, **transition delay** en **transition-timing-function** in één lijn als shorthand notatie schrijven, gedeclareerd per property.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: gainsboro;
    }
    .box{
      height: 200px;
      width: 200px;
      background-color: gray;
      /* transition-property: height, width;
      transition-duration: 1s,3s;
      transition-timing-function: ease-out,
      cubic-bezier(1,-0.29,.25,0.77);*/

      transition: height 1s ease-out, width 3s cubic-bezier(1,-0.29,.25,0.77);
    }
    .box:hover{
      height: 400px;
      width: 600px;
    }
  </style>
</head>
<body>

<div class="box">

</div>

</body>
</html>
```

Je zou in principe nog een delay kunnen toevoegen, bijv.:
transition-property, transition-duration, **transition-delay**, transition-timing-function
height 1s **3s** ease-out

OPMERKING:

wanneer de transition-duration en de transition-delay voor ALLE properties gelijk zijn dan kunnen we het ALL keyword toepassen

transition: **all** 1s 3s ease-out

TRANSITION 08: SELECTOREN - LINEAIR GRADIENT

De linear-gradient werkt met 3 parameters, nl. de position naar waar de transition zal vorderen, beginkleur en eindkleur.

Deze werkt samen met de background-position property, die voor de startpositie zorgt van de transition. De hover zal uiteindelijk de eindpositie bepalen.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>

  <style>
    *{
      margin: 0;
      padding: 0;
    }
    .box{
      width: 100%;
      height: 500px;
      background: white; /* fallback oude browsers */
      background: -webkit-linear-gradient(to left, white, red); /* Chrome 10-25, Safari 5.1-6 */
      background: linear-gradient(to left, white, red); /* W3C, IE 10+/ Edge, Firefox 16+, Chrome 26+, Opera 12+, Safari 7+ */
      background-size: 200%;
      background-position: top left;
      margin: 0 auto;
      transition: background-position 1s;
    }
    .box:hover{
      background-position: top right;
    }
    .box:hover h1{
      color: red;
    }
    .box h1{
      line-height: 500px;
      text-align: center;
      font-size: 5em;
      color: white;
      text-shadow: 0px 2px 5px rgba(0,0,0,0.1);
      transition: color 1s;
    }
  </style>
</head>
<body>

  <div class="box">
    <h1>TEKST</h1>
  </div>

</body>
</html>
```

TRANSITION 09: SELECTOREN - background-position

Kopieer het css bestand en plak deze in het html bestand.

```
<body>
  <div class="container">
    <div class="text">
      <h2>Bezoek</h2>
      <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Cupiditate minima accusantium laborum h</p>
    </div>

    <div class="cards-container">
      <h3>Cards</h3>

      <button>
        Hover om de cards te zien
      </button>

      <div class="card">
        <div class="card-img">
          
        </div>
        <div class="card-desc">
          <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Rem maiores aliquam ipsa eos.</p>
        </div>
      </div>
      <div class="card">
        <div class="card-img">
          
        </div>
        <div class="card-desc">
          <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Rem maiores aliquam ipsa eos.</p>
        </div>
      </div>
    </div>
  </div>
</body>
</html>
```

TRANSITION 09: SELECTOREN - OEFENINGEN

Maak 10 buttons in de body tag van je html.

Startpositie



2. CSS3: TRANSFORMS EN ANIMATIONS

2.1. TRANSFORMS

2.1.1. ROTATE

transform: rotate(20deg) wordt gebruikt om html tags te laten draaien d.m.v. een aantal graden in te geven.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: orange;
    }
    .box{
      height: 300px;
      width: 70%;
      margin: 50px auto;
      background-image: url('randomimage.jpg');
      background-size: cover;
      background-repeat: no-repeat;
      border: 5px solid #fff;
      transition: transform 1s ease-out;
    }
    .box:hover{
      transform: rotate(20deg);
    }
  </style>
</head>
<body>
  <div class="box">
  </div>
</body>
</html>
```


2.1.2. SCALE

transform02:Scale wordt gebruikt om een image uit of in te zoomen

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: orange;
    }
    .box{
      height: 300px;
      width: 70%;
      margin: 50px auto;
      background-image: url('randomimage.jpg');
      background-size: cover;
      background-repeat: no-repeat;
      border: 5px solid #fff;
      transition: transform 1s ease-out;
    }
    .box:hover{
      transform: scale(1.5);
    }
  </style>
</head>
<body>
  <div class="box">
  </div>
</body>
</html>
```

2.1.3. SKEW

transform03:Skew zorgt voor een vervorming uitgedrukt als parameter in graden van een html tag.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: orange;
    }
    .box{
      height: 300px;
      width: 70%;
      margin: 50px auto;
      background-image: url('randomimage.jpg');
      background-size: cover;
      background-repeat: no-repeat;
      border: 5px solid #fff;
      transition: transform 1s ease-out;
    }
    .box:hover{
      transform: skew(10deg);
    }
  </style>
</head>
<body>
  <div class="box">
  </div>
</body>
</html>
```

2.1.4. TRANSLATE

transform04: translate zorgt voor een verplaatsing van een html tag binnen de pagina

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: orange;
    }
    .box{
      height: 300px;
      width: 70%;
      margin: 50px auto;
      background-image: url('randomimage.jpg');
      background-size: cover;
      background-repeat: no-repeat;
      border: 5px solid #fff;
      transition: transform 1s ease-out;
    }
    .box:hover{
      transform: translate(50px);
    }
  </style>
</head>
<body>
  <div class="box">

  </div>
</body>
</html>
```

transform05:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: orange;
    }
    .box{
      height: 300px;
      width: 70%;
      margin: 50px auto;
      background-image: url('random.jpg');
      background-size: cover;
      background-repeat: no-repeat;
      border: 5px solid #fff;
      transition: transform 5s ease-out;
      transform-origin: 150% 50%;
    }
    .box:hover{
      transform: rotate(50deg);
    }
  </style>
</head>
<body>
  <div class="box">

  </div>
</body>
</html>
```

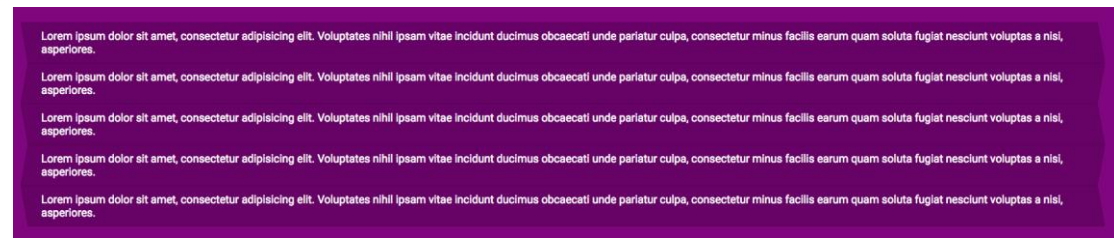
2.1.5. MIX

Zoals we reeds weten kunnen we bij transitions meerdere properties aanspreken in 1 keer. Bij een transform kunnen we dit namelijk ook! transform06:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: orange;
    }
    .box{
      height: 300px;
      width: 70%;
      margin: 50px auto;
      background-image: url('randomimage.jpg');
      background-size: cover;
      background-repeat: no-repeat;
      border: 5px solid #fff;
      transition: transform 0.3s cubic-bezier(.15,-0.61,.24,1.73);
    }
    .box:hover{
      transform: rotate(20deg) scale(1.5);
    }
  </style>
</head>
<body>
  <div class="box">
  </div>
</body>
</html>
```

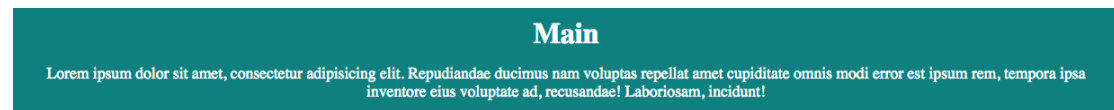
Oefening: skew

Zorg ervoor dat je de randen skewed van onderstaande paragrafen



Oefening: sliding overlap

Zorg ervoor dat er vanaf de linkerkant een overlap komt van een tweede kader.



2.2. 3D TRANSFORMS

2.2.1. 3D transform:perspective property - rotateX, rotateY, rotateZ

Transform3D01.html: de perspectieve property zorgt ervoor dat je een effect krijgt waarin alles achterwaarts wordt gedrukt in dit voorbeeld in combinatie met rotateX. (x-as). rotateY drukt naar binnen toe. Probeer maart uit. Vergroot en verklein het aantal pixels in de perspective property en ontdek de verschillen.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: purple;
    }
    .container{
      perspective: 800px;
    }
    .box{
      height: 300px;
      width: 70%;
      margin: 50px auto;
      background-image: url('randomimage.jpg');
      background-size: cover;
      background-repeat: no-repeat;
      border: 5px solid #fff;
      transition: transform 0.3s;
    }
    .box:hover{
      transform: rotateX(30deg);
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="box">
    </div>
  </div>
</body>
</html>
```

2.2.2. 3D transform:perspective property - rotate3d

Transform3D04.html: rotate3d(x,y,z,a)

het x-coördinaat bepaalt de rotatie op de x-as (waarden tussen 0 en 1)

het y-coördinaat bepaalt de rotatie op de y-as (waarden tussen 0 en 1)

het z-coördinaat bepaalt de rotatie op de z-as (waarden tussen 0 en 1)

het a-coördinaat zal de rotatiehoek bepalen. Een positieve hoek draait met de klok mee, een negatieve hoek draait tegen de wijzers van de klok in.

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: purple;
    }
    .container{
      perspective: 800px;
    }
    .box{
      height: 300px;
      width: 70%;
      margin: 50px auto;
      background-image: url('randomimage.jpg');
      background-size: cover;
      background-repeat: no-repeat;
      border: 5px solid #fff;
      transition: transform 0.3s;
    }
    .box:hover{
      transform: rotate3d(1,1,1,30deg);
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="box">

    </div>
  </div>
</body>
</html>

```

2.2.3. 3D transform:perspective property - backface-visibility

Transform3D03.html Wanneer we 2 kanten hebben aan een html tag, nl. een voor- en een achterzijde dan kunnen we ervoor kiezen om de achterkant al dan niet te tonen aan de gebruiker. In dit voorbeeld doen we dit niet en zetten we de backface-visibility op hidden.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: gainsboro;
    }
    .container{
      perspective: 800px;
    }
    .box{
      position: relative;
      height: 200px;
      width: 50%;
      margin: 80px auto;
      border: 5px solid #fff;
      transition: transform 0.6s;
      transform-style: preserve-3d;
    }
    .box:hover{
      transform: rotateX(180deg);
    }
    .front{
      position: absolute;
      height: 200px;
      width: 100%;
      background-color: rebeccapurple;
      backface-visibility: hidden;
      transform: rotateX(180deg);
    }
    .back{
      position: absolute;
      height: 200px;
      width: 100%;
      background-color: teal;
      backface-visibility: hidden;
      transform: rotateX(0deg);
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="box">
      <div class="front">
      </div>
      <div class="back">
      </div>
    </div>
  </div>
</body>
</html>
```

2.2.4. 3D transform:translate3d

Transform3D04.html: translate3d(tx,ty,tz)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: gainsboro;
    }
    .container{
      perspective: 800px;
    }
    .box{
      height: 300px;
      width: 70%;
      margin: 50px auto;
      background-image: url('1.jpg');
      background-size: cover;
      background-repeat: no-repeat;
      border: 5px solid #fff;
      transition: transform 0.3s;
    }
    .box:hover{
      transform: translate3d(20px,50px, 100px);
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="box">
    </div>
  </div>
</body>
</html>
```

2.2.5. 3D transform:scale3d

Transform3D05.html: scale3d(sx,sy,sz)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    body{
      background-color: gainsboro;
    }
    .container{
      perspective: 800px;
    }
    .box{
      height: 300px;
      width: 70%;
      margin: 50px auto;
      background-image: url('1.jpg');
      background-size: cover;
      background-repeat: no-repeat;
      border: 5px solid #fff;
      transition: transform 0.3s;
    }
    .box:hover{
      transform: scale3d(1.2, 0.5, 1);
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="box">
    </div>
  </div>
</body>
</html>
```


3. CSS3: ANIMATIONS

3.1. CSS3: ANIMATIONS EN KEYFRAMES

Animations en keyframes

Animations01.html

Een animatie laat je gradueel van één stijl naar een andere stijl overgaan.

Je kan onbeperkt css properties wijzigen.

Om css animation te gebruiken dien je keyframes te gebruiken.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    *{
      box-sizing: border-box;
    }
    body{
      background-color: gainsboro;
    }
    .box{
      position: relative;
      height: 200px;
      width: 50%;
      margin: 80px auto;
      border: 5px dashed #fff;
      animation-name: anim;
      animation-duration: 2s;
    }
    @keyframes anim{
      0%{
        background-color: pink;
      }
      50%{
        background-color: chocolate;
      }
      75%{
        background-color: crimson;
      }
      100%{
        background-color: tomato;
      }
    }
  </style>
</head>
<body>
  <div class="box">

  </div>
</body>
</html>
```

3.2. CSS3: ANIMATIONS - animation-delay

Animations02.html: ook bij animations kan je een delay (vertraging) in de animatie toepassen d.m.v. de animation-delay property. Animation-iteration-count mogelijkheden zijn: aantal, infinite(oneindig), initial

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    *{
      box-sizing: border-box;
    }
    body{
      background-color: gainsboro;
    }
    .box{
      position: relative;
      height: 200px;
      width: 50%;
      margin: 80px auto;
      border: 5px dashed #fff;
      animation-name: anim;
      animation-duration: 2s;
      animation-delay: 0.3s;
      animation-iteration-count: infinite;
    }
    @keyframes anim{
      0%{
        background-color: pink;
      }
      50%{
        background-color: chocolate;
      }
      75%{
        background-color: crimson;
      }
      100%{
        background-color: tomato;
      }
    }
  </style>
</head>
<body>
  <div class="box">
  </div>
</body>
</html>
```

3.3. CSS3: ANIMATIONS - animation-direction/animation-timing

Animations03.html: met animation-direction kan je de animatie beweging bepalen. Animation-direction mogelijkheden zijn: reverse, alternate, alternate-reverse. Animation timing mogelijkheden zijn: linear, ease, ease-in, ease-out, ease-in-out, step-start, step-end, cubic bezier

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    *{
      box-sizing: border-box;
    }
    body{
      background-color: gainsboro;
    }
    .box{
      position: relative;
      height: 200px;
      width: 50%;
      background-color: teal;
      margin: 80px auto;
      border: 5px solid #fff;
      animation-name: anim;
      animation-duration: 2s;
      animation-iteration-count: infinite;
      animation-direction: alternate-reverse;
      animation-timing-function: linear;
    }
    @keyframes anim{
      0%{
        transform: translateX(0px);
      }
      100%{
        transform: translateX(150px);
      }
    }
  </style>
</head>
<body>
  <div class="box">

  </div>
</body>
</html>
```

3.4. CSS3: ANIMATIONS - animation-play state

Animations04: animation play state mogelijkheden zijn: paused of running

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    *{
      box-sizing: border-box;
    }
    body{
      background-color: gainsboro;
    }
    .box{
      position: relative;
      height: 200px;
      width: 50%;
      background-color: teal;
      margin: 80px auto;
      border: 5px solid #fff;
      animation-name: anim;
      animation-duration: 2s;
      animation-iteration-count: infinite;
      animation-direction: alternate-reverse;
      animation-timing-function: linear;
      animation-play-state: paused;
    }
    .box:hover{
      animation-play-state: running;
    }
    @keyframes anim{
      0%{
        transform: translateX(0px);
      }
      100%{
        transform: translateX(150px);
      }
    }
  </style>
</head>
<body>
  <div class="box">

  </div>
</body>
</html>
```

3.5. CSS3: ANIMATIONS - animation-fill-mode

Animations05: animation fill-mode mogelijkheden zijn: none, forwards, backwards, both, initial

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    *{
      box-sizing: border-box;
    }
    body{
      background-color: gainsboro;
    }
    .box{
      position: relative;
      height: 200px;
      width: 50%;
      background-color: teal;
      margin: 80px auto;
      border: 5px solid #fff;
      animation-name: anim;
      animation-duration: 2s;
      animation-fill-mode: forwards;
    }
    .box:hover{
    }
    @keyframes anim{
      0%{
        transform: translateX(-150px);
      }
      100%{
        transform: translateX(150px);
      }
    }
  </style>
</head>
<body>
  <div class="box">

  </div>
</body>
</html>
```

3.6. CSS3: ANIMATIONS - animation-name

Animations06:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    *{
      box-sizing: border-box;
    }
    body{
      background-color: gainsboro;
    }
    .box{
      position: relative;
      height: 200px;
      width: 50%;
      margin: 50px auto;
      border: 5px solid #fff;
    }
    @keyframes bg{
      0%{
        background-color: gray;
      }
      100%{
        background-color: tomato;
      }
    }
    @keyframes move{
      0%{
        transform: translateX(0px);
      }
      100%{
        transform: translateX(150px);
      }
    }
    #bg{
      animation-name: bg;
      animation-duration: 2s;
    }
    #move{
      animation-name: move;
      animation-duration: 2s;
    }
    #both{
      animation-name: move, bg;
      animation-duration: 2s;
    }
  </style>
</head>
<body>
  <div class="box" id="bg">

  </div>
  <div class="box" id="move">

  </div>
  <div class="box" id="both">

  </div>
</body>
</html>
```

3.7. CSS3: ANIMATIONS - shorthand notatie

Animations07:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    *{
      box-sizing: border-box;
    }
    body{
      background-color: gainsboro;
    }
    .box{
      position: relative;
      height: 200px;
      width: 50%;
      background-color: teal;
      margin: 80px auto;
      border: 5px solid #fff;
      animation: anim 2s infinite alternate;
    }
    .box:hover{
    }
    @keyframes anim{
      0%{
        transform: translateX(-150px);
      }
      100%{
        transform: translateX(150px);
      }
    }
  </style>
</head>
<body>
  <div class="box">

  </div>
</body>
</html>
```