

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/3.12.4/gsap.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/3.12.4/ScrollTrigger.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/3.12.4/TextPlugin.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/gsap@3.12.5/dist/MotionPathPlugin.min.js"></script>
<script src="mijnJavaScript.js"></script>
```

## Voorbeelden met duration, opacity, rotation, scale, repeat, delay

Beweeg **CLASS tekst** van zijn originele plaats naar rechts (1000 pixels) in een tijd van 3 seconden

```
gsap.to(".tekst", {x:1000, duration:3});
```

Beweeg **ID koe** van zijn originele plaats naar rechts-onder in een tijd van 6 seconden, naar een doorzichtigheid van 50%

```
gsap.to("#koe", {x:500, y:300, duration:6, opacity: 0.5});
```

Beweeg **CLASS zebra** van zijn originele plaats naar rechts-boven in een tijd van 3 seconden, naar een doorzichtigheid van 20%

```
gsap.to(".zebra", {x:300, y:-200, duration:3, opacity: 0.2});
```

Beweeg **CLASS olifant** van zijn originele plaats naar links in een tijd van 7 seconden en roteer 90 graden tegen de klok

```
gsap.to(".olifant", {x:-20, duration:7, rotation: -90});
```

Beweeg **CLASS beer** van zijn originele plaats naar rechts in een tijd van 7 seconden en zet afbeelding ondersteboven (180 graden)

```
gsap.to(".beer", {x:800, duration:7, rotation: 180});
```

Beweeg **CLASS kameel** NAAR zijn originele plaats, vertrekkende van rechts-onder, in een tijd van 11 seconden

```
gsap.from(".kameel", {x:800, y:550, duration:11});
```

Beweeg **ID koe** van zijn originele plaats naar boven, 8 seconden na laden pagina, in een tijd van 2 seconden

```
gsap.to("#koe", {y:-100, duration:2, delay: 8});
```

Beweeg **ID koe** van zijn originele plaats naar boven, in een tijd van 2 seconden, maak 2 maal zo groot

```
gsap.to("#koe", {y:-100, duration:2, scale: 2});
```

Beweeg **ID koe** van zijn originele plaats naar boven, in een tijd van 2 seconden, maak 3 maal zo klein

```
gsap.to("#koe", {y:-100, duration:2, scale: 0.33});
```

Beweeg **CLASS zebra** van zijn originele plaats naar rechts, in een tijd van 4 seconden en herhaal dit nog 1 extra keer

```
gsap.to(".zebra", {x:500, duration:4, repeat: 1});
```

Beweeg **CLASS olifant** van zijn originele plaats naar rechts, in een tijd van 2 seconden, herhaal dit nog 4 extra keer, heen en terug

```
gsap.to(".olifant", {x:500, duration:2, repeat: 4, yoyo:true});
```

Beweeg **CLASS beer** van zijn originele plaats naar rechts, in een tijd van 2 seconden, en herhaal dit voor altijd

```
gsap.to(".beer", {x:800, duration:2, repeat: -1});
```

Beweeg **CLASS beer** van zijn originele plaats naar rechts, in een tijd van 2 seconden, en herhaal dit voor altijd, heen en terug

```
gsap.to(".beer", {x:800, duration:2, repeat: -1, yoyo:true });
```

Roteer **CLASS beer** 180 graden om zijn eigen as (Y-as) in een tijd van 7 seconden

```
gsap.to(".beer", {duration:7, rotationY: 180});
```

**NB:** rotationX kan ook

## Defaults van de animaties instellen

```
gsap.defaults({y: 200, duration: 0.8, opacity: 0});
```

## Voorbeelden met ease

```
gsap.to(".olifant1", {ease: "none"      });
gsap.to(".koel",    {ease: "bounce"    });
gsap.to(".beer1",   {ease: "elastic(0.4)"  });
gsap.to(".zebra1",  {ease: "elastic(0.8)"  });
gsap.to(".schaap1", {ease: "power1"      });

gsap.to(".olifant2", {ease: "power2"      });
gsap.to(".koe2",    {ease: "power3"      });
gsap.to(".beer2",   {ease: "power4"      });
gsap.to(".zebra2",  {ease: "back(0.8)"    });
gsap.to(".schaap2", {ease: "back(2)"     });
```

➔ combinaties ease met **.in .Out .inOut**      voorbeeld: **ease: "power1.inOut"**

## Voorbeelden met stagger

```
gsap.from(".afbeeldingen img", {x: 500, stagger: {each: 0.15, from: "center" } });
gsap.from(".afbeeldingen img", {x: 500, stagger: {each: 0.15, from: "end" } });
gsap.from(".afbeeldingen img", {x: 500, stagger: {each: 0.15, from: "begin" } });
gsap.from(".afbeeldingen img", {x: 500, stagger: {amount: 2, from: "center" } });
gsap.from(".afbeeldingen img", {x: 500, stagger: {amount: 2, from: "end" } });
gsap.from(".afbeeldingen img", {x: 500, stagger: {amount: 2, from: "begin" } });
```

➔ dit gaat met x, y, scale, rotate, ...

## Voorbeelden met vw (horizontaal) en vh (verticaal) voorbeelden

```
gsap.to(".olifant1", {x: "20vw"});
gsap.to(".koel", {x: "50vw"});
gsap.to(".beer1", {x: "90vw"});

gsap.to(".zebra1", {y: "10vh"});
gsap.to(".schaap1", {y: "50vh"});
```

## Voorbeelden met transformOrigin

```
gsap.to(".olifant2", {rotation: 180, transformOrigin: "left bottom"});
gsap.to(".olifant2", {rotation: 180, transformOrigin: "90% 90%"});
gsap.to(".olifant2", {scale: 1.5, transformOrigin: "200px 50px"});
```

➔ andere mogelijkheden te combineren met: left center right top center bottom

```
gsap.to(".olifant2", {rotation: 180, transformOrigin: "left bottom"});
```

**andere notatie:**

```
gsap.to(".olifant2",
{
    rotation: 180,
    transformOrigin: "left bottom"
});
```

## Werken met een tijdlijn (timeline) zonder delay te gebruiken

```
var animatie = gsap.timeline();
```

start 2 seconden nadat de vorige animatie gedaan is  
`animatie.to(".koe", {y:300, duration: 5}, "+=2")`



start 1 second voordat de vorige animatie gedaan is  
`animatie.to(".beer", {x:300, duration: 5}, "-=1")`



start tesamen met de vorige animatie  
`animatie.to(".olifant", {rotation:90, duration: 5}, "<")`



start 1 seconde na de start van de vorige animatie  
`animatie.to(".zebra", {opacity:0.5, duration: 5}, "<1")`



start exact bij seconde 8  
`animatie.to(".schaap", {x:200, duration: 5}, 8)`

## Responsive design

```
var animatie1 = gsap.timeline();
var animatie2 = gsap.timeline();

var scherm = gsap.matchMedia();

scherm.add("(min-width: 990px)", () => {
    var animatie1 = gsap.timeline();
    zet hier je code
});

scherm.add("(max-width: 989px)", () => {
    var animatie2 = gsap.timeline();
    zet hier je code
});
```

## ScrollTrigger (werkt met scrolling, dus geen timeline gebruiken)

**Steeds toevoegen:** `gsap.registerPlugin(ScrollTrigger);`

**Met een duration**

Laat **CLASS koe** van zijn originele plaats 1 volledige omwenteling maken, in een tijd van 3 seconden  
Dit start als de bovenkant van **CLASS koe** in de helft van het beeldscherm komt.

```
gsap.to(".koe", {duration: 3, rotation:360,
    scrollTrigger: {trigger: ".koe", start: "0% 50%", markers: true} });
```

Laat **CLASS zebra** NAAR zijn originele plaats gaan door 1 volledige omwenteling te maken, in een tijd van 5 seconden  
Dit start als de onderkant van **CLASS beer** op 1/3<sup>de</sup> van het beeldscherm komt.

```
gsap.from(".zebra", {duration: 5, rotation:360,
    scrollTrigger: {trigger: ".beer", start: "100% 33%", markers: false} });
```

**Zonder een duration (snelheid wordt bepaald door snelheid van scrolling)**

Laat **CLASS zebra** NAAR zijn originele plaats gaan door 1 volledige omwenteling te maken  
Dit start als de bovenkant van **CLASS beer** op 80% van het beeldscherm komt en stopt als de bovenkant van **CLASS beer** op 1/3<sup>de</sup> van het beeldscherm komt. De vertraging is "2".

```
gsap.from(".zebra", {rotation:360,
    scrollTrigger: {trigger: ".beer", start: "0% 80%",
        end: "0% 33%", scrub: 2, markers: false} });
```

## Voorbeelden met tekst (TextPlugin)

**Steeds toevoegen:** `gsap.registerPlugin(TextPlugin);`

```
gsap.to(".tekst",
    {
        text: "Dit vervangt de huidige tekst",
        duration: 3
    });

gsap.to(".tekst",
    {
        text: "Dit vervangt de huidige tekst ",
        scrollTrigger: {trigger:".tekst", start: "0% 60%", end: "0% 30%", scrub: 2}
    });
```

## Werken met paden (MotionPathPlugin)

Steeds toevoegen: `gsap.registerPlugin(MotionPathPlugin);`

<https://boxy-svg.com/app> (google Chrome)

```
// ---> werken met een PATH
// -----
let bewegenViaPath;
function createBewegenViaPath() {
  let progress=0; if (bewegenViaPath) {progress=bewegenViaPath.progress();
  bewegenViaPath.kill();}

  bewegenViaPath =

  // hieronder staat de eigenlijke animatie:
  animatie1.from("#miniLeaf1",
    {motionPath: {path: ".curve1", align: ".curve1", alignOrigin: [0.5, 0.5], start: 1, end: 0, autoRotation: true},
    duration: 3
  });
  // hieronder staat nog een animatie:
  animatie1.from("#miniLeaf2",
    {motionPath: {path: ".curve2", align: ".curve2", alignOrigin: [0.5, 0.5], start: 1, end: 0, autoRotation: true},
    duration: 6
  });

  bewegenViaPath.progress(progress);
};

createBewegenViaPath();
window.addEventListener("resize", createBewegenViaPath);
// -----
```

### Opmerkingen:

- Maak het path onzichtbaar in CSS:

```
curve1, curve2 {
  visibility: hidden;
}
```

- Zet je SVG-code in de HTML, en maak nieuwe **div** aan !:

```
<div class="curves">
```

```
<svg viewBox="0 0 1280 720" xmlns="http://www.w3.org/2000/svg">
<defs></defs>
<path class="curve1" style="stroke: rgb(0, 0, 0); fill: none;" d="M 0 31.929 .943"></path>
<path class="curve2" style="stroke: rgb(0, 0, 0); fill: none;" d="M -13. 667 707.848"></path>
</svg>
```

```
</div>
```

- Zorg dat de div van je SVG-code ook in de CSS staat:

```
.curves svg{
  position: absolute;
  top: 0%;
  width: 100%;
  height: 100%;
  z-index: 10;
}
```