

# TUTORIEL “ANIMATE”

Comment mettre une animation en javascript qui se déclenche sur le scrolling d'une page html avec ScrollMagic et GreenSock.

## Nous allons utiliser plusieurs librairies Javascript :

- la librairie **ScrollMagic** v2.0.7 pour créer une animation :

<https://github.com/janpaepke/ScrollMagic>

ScrollMagic est un contrôleur d'interaction de défilement. chaque scène est utilisée pour définir une action quand le container est scrollé d'une progression spécifique.

Cette librairie permet de réagir à la position courante du scroll afin d'animer un objet.

pour installer cette librairie, ajouter les lignes suivantes en fin de <body> :

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/ScrollMagic/2.0.7/ScrollMagic.min.js"
    integrity="sha256-2p2tRZ1Powp3P/04Pw2rqVCSbhyV/IB7ZEVUglrDS/c="
crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/ScrollMagic/2.0.7/plugins/debug.addIndicators.js"
    integrity="sha256-31FC/OT6XpfjAhj9FuXjw5/wPXXawCAjJQ29E23/XPk="
crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/ScrollMagic/2.0.7/plugins/animation.gsap.min.js"
    integrity="sha256-+9YNuItWuR4sbqeaNiJ0xG0BvptYz4fbUXbIZoH5Jwo="
crossorigin="anonymous"></script>
```

- la librairie **TweenLite** de la plateforme d'animation de GreenSock (GSAP) pour gérer l'interpolation d'une ou plusieurs propriétés de l'objet animé au fil du temps :

<https://greensock.com/docs/v2/TweenLite>

pour installer cette librairie, ajouter les lignes suivantes en fin de <body> :

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/2.1.3/TweenLite.min.js"
    integrity="sha256-VV47uJSOHzUeiBcCs3FcB0QLMn++yeG/zqZvaUkvGZM="
crossorigin="anonymous"></script>
```

- la librairie **TimelineLite** pour gérer le séquençement de tweens et autres timelines, facilitant ainsi leur contrôle global et la gestion précise de leur minutage:

<https://greensock.com/docs/v2/TimelineLite>

pour installer cette librairie, ajouter les lignes suivantes en fin de <body> :

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/2.1.3/TimelineLite.min.js"
    integrity="sha256-nuhNsfXzBFR6G1lKP8bK77dakkQDqHcQ40CFZvk6Qo="
crossorigin="anonymous"></script>
```

- les plugins **CSSPlugin** et **BezierPlugin** pour définir un chemin de Bézier incurvé:

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/2.1.3/plugins/CSSPlugin.min.js"
    integrity="sha256-LBjlnPrM6Aig8LDFc9PJctPHLGUC6RaUvnmXE4hV5Y="
crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/gsap/2.1.3/plugins/BezierPlugin.min.js"
```

```
integrity="sha256-5jxCMRC1PYU02qJn+fj+DPuxcQZCh0DR8GRwi4iKoRc="
crossorigin="anonymous"></script>
```

## Voici la procédure à suivre :

- Créer un fichier index.html avec

- un lien sur googlefonts pour la police de caractères:

```
<link href="https://fonts.googleapis.com/css?family=Roboto&display=swap"
rel="stylesheet">
```

- un lien sur notre fichier de style : style.css

```
<link rel="stylesheet" href="style.css">
```

- dans le body :

- un entête (header) contenant le titre h1 "Scroll animation"
- une section de classe "animation" contenant une image de classe "paper-plane", d'un fichier "paper.png",
- un footer contenant un titre h1 "cool"
- les lignes de scripts pour utiliser scrollMagic (voir ci-dessus)
- la ligne pour lier notre script app.js :

```
<script src="app.js"></script>
```

- Créer le fichier **style.css** pour définir le style général de la page

- personnaliser tout sans marge, sans padding, avec un 'border-box'

```
*{margin: 0;
padding: 0;
box-sizing: border-box;
}
```

- personnaliser le header et le footer : taille à 100vh pour occuper tout l'écran, display flex, centré en largeur et en hauteur, avec la police Roboto de google fonts (et Arial sans-serif par défaut)

```
header, footer{
height: 100vh;
display: flex;
justify-content: center;
align-items: center;
font-family: 'Roboto', Arial, sans-serif;
}
```

- personnaliser la police des titres h1 avec une taille de 60px

```
header h1{
font-size: 60px;
}
```

- personnaliser la taille de la section "animation" à 100vh pour occuper tout l'écran, un fond dégradé, une position relative et un overflow "hidden"

```
.animation{
```

```

height: 100vh;
background-image: linear-gradient(45deg, #ff9a9e 0%, #fad0c4 99%, #fad0c4
100%);
position: relative;
overflow: hidden;
}

```

- personnaliser l'image de classe "paper-plane" avec une taille de 100px, une position relative, pour décaler le haut (top) de 50% et la gauche (left) de 0%

```

.paper-plane{
height: 100px;
position: relative;
top: 50%;
left: 0%;
}

```

- Créer un fichier **app.js** pour coder l'animation en utilisant la librairie scrollMagic :

- La classe TweenLite permet de créer une interpolation sur un objet, qui utilisera un chemin de Bézier. On va donc déclarer un JSON en constante, nommé "flightPath", contenant les données suivantes qui décrivent les paramètres du chemin de Bézier à parcourir (pour le plugin bezier) :

```

const flightPath = {
  curviness: 1.25,
  autoRotate: true,
  values: [
    { x: 100, y: -20 },
    { x: 300, y: 10 },
    { x: 500, y: 100 },
    { x: 750, y: -100 },
    { x: 350, y: -50 },
    { x: 600, y: 100 },
    { x: 800, y: 0 },
    { x: window.innerWidth, y: -250 }
  ]
}

```

où :

**curviness** : règle la tension sur le Bézier (1 correspond à la courbure normale)

**autoRotate: true** fait pivoter automatiquement la cible en fonction de sa position sur le chemin de Bézier (true indique qu'il n'est pas besoin de décaler la rotation d'un certain montant de plus que la normale)

**values**: un tableau de valeurs des coordonnées. le tableau de vos valeurs de Bézier sous forme d'objets génériques. Chaque objet du tableau doit avoir des noms de propriété correspondants (comme "x" et "y").

puis on crée une instance de la classe de séquençage :

```
const tween = new TimelineLite();
```

à laquelle on ajoute une séquence, contenant une interpolation avec valeurs de destination sur la classe "paper-plane" de l'image à animer avec l'autre librairie TweenLite , en utilisant les paramètres définis dans le JSON flightPath pour le chemin de Bézier incurvé

(<https://greensock.com/docs/v2/Plugins/BezierPlugin>) :

```
tween.add(  
  TweenLite.to(".paper-plane", 1, {  
    bezier: flightPath,  
    ease: Power1.easeInOut  
  })  
);
```

- Le modèle de conception ScrollMagic est constitué d'un contrôleur auquel une ou plusieurs scènes sont associées. Chaque scène est utilisée pour définir ce qui se passe lorsque le scroll défile jusqu'à un décalage spécifique.
- On crée donc une instance de la classe ScrollMagic.Controller pour initialiser ce contrôleur:

```
const controller = new ScrollMagic.Controller();
```

- Puis on crée une scène avec une instance de la classe ScrollMagic.Scene en lui précisant l'élément de départ (la section de classe "animation" qui contient l'image à animer), la progression de la scène en termes d'unités de scrolling (duration), on lui fournit l'objet de séquençage créé précédemment (tween), puis on ajoute cette scène au contrôleur par "addTo(controller):

```
const scene = new ScrollMagic.Scene({  
  triggerElement: '.animation',  
  duration: 500,  
  triggerHook: 0.4  
})  
  .setTween(tween)  
  // .addIndicators() //optional  
  .addTo(controller);
```

**Lancer index.html dans le navigateur, et scroller pour lancer l'animation.**