

Introduction à l'animation en CSS

Les animations en CSS permettent de créer des effets visuels dynamiques et interactifs sur des éléments HTML sans avoir besoin de JavaScript. Ces animations peuvent rendre une page plus attrayante et interactive.

En CSS, nous avons deux principaux concepts d'animation :

- Les transitions
- Les animations clés (keyframes)

2. Les Transitions en CSS

Une transition permet de changer la valeur d'une propriété CSS d'un état à un autre sur une période de temps définie. Cela se fait généralement lors d'une interaction, comme un survol de souris.

Exemple de transition simple :

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Transition CSS</title>
  <style>
    .box {
      width: 100px;
      height: 100px;
      background-color: red;
      transition: background-color 0.5s ease, transform 0.5s ease;
    }

    .box:hover {
      background-color: blue;
      transform: rotate(45deg);
    }
  </style>
</head>
<body>
  <div class="box"></div>
</body>
</html>
```

Explication :

- L'élément .box commence avec un fond rouge.
- Lorsqu'on survole cet élément avec la souris (:hover), le fond devient bleu et l'élément est tourné de 45 degrés.
- La transition se fait sur 0.5 seconde grâce à transition: background-color 0.5s ease.

3. Les Animations avec Keyframes en CSS

Les **keyframes** sont utilisées pour créer des animations complexes qui modifient les propriétés d'un élément à différents points dans le temps.

Syntaxe des keyframes :

```
@keyframes nom-de-l-animation {
  0% {
    /* état initial */
  }
  50% {
    /* état à mi-chemin */
  }
  100% {
    /* état final */
  }
}
```

Exemple d'animation avec keyframes :

```
<!DOCTYPE html>
<html lang="fr">
```

```

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Animation CSS avec Keyframes</title>
  <style>
    .animation-box {
      width: 100px;
      height: 100px;
      background-color: green;
      animation: slide 3s infinite; /* 3s pour une animation de 3 secondes, répétée à l'infini */
    }

    @keyframes slide {
      0% {
        transform: translateX(0);
        background-color: green;
      }
      50% {
        transform: translateX(200px);
        background-color: yellow;
      }
      100% {
        transform: translateX(0);
        background-color: green;
      }
    }
  </style>
</head>
<body>
  <div class="animation-box"></div>
</body>
</html>

```

Explication :

- L'élément `.animation-box` se déplace de 0 à 200 pixels sur l'axe X, puis revient à sa position initiale.
- L'animation change également la couleur de fond de l'élément à chaque étape.
- L'animation dure 3 secondes et se répète indéfiniment grâce à `infinite`.

4. Propriétés importantes des animations en CSS

- **animation-name** : Le nom de l'animation (lié à un `@keyframes`).
- **animation-duration** : La durée de l'animation (par exemple, 2s pour 2 secondes).
- **animation-timing-function** : La courbe de timing de l'animation (ex: `ease`, `linear`, `ease-in`, `ease-out`, etc.).
- **animation-delay** : Le délai avant de commencer l'animation.
- **animation-iteration-count** : Le nombre de fois que l'animation sera répétée (par exemple, `infinite` pour une répétition infinie).
- **animation-direction** : Définit si l'animation doit revenir en arrière (`reverse`), alternée (`alternate`), etc.

Exemple combiné avec plusieurs propriétés :

```

@keyframes bounce {
  0%, 100% {
    transform: translateY(0);
  }
  50% {
    transform: translateY(-50px);
  }
}

.bouncing-box {
  width: 100px;
  height: 100px;

```

```
background-color: red;
animation: bounce 2s ease-in-out infinite;
}
```

5. Combiner Transitions et Animations

Parfois, vous pouvez utiliser des transitions et des animations en même temps pour créer des effets encore plus intéressants.

Exemple combiné :

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Transition et Animation combinées</title>
  <style>
    .animated-box {
      width: 100px;
      height: 100px;
      background-color: orange;
      transition: transform 0.5s ease;
      animation: scaleUp 2s infinite alternate;
    }

    @keyframes scaleUp {
      0% {
        transform: scale(1);
      }
      100% {
        transform: scale(1.5);
      }
    }

    .animated-box:hover {
      transform: rotate(45deg);
    }
  </style>
</head>
<body>
  <div class="animated-box"></div>
</body>
</html>
```

Dans cet exemple :

- **Animation** : L'élément `.animated-box` effectue une animation de mise à l'échelle qui alterne entre `scale(1)` et `scale(1.5)`.
- **Transition** : Lorsqu'on survole l'élément, il effectue une rotation de 45 degrés grâce à la propriété `transform`.