

# Séance 8 : Unités de Mesure et Transitions

Précision et Dynamisme en CSS

---

Fossé Stéphane

30 Mai 2025

IUT de Bordeaux

## Aperçu de la Séance 8

### Objectifs de la Séance

- Comprendre les différentes unités de mesure en CSS.
- Maîtriser les transitions CSS pour des animations fluides.
- Créer des interfaces plus précises, flexibles et dynamiques.

## 1. Introduction : Précision et Dynamisme en CSS

- Plongée dans deux aspects fondamentaux du CSS : les unités de mesure et les transitions CSS.
- Unités de mesure : cruciales pour des mises en page responsives et adaptatives.
- Transitions : clé pour ajouter des animations fluides et élégantes, améliorant l'expérience utilisateur.

## 2. Les Unités de Mesure en CSS

### 2.1 Unités Absolues (Fixes)

- Fixes, ne changent pas en fonction d'autres éléments.
- Utiles pour une taille précise et constante, moins pour le responsive.
- px (pixels) : unité la plus courante, fixe par rapport au périphérique.
- pt (points) : typographie pour l'impression (1pt = 1/72 de pouce).
- cm, mm, in : unités physiques, rarement utilisées pour le web.

```
.element-fixe {  
    width: 200px;  
    font-size: 16pt;  
}
```

### 2.2 Unités Relatives (Flexibles)

- Pierre angulaire du design responsive.
- Taille calculée par rapport à un autre élément.
- em : relative à la taille de police de l'élément parent.
- rem (root em) : relative à la taille de police de l'élément racine (<html>). Préférable pour cohérence.
- % (pourcentage) : relative à la taille de l'élément parent.
- vw (viewport width) : 1% de la largeur de la fenêtre d'affichage.
- vh (viewport height) : 1% de la hauteur de la fenêtre d'affichage.
- vmin (viewport minimum) : 1% de la plus petite dimension du viewport.
- vmax (viewport maximum) : 1% de la plus grande dimension du viewport.

## Exemples d'Unités Relatives

```
html {  
    font-size: 16px; /* Taille de base pour rem */  
}  
  
.container {  
    width: 80%; /* 80% de la largeur du parent */  
    padding: 1em; /* 16px si le parent a 16px de font-size */  
}  
  
.titre {  
    font-size: 2rem; /* 32px si html a 16px de font-size */  
    margin-bottom: 2vh; /* 2% de la hauteur du viewport */  
}  
  
.hero-image {  
    width: 100vw; /* Pleine largeur du viewport */
```

# Exemples d'Unités Relatives

```
height: 50vh; /* Moitié de la hauteur du viewport */  
}
```

# 3. Les Transitions CSS : Animer Vos Interfaces

## 3.1 Introduction aux Transitions

- Créent des changements d'état fluides entre les propriétés CSS.
- Animation douce sur une durée spécifiée au lieu d'un changement instantané.
- Parfaites pour `:hover`, `:active`, ou modifications de classe JavaScript.



### 3.2 Propriétés des Transitions

- `transition-property` : propriétés CSS à animer (ex: `width`, `background-color`, `opacity`, `transform`, `all`).
- `transition-duration` : durée de la transition (en s ou ms).
- `transition-timing-function` : courbe de vitesse (`ease`, `linear`, `ease-in`, `ease-out`, `ease-in-out`, `cubic-bezier`).
- `transition-delay` : délai avant le début de la transition.

## 3.3 Syntaxe Raccourcie (transition)

### 3.3 Syntaxe Raccourcie (transition)

- Combine toutes les propriétés en une seule ligne.

```
.bouton {  
    background-color: #3498db;  
    color: white;  
    padding: 10px 20px;  
    border-radius: 5px;  
    transition: background-color 0.3s ease-in-out, transform 0.2s ease;  
}
```

```
.bouton:hover {  
    background-color: #2980b9;  
    transform: scale(1.05);  
}
```

- Exemple : background-color en 0.3s ease-in-out, transform en 0.2s ease.

## 4. TP : Bouton Interactif et Carte Responsive

### Objectif

- Créer un bouton avec des effets de survol fluides.
- Concevoir une carte de contenu responsive et réactive au survol.

### Consignes

- Nouveau dossier `tp-seance8` avec `index.html` et `styles.css`.
- `index.html` : un bouton, une carte (`div.card`).
- `styles.css` :
  - `font-size` de base sur `<html>` (ex: 16px).
  - Bouton : `padding/font-size` en `em/rem`, `background-color`, `color`, `transition` sur `background-color` et `transform`. Styles `:hover` pour changement de couleur et `scale(1.1)`.

- Carte : width en vw ou %, padding/margin en rem, box-shadow, border-radius, transition sur box-shadow et transform. Au :hover, box-shadow augmentée et transform: translateY(-5px);. Texte en rem.
- Vérification : Tester bouton, carte, et réactivité.

## Fin de la Séance 8

### Récapitulatif

- Maîtrise des unités de mesure absolues et relatives.
- Création d'animations fluides avec les transitions CSS.
- Conception d'interfaces plus dynamiques et responsives.

### Prochaines Étapes

- Approfondissement des formulaires HTML/CSS.