

Séance 6 : Positionnement Relatif et Absolu

Contrôle Précis des Éléments

Fossé Stéphane

30 Mai 2025

IUT de Bordeaux

Aperçu de la Séance 6

Objectifs de la Séance

- Maîtriser les différentes valeurs de la propriété position.
- Comprendre le rôle de z-index pour gérer l'empilement.
- Découvrir la pseudo-classe :target pour l'interactivité CSS.

1. Maîtriser le Positionnement CSS

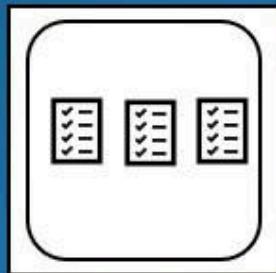
1.1 Introduction au Positionnement CSS

- Modifier le flux normal du document.
- Positionner un élément de manière spécifique.

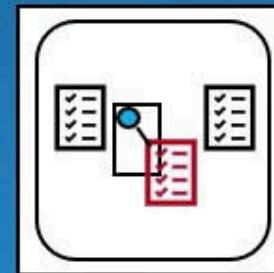
1.1 Introduction au Positionnement CSS

CSS Position

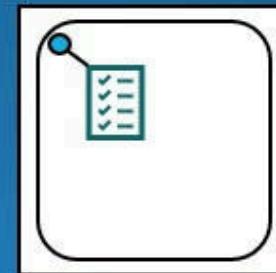
Static



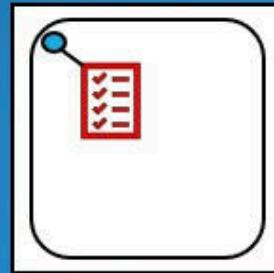
Relative



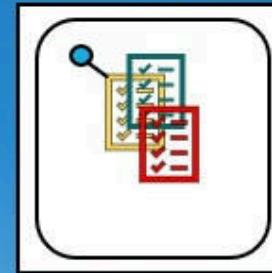
Fixed



Absolute



Z-index



1.2 Positionnement Statique (static)

- Valeur par défaut, éléments dans l'ordre normal.
- top, bottom, left, right n'ont aucun effet.

```
.element-static {  
    position: static; /* C'est la valeur par défaut, souvent non déclarée */  
    top: 20px; /* N'aura aucun effet */  
}
```

1.3 Positionnement Relatif (relative)

- Positionné par rapport à sa **position normale**.
- Décalé par top, bottom, left, right.
- L'espace qu'il aurait occupé **reste réservé**.

```
<div class="container">
    <div class="box">Boîte 1</div>
    <div class="box relative-box">Boîte Relative</div>
    <div class="box">Boîte 3</div>
</div>

.container {
    border: 2px dashed #A5B4FC;
    padding: 20px;
    display: flex;
}
.box {
```

1.3 Positionnement Relatif (relative)

```
width: 80px;  
height: 80px;  
background-color: #BFDBFE;  
border: 1px solid #60A5FA;  
margin: 10px;  
display: flex;  
justify-content: center;  
align-items: center;  
}  
.relative-box {  
position: relative;  
top: 20px; /* Décalé de 20px vers le bas par rapport à sa position normale */  
left: 20px; /* Décalé de 20px vers la droite par rapport à sa position normale */  
background-color: #FCA5A5; /* lightcoral */  
}
```

1.4 Positionnement Absolu (absolute)

- Retiré du flux normal du document.
- Positionné par rapport à son **ancêtre positionné le plus proche**.
- Règle d'or: parent doit avoir position autre que static.

```
<div class="parent-relative">
    Parent Relative
    <div class="absolute-box">Boîte Absolue</div>
</div>
<div class="sibling-box">Boîte Frère</div>

.parent-relative {
    position: relative; /* Le parent est positionné */
    width: 300px;
    height: 200px;
    border: 2px solid #4F46E5; /* indigo-600 */
    padding: 20px;
```

1.4 Positionnement Absolu (absolute)

```
margin-bottom: 20px;  
background-color: #EEF2FF; /* indigo-50 */  
}  
.absolute-box {  
    position: absolute;  
    top: 10px;  
    right: 10px;  
    width: 100px;  
    height: 100px;  
    background-color: #FCA5A5; /* lightcoral */  
    border: 1px solid #DC2626; /* red-600 */  
    display: flex;  
    justify-content: center;  
    align-items: center;  
}  
.sibling-box {  
    width: 150px;
```

1.4 Positionnement Absolu (absolute)

```
height: 50px;  
background-color: #BFDBFE;  
border: 1px solid #60A5FA;  
margin-top: 10px; /* La boîte absolue ne l'affecte plus */  
}
```

1.5 Positionnement Fixe (fixed)

- Positionné par rapport à la **fenêtre d'affichage (viewport)**.
- Retiré du flux normal, reste au même endroit au défilement.
- Idéal pour barres de navigation persistantes, boutons “retour en haut”.

```
<div class="fixed-header">En-tête Fixe</div>
<div class="content-placeholder">
    <p>Contenu de la page... (faites défiler pour voir l'effet du fixe)</p>
    <!-- Répétez ce paragraphe plusieurs fois pour créer du défilement -->
</div>

.fixed-header {
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    background-color: #4F46E5; /* indigo-600 */
```

1.5 Positionnement Fixe (fixed)

```
color: white;
padding: 15px;
text-align: center;
z-index: 1000; /* Pour s'assurer qu'il est au-dessus du contenu */
}
.content-placeholder {
    margin-top: 60px; /* Pour éviter que le contenu ne soit caché par l'en-tête
fixe */
    padding: 20px;
    height: 1200px; /* Pour forcer le défilement */
    background-color: #F8FAFC;
}
```

1.6 Positionnement Sticky (sticky)

- Hybride entre `relative` et `fixed`.
- Se comporte comme `relative` jusqu'à un seuil de défilement, puis devient "fixe".
- Parfait pour en-têtes de section.

```
<div class="scroll-container">
    <h3>Titre Sticky</h3>
    <p>Contenu qui défile...</p>
    <!-- Répétez ce paragraphe plusieurs fois -->
    <p>Plus de contenu...</p>
</div>

.scroll-container {
    height: 400px; /* Conteneur avec défilement */
    overflow-y: scroll;
    border: 2px solid #10B981; /* emerald-500 */
    padding: 10px;
```

1.6 Positionnement Sticky (sticky)

```
}

.sticky-header {
    position: sticky;
    top: 0; /* Devient fixe à 0px du haut du conteneur parent en défilement */
    background-color: #D1FAE5; /* emerald-100 */
    padding: 10px;
    border-bottom: 1px solid #10B981;
    z-index: 10; /* Pour s'assurer qu'il est au-dessus du contenu */
}

.scroll-container p {
    margin-bottom: 15px;
    line-height: 1.6;
}
```

1.7 La Propriété z-index : Gérer les Calques

- Contrôle l'ordre d'empilement des éléments qui se chevauchent.
- Valeur plus élevée = au-dessus.
- Ne fonctionne que sur éléments avec position autre que static.

```
<div class="z-container">
    <div class="z-box z-box-1">Z-index 1</div>
    <div class="z-box z-box-2">Z-index 2</div>
    <div class="z-box z-box-3">Z-index 3</div>
</div>

.z-container {
    position: relative;
    width: 300px;
    height: 200px;
    border: 2px dashed #A5B4FC;
    margin: 30px;
```

1.7 La Propriété z-index : Gérer les Calques

```
}

.z-box {
    position: absolute;
    width: 100px;
    height: 100px;
    border: 1px solid white;
    display: flex;
    justify-content: center;
    align-items: center;
    color: white;
}

.z-box-1 {
    background-color: #EF4444; /* red-500 */
    top: 10px;
    left: 10px;
    z-index: 1;
}
```

1.7 La Propriété z-index : Gérer les Calques

```
.z-box-2 {  
    background-color: #F97316; /* orange-500 */  
    top: 30px;  
    left: 30px;  
    z-index: 2;  
}  
.z-box-3 {  
    background-color: #EAB308; /* yellow-500 */  
    top: 50px;  
    left: 50px;  
    z-index: 3;  
}
```

1.8 Interactivité en Pur CSS : La Pseudo-classe :target

- Styliser un élément lorsqu'il est la "cible" de l'URL (id correspond au hash).
- Simule interactivité sans JavaScript.

Fin de la Séance 6

Récapitulatif

- Maîtrise des différentes propriétés de positionnement CSS.
- Compréhension de `z-index` pour l'ordre d'empilement.
- Utilisation de `:target` pour l'interactivité CSS.

Prochaines Étapes

- Synthèse de tous les concepts vus jusqu'à présent.