

Conception de documents et d'interfaces numériques

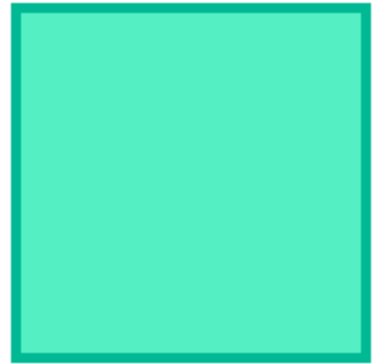
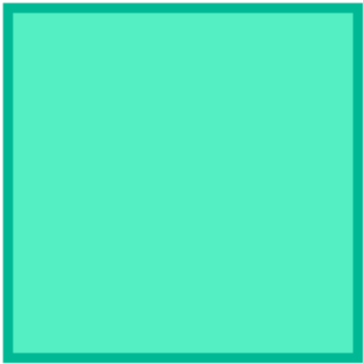
CSS : Positionnement avancé

```
<div class="container">  
  <div class="item"></div>  
  <div class="item relative"></div>  
  <div class="item"></div>  
</div>
```

```
.container {  
  display: flex;  
  justify-content: space-around;  
}
```

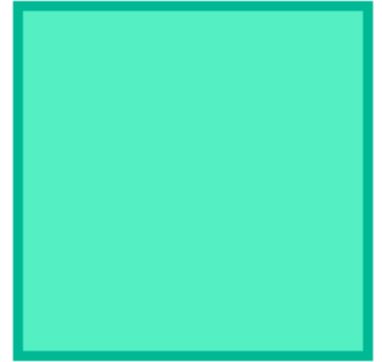
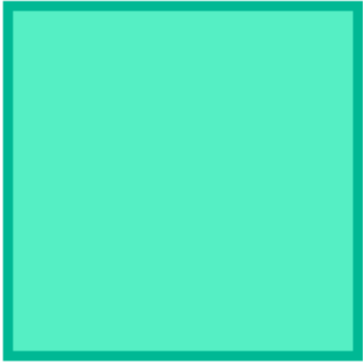
```
.item {  
  flex-basis: 180px;  
  height: 180px;  
  flex-shrink: 0;  
  flex-grow: 0;  
  background-color: #55efc4;  
  border: 5px solid #00b894;  
}
```

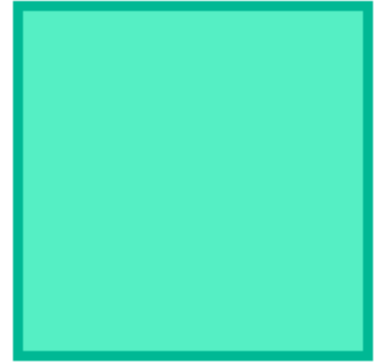
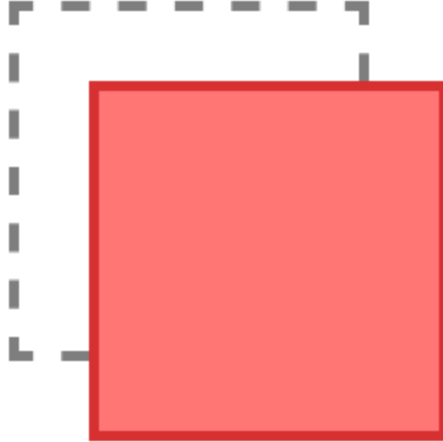
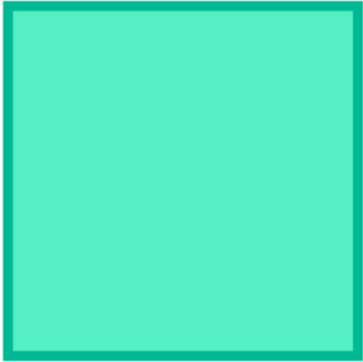
```
.relative {  
  background-color: #ff7675;  
  border-color: #d63031;  
}
```



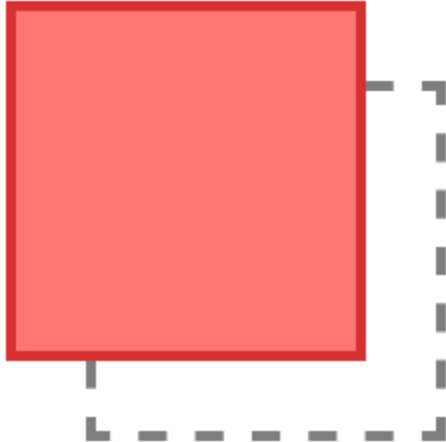
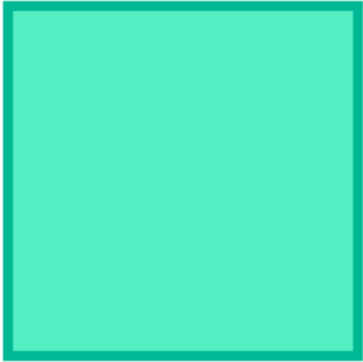
Positionnement relatif

```
.relative {  
  position: relative;  
  top: 40px;  
  left: 40px;  
}
```

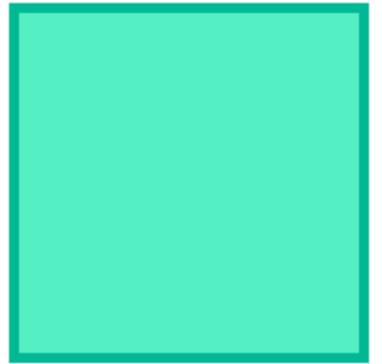
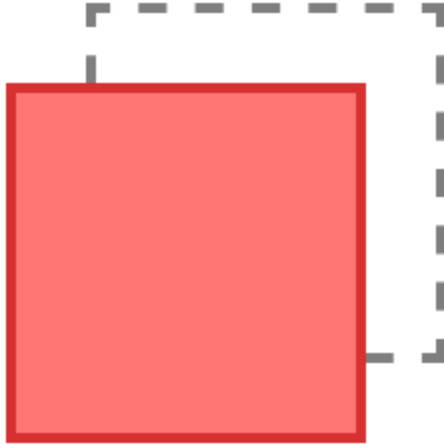
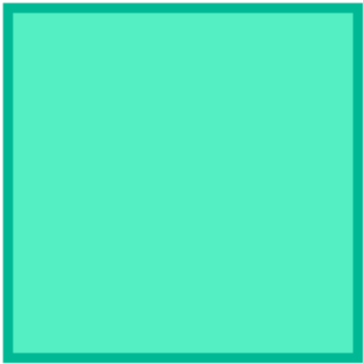





```
.relative {  
    position: relative;  
    bottom: 40px;  
    right: 40px;  
}
```



```
.relative {  
  position: relative;  
  bottom: 40px;  
  top: 40px;  
  right: 40px;  
}
```





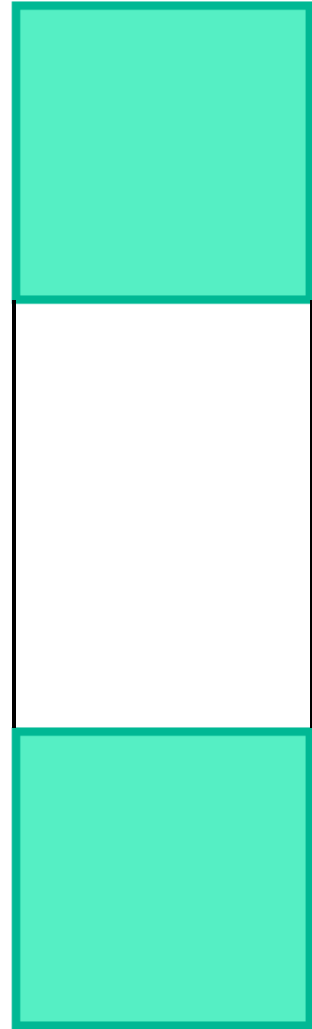
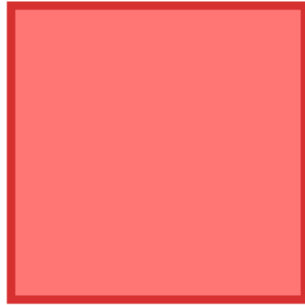
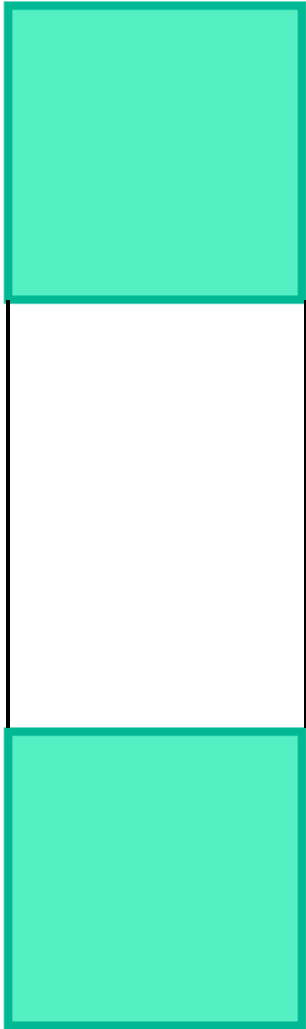
En positionnement relatif, si top et bottom sont spécifiés, top est appliqué et bottom est ignoré. De même pour left et right : left gagne à tous les coups.

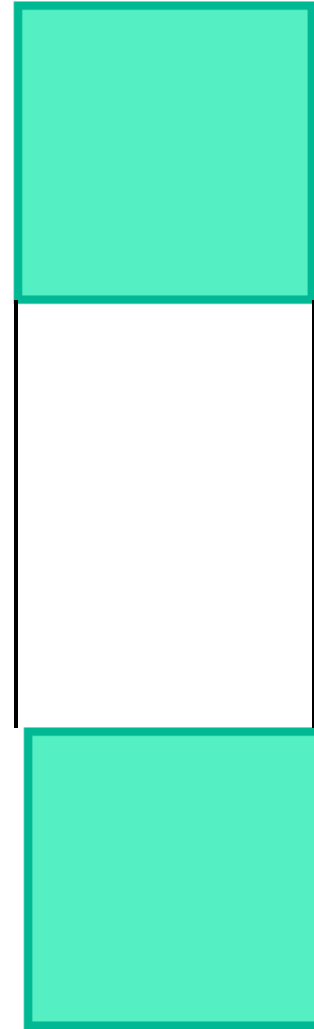
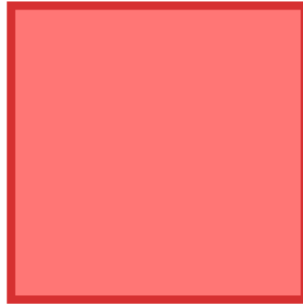
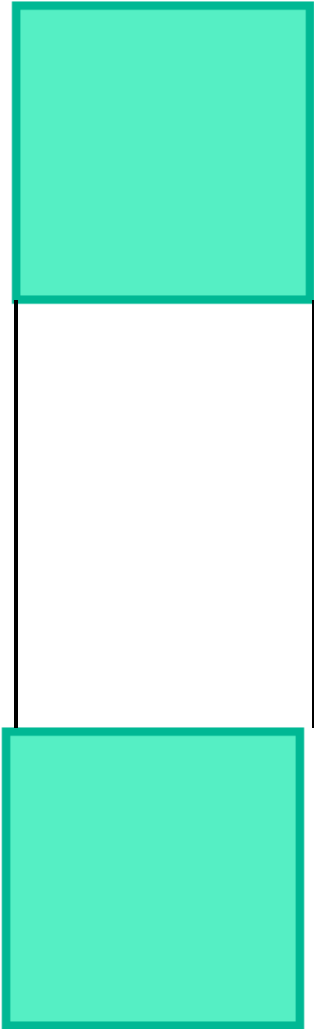


*On aurait pu décaler l'élément de la même manière
avec des margins, non ?*



La différence entre le positionnement relatif est des marges, c'est que le positionnement relatif n'affecte pas la position des éléments autour de l'élément positionné





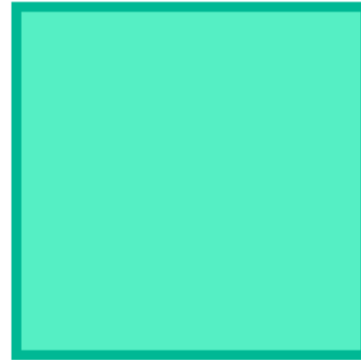
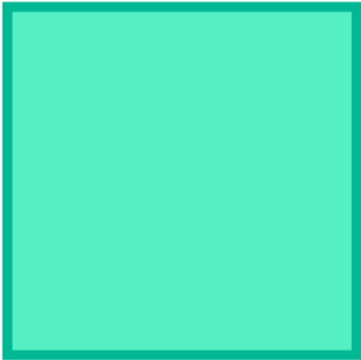


Il est aussi possible d'utiliser des valeurs négatives pour top, bottom, left et right. Par exemple, top: -30px est équivalent à bottom: 30px

Positionnement absolu

```
<div class="container">  
  <div class="item"></div>  
  <div class="item absolute"></div>  
  <div class="item"></div>  
</div>
```

```
.absolute {  
  position: absolute;  
  top: 40px;  
  left: 40px;  
}
```





*Le positionnement absolu prend comme origine
le coin supérieur gauche de la fenêtre*

Ce n'est pas exactement ça, mais disons, pour l'instant

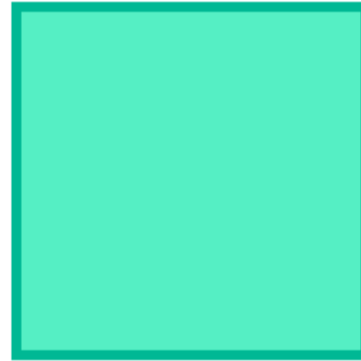
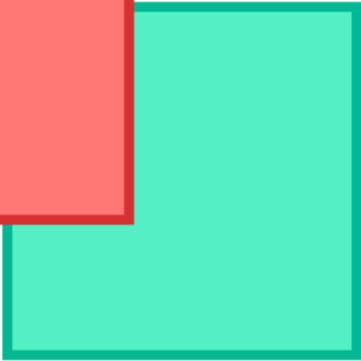
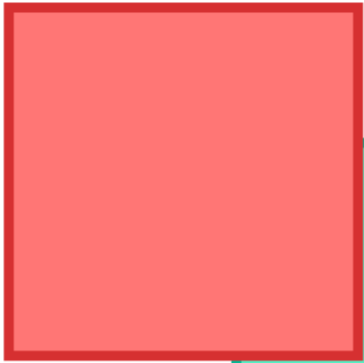


Un élément positionné de manière absolue prend la taille de son contenu si aucune taille n'est spécifiée.



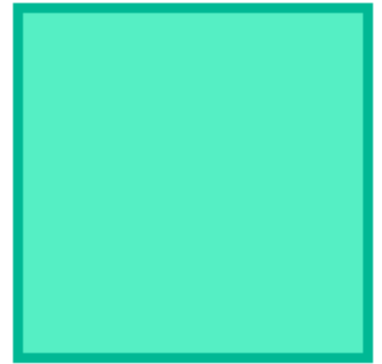
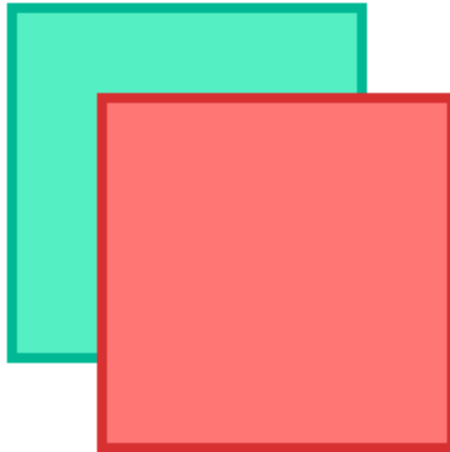
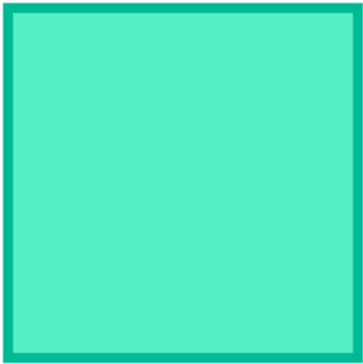
Contrairement au positionnement relatif, le positionnement absolu « libère » l'espace original de l'élément positionné. On dit que l'élément positionné est « sortit du flux »

```
.absolute {  
  position: absolute;  
  top: 40px;  
  left: 40px;  
  width: 180px;  
}
```




```
.relative {  
    position: relative;  
}
```

```
.absolute {  
    position: absolute;  
    top: 40px;  
    left: 40px;  
    width: 180px;  
}
```





Le positionnement absolu prend comme origine le coin supérieur gauche de son plus proche parent positionné. Si aucun ne l'est, alors il prend comme origine le coin supérieur gauche de la fenêtre

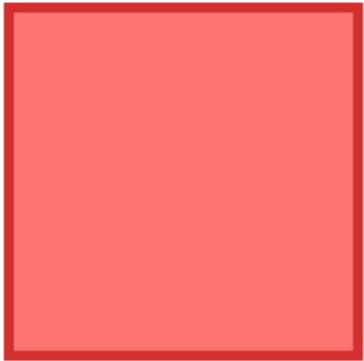


Un élément est dit « positionné » si la valeur de sa propriété « position » est autre que « static » (qui est la valeur par défaut de la propriété « position »)

Positionnement fixe

```
<div class="container">  
  <div class="item"></div>  
  <div class="item fixed"></div>  
  <div class="item"></div>  
</div>
```

```
.fixed {  
  position: fixed;  
  top: 40px;  
  left: 40px;  
  width: 180px;  
}
```





*Le positionnement fixe prend **toujours** le coin supérieur gauche de la fenêtre comme origine*

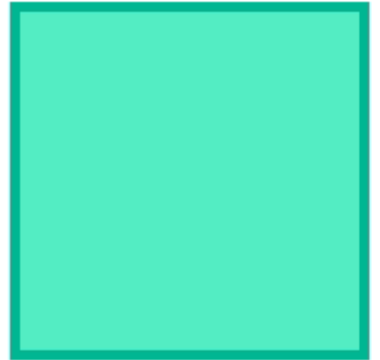
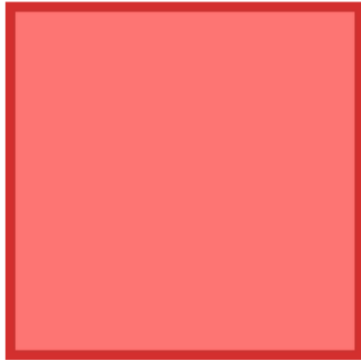
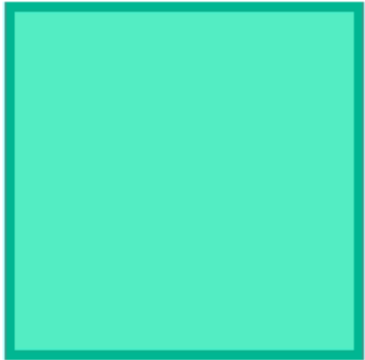
Le positionnement « sticky »

```
<div class="container">  
  <div class="item"></div>  
  <div class="item sticky"></div>  
  <div class="item"></div>  
</div>  
<div class="footer"></div>
```

```
.container {  
    height: 2000px;  
}
```

```
.footer {  
    height: 300px;  
    background-color: grey;  
}
```

```
.sticky {  
    position: sticky;  
    top: 40px;  
}
```



Un élément « sticky » est « static » tant que le niveau de scroll n'a pas atteint la valeur de top/bottom/left/right spécifié. Puis il devient « fixed » tant que le niveau de scroll n'a pas atteint l'extrémité de son élément parent

Le positionnement statique

```
.static {  
    position: static;  
}
```



*C'est la valeur par défaut de tous les éléments.
Elle est utile uniquement si on a modifié le
positionnement d'un élément, puis qu'on
souhaite le réinitialiser à sa valeur par défaut
(dans une media query, par exemple)*

TD