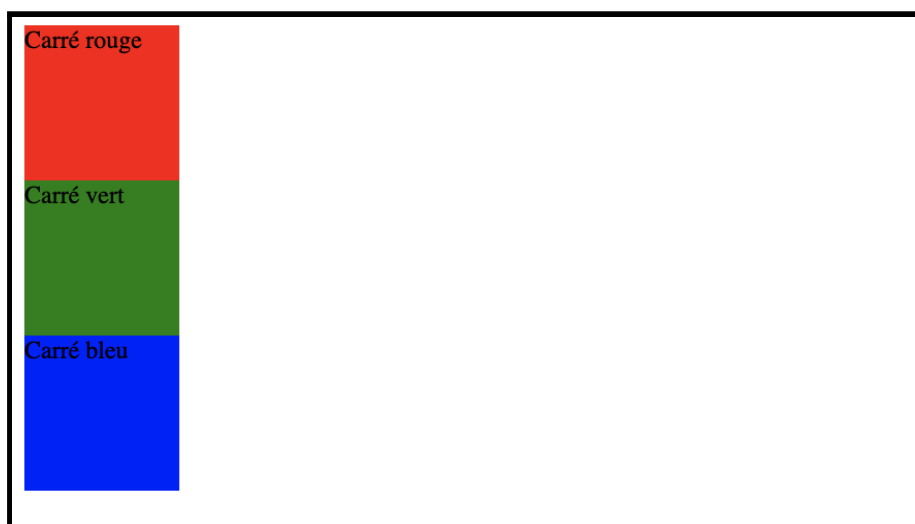


Display

display: block;

HTML :	CSS :
<pre><div class="carre rouge"> Carré rouge </div> <div class="carre vert"> Carré vert </div> <div class="carre bleu"> Carré bleu </div></pre>	<pre>.rouge { background-color: red; } .vert { background-color: green; } .bleu { background-color: blue; } .carre { width : 100px; height: 100px; display: block; }</pre>



Info : **display: block;** est déjà nativement sur les éléments de type **div**. Dans ce cas, nous n'étions pas obligé de rajouter cette ligne dans notre class *carre*.

display: inline;

HTML :	CSS :
<pre><div class="carre rouge"> Carré rouge </div> <div class="carre vert"> Carré vert </div> <div class="carre bleu"> Carré bleu </div></pre>	<pre>.rouge { background-color: red; } .vert { background-color: green; } .bleu { background-color: blue; } .carre { width : 100px; height: 100px; display: inline; }</pre>

Carré rouge Carré vert Carré bleu

Info : `display: inline;` prend la taille de notre contenu et non pas la taille de notre élément. Les propriétés `width` et `height` ne sont donc pas prises en compte.

display: inline-block;

HTML :	CSS :
<pre><div class="carre rouge"> Carré rouge </div> <div class="carre vert"> Carré vert </div> <div class="carre bleu"> Carré bleu </div></pre>	<pre>.rouge { background-color: red; } .vert { background-color: green; } .bleu { background-color: blue; } .carre { width : 100px; height: 100px; display: inline-block; }</pre>



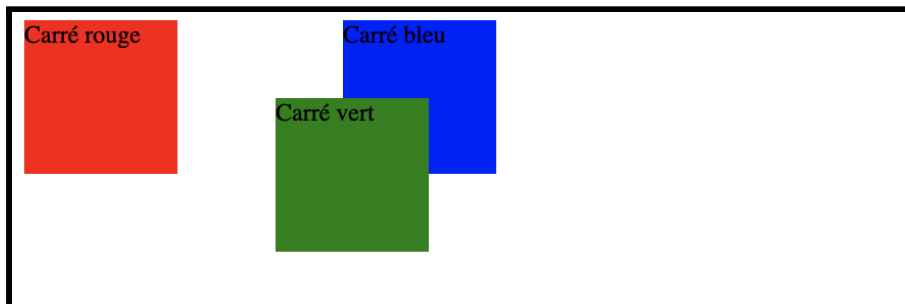
Info : `display: inline-block;` prend la taille de notre élément. Les propriétés `width` et `height` sont donc prises en compte.

Position

Info: Nativement, tous les éléments sont à `position: static;`

`position: relative;`

HTML :	CSS :
<pre><div class="carre rouge"> Carré rouge </div> <div class="carre vert"> Carré vert </div> <div class="carre bleu"> Carré bleu </div></pre>	<pre>.rouge { background-color: red; } .vert { background-color: green; } .bleu { background-color: blue; position: relative; top: 50px; left: 60px; } .carre { width : 100px; height: 100px; display: inline-block; }</pre>



Info : Le carré vert est décalé de 50px en haut et de 60px à gauche par rapport à sa **position original**.

position: absolute;

HTML :	CSS :
<pre><div class="carre rouge"> Carré rouge </div> <div class="carre vert"> Carré vert </div> <div class="carre bleu"> Carré bleu </div></pre>	<pre>.rouge { background-color: red; } .vert { background-color: green; } .bleu { background-color: blue; position: absolute; top: 50px; left: 60px; } .carre { width : 100px; height: 100px; display: inline-block; }</pre>



Info : Le carré vert est décalé de 50px en haut et de 60px à gauche par rapport à son élément parent direct. Le carré vert est sorti du flux de notre page web.

position: fixed;

HTML :	CSS :
<pre><div class="contenu"> <div class="carre rouge"> Carré rouge </div> <div class="carre vert"> Carré vert </div> <div class="carre bleu"> Carré bleu </div> </div></pre>	<pre>.rouge { background-color: red; } .vert { background-color: green; } .bleu { background-color: blue; position: fixed; top: 50px; left: 60px; } .carre { width : 100px; height: 100px; display: inline-block; } .contenu { height: 2000px; }</pre>



Info : Le carré vert est décalé de 50px en haut et de 60px à gauche par rapport à l'élément parent direct. Le carré vert a les mêmes caractéristiques de

positionnement que le position absolute. Seulement le carré vert sera fixe dans la page. Ajoutez une div englobante avec une taille de 2000px pour pouvoir constater le défilement de notre élément.