

Web Apps - CSS Layout

TOM PEETERS



Once upon a time

```
table
{
  width:300px;
  height:300px;

}

table,tr,td
{
  border-width: 1;
  border-style: solid;
}
</style>

</head>

<body>

<table>

  <tr><td>Once upon a tim</td><td rowspan="2">ROWSPAN</td></tr>

  <tr><td>HTML tables</td></tr>

  <tr><td colspan="2">where the most advanced thing there was</td></tr>

</table>
```

Once upon a tim	ROWSPAN
HTML tables	
where the most advanced thing there was	

Traditional Layout

Positioning

Display

Float

Z-index

Positioning

```
#id01
{
  position:absolute;
  top:25px;
  left:50px;
  z-index:0;
  background-color: orange;
}

#id02
{
  position:relative;
  top:25px;
  left:50px;
  z-index:1;
  background-color: green;
}
```

Div #id01
Div #id02

Positioning

Normal flow algoritme: men plaatst elementen van boven naar beneden, eventueel van links naar rechts

- Niet altijd nood aan, want scherm is breed genoeg om (blok)-elementen naast elkaar te zetten.
- Mogelijkheden om dit te doen zijn: tabel met meerdere kolommen maken
- Div structuren gebruiken en positioneren (*gebruiken als container elementen*)
- ...

Positioning: Relative

- Een “**relatief gepositioneerd**” element wordt relatief gepositioneerd tegenover zijn normale positie.

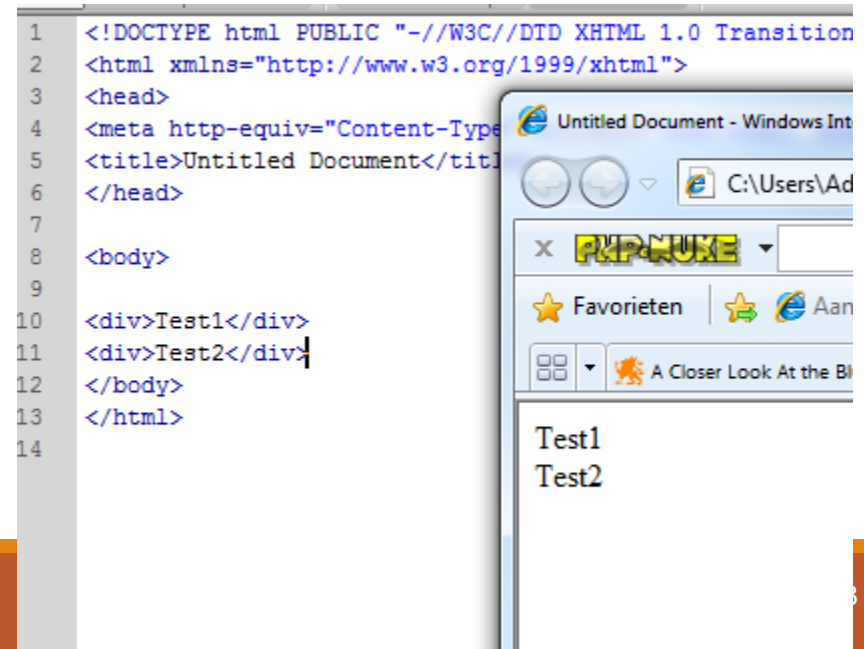
Positioning: Absolute

- Door een element (afbeelding, tabel, ...) absoluut te positioneren, zal deze op de exacte pixel lokatie terecht komen vanaf zijn parent
- Door absolute positionering kan men elementen laten overlappen

Positioning : float

- Floating laat toe om block elementen links of rechts van zijn parent element te aligneren, en de volgende elementen errond te laten “flowen”. Een floated element wordt automatisch een block element, dus kan je het een width en height geven (dit is dan ook aan te raden, anders kan het renderen van de pagina wel eens onvoorspelbare resultaten opleveren).
- Anders doen dan de normale flow: elementen floaten naar 1 kant.

• Vb.: Normal Flow =>



Positioning : float

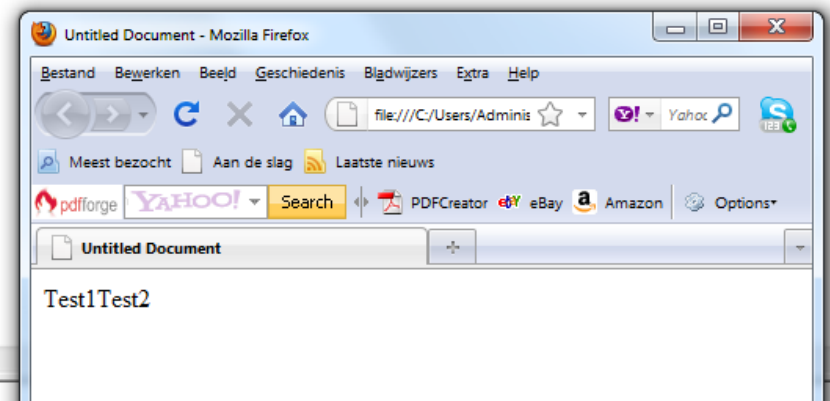
- Anders doen dan de normale flow: elementen floaten naar 1 kant. Veelal gebruikt om tekst rond een afbeelding te 'wrappen'
- Vb.: Float left =>

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Untitled Document</title>

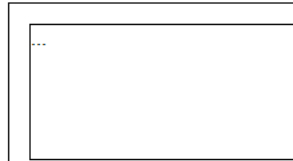
<link href="floatStijl.css" rel="stylesheet" type="text/css" />
</head>

<body>

<div>Test1</div>
<div>Test2</div>
</body>
</html>
```



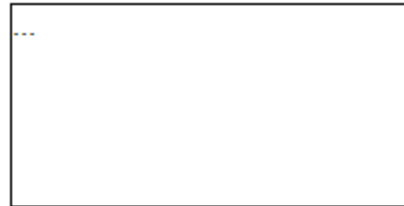
```
<style>
.box
{
  float: left;
  width: 200px;
  height: 100px;
  margin: 1em;
  border: 1px solid black;
}
```



>>

```
section
{
  border: 1px solid black;
}

</style>
```



... de tekst wordt gefloat rond de div met class box Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta risque velit egestas ac id est molestie. In hac habitasse platea dictumst.

```
<style>
.box
{
  float: left;
  width: 200px;
  height: 100px;
  margin: 1em;
  border: 1px solid black;
}
```

```
section
{
  clear: left;
}

</style>
```

De "section" is nu verschoven onder de floated div.
(je kan ook right en both meegeven)

Clear = geen floated elementen toegelaten aan linkse of rechtse of beide zijden van floated element

Oeps.. de afbeelding is groter dan het element dat het bevat is wordt gefloat, maw overflowing buiten de container

```
<style>
```

```
  img
  {
    float:right;

  }
  div
  {
    border:1px solid black;

  }
```

```
</style>
```



Oeps.. de afbeelding is groter dan het element dat het bevat is wordt gefloat, maw overflowing buiten de container

```
<style>
```

```
  img
  {
    float:right;

  }
  div
  {
    border:1px solid black;
    overflow:auto;

  }
```

```
</style>
```



Float layout voorbeeld

Maak het zelfde voorbeeld maar dan met “float”

```
nav
{
    float:left;
}
section
{
    margin-left:200px;
}
```

- Home
- Taco Menu
- Draft List
- Hours
- Directions
- Contact

... margin-left style for sections makes sure there is room for the nav. Otherwise the absolute and static elements would overlap

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.

Notice what happens when you resize your browser. It works nicely!

Pseudo Classes

Selector	Example	Example description
<u>:link</u>	<code>a:link</code>	Selects all unvisited links
<u>:visited</u>	<code>a:visited</code>	Selects all visited links
<u>:active</u>	<code>a:active</code>	Selects the active link
<u>:hover</u>	<code>a:hover</code>	Selects links on mouse over

- CSS pseudo-classes worden gebruikt voor speciale effecten op selecties
- Bijvoorbeeld => `a:visited {color:#00FF00;}`

Bouwen van een CSS Drop-Down Menu - 1

- Het menu in HTML:

```
<body>

<div class="menu">Menu
<a href="#">Home</a>
<a href="#">News</a>
<a href="#">Articles</a>
<a href="#">Blog</a>
<a href="#">Contact</a>

</div>
</body>
```

Bouwen van een CSS Drop-Down Menu - 2

- De CSS Style:

```
<style type="text/css">
div.menu
{
    font-weight:bold;
    color:white;
    border:2px solid #225599;
    text-align:center;
    background-color:#225599;
    width:100px;
}

div.menu:hover a {display:block;}

div.menu a {
    display:none;
    background-color:white;
    color:black;
    width:100px;
}

</style>
```

Display

Belangrijke CSS eigenschappen voor layouting. Elk element heeft een “default display”: inline / block

Via de display eigenschap kan je elementen “vervormen”

- display:block
- display:inline
- *Verschil met “**visibility**” & “**display none**” rendert de pagina alsof het element niet bestaat, terwijl visibility:hidden enkel het element verstoopt.*

Margin auto

Door het zetten van een breedte en margins op auto kan je een container centreren

```
<style>
  #wrap
  {
    margin: 0 auto;
    width: 500px;
    background-color: orange;
  }
</style>
</head>

<body>

  <div id="wrap">

    wrapper
  </div>
```



wrapper

Verbeterde versie

```
<style>
  #wrap
  {
    margin: 0 auto;
    max-width: 500px;
    background-color: orange;
  }
</style>
</head>

<body>

  <div id="wrap">

    wrapper
  </div>
```



- *Max-width verbetert hoe de browser op kleinere schermen reageert*

Box sizing

Box-sizing is een nieuwe CSS eigenschap: Wanneer je box-sizing op border-box zet zal de border en padding dit element niet meer groter maken. *Prefixes for opera, chrome, IE, safari..*

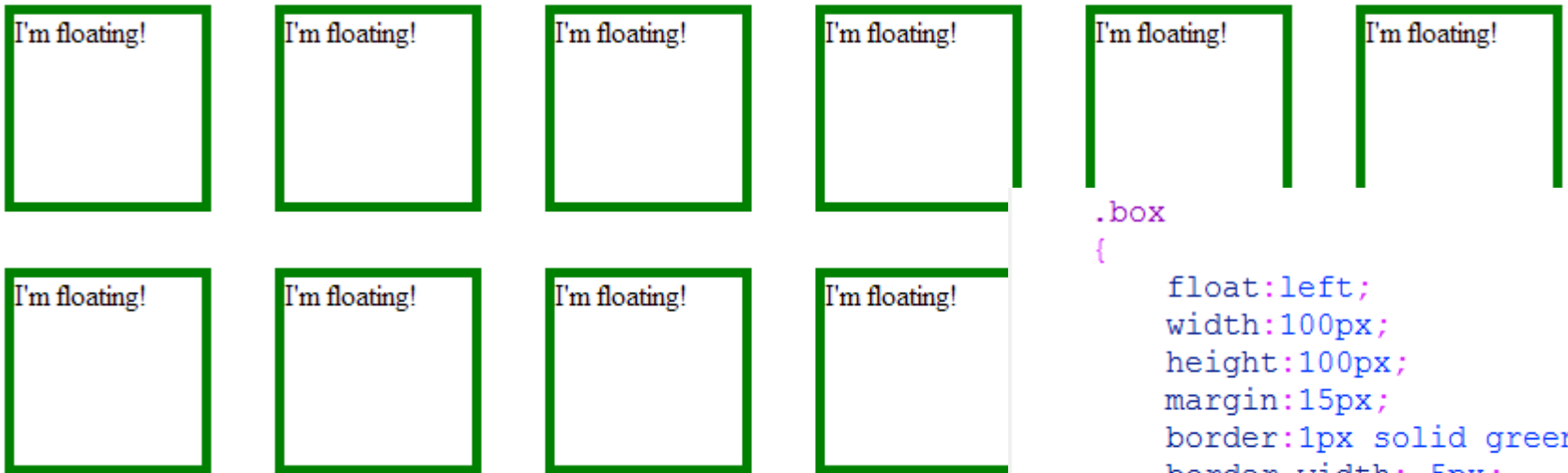
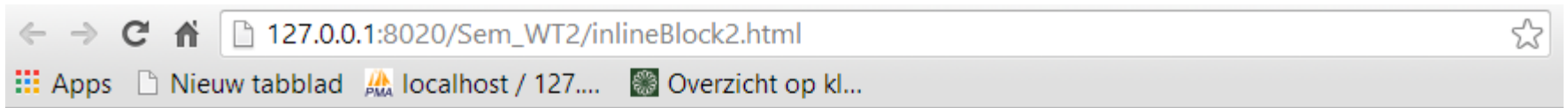
```
.simple {  
  width: 500px;  
  margin: 20px auto;  
  border: solid black 1px;  
  -webkit-box-sizing: border-box;  
  -moz-box-sizing: border-box;  
  box-sizing: border-box;  
}  
  
.fancy {  
  width: 500px;  
  margin: 20px auto;  
  padding: 50px;  
  border: solid blue 10px;  
  -webkit-box-sizing: border-box;  
  -moz-box-sizing: border-box;  
  box-sizing: border-box;  
}
```

Ik ben kleiner..

Ik ben groter..

Firefox: -moz-
Internet Explorer: -ms-
iOS: -webkit-
Opera: -o-
Safari: -webkit-

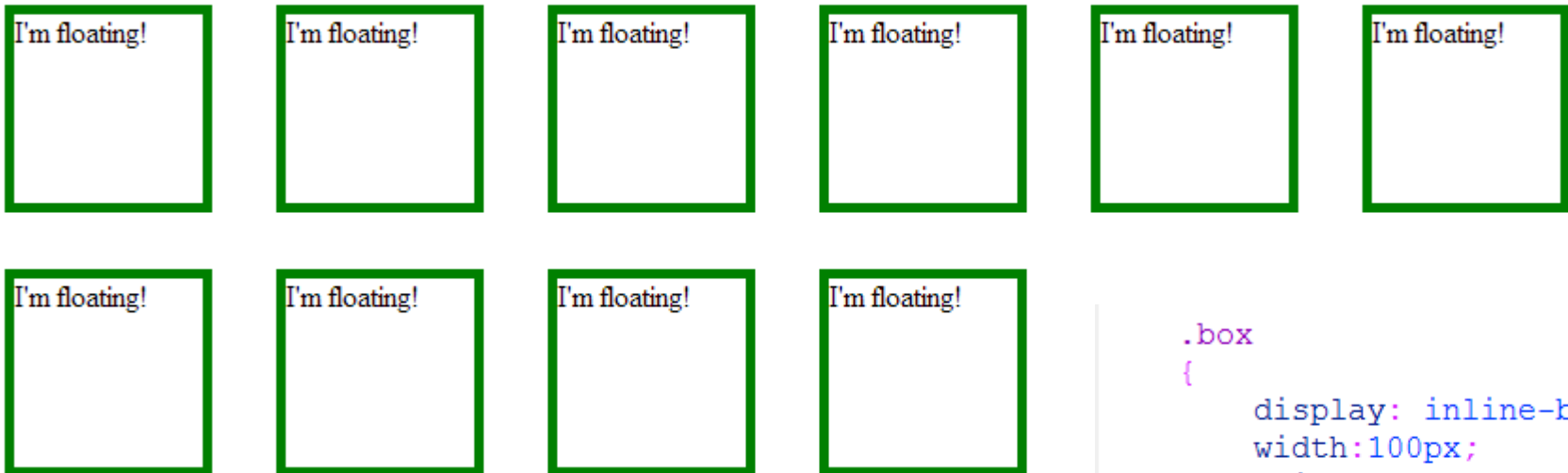
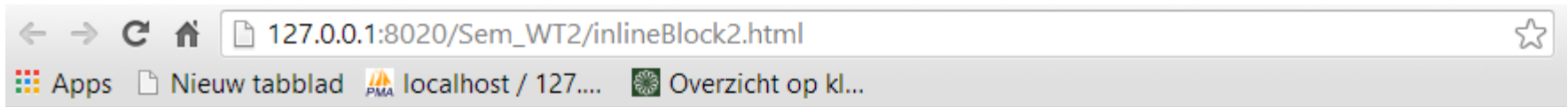
Inline block (toon blok als een span-element)



I'm Clear !

```
.box
{
    float:left;
    width:100px;
    height:100px;
    margin:15px;
    border:1px solid green;
    border-width: 5px;
}
.after-box
{
    clear:left;
    border:1px solid black;
}
```

Inline block



I'm Clear !



(geen clear nodig!)

```
.box
{
    display: inline-block;
    width:100px;
    height:100px;
    margin:15px;
    border:1px solid green;
    border-width: 5px;
}
</style>
```

Layout met inline-block

```
<style>
  nav
  {
    display:inline-block;
    width:20%;
    vertical-align:top;
    border:1px solid black;
  }
  .column
  {
    display:inline-block;
    width:75%;
    vertical-align:top;
    border:1px solid black;
  }
</style>
```

- Home
- Taco Menu
- Draft List
- Hours
- Directions
- Contact

Tada!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.

Steeds vertical-align eigenschap meegeven (op top zetten waarschijnlijk)

Steeds width specificieëren

Column – nieuwe CSS eigenschap

```
<style>
```

```
.five-column {  
  padding: 10px;  
  -moz-column-count: 5;  
  -moz-column-gap: 10px;  
  -webkit-column-count: 5;  
  -webkit-column-gap: 10px;  
  column-count: 5;  
  column-gap: 10px;  
}
```

```
</style>
```

Lorem ipsum dolor sit
amet, consectetur
adipiscing elit.
Phasellus imperdiet,
nulla et dictum
interdum, nisi lorem
egestas odio, vitae
scelerisque enim ligula

venenatis dolor.
Maecenas nisl est,
ultrices nec congue
eget, auctor vitae
massa. Fusce luctus
vestibulum augue ut
aliquet. Mauris ante
ligula, facilisis sed

ornare eu, lobortis in
odio. Praesent
convallis urna a lacus
interdum ut hendrerit
risus congue. Nunc
sagittis dictum nisi, sed
ullamcorper ipsum
dignissim ac. In at

libero sed nunc
venenatis imperdiet
sed ornare turpis.
Donec vitae dui eget
tellus gravida
venenatis. Integer
fringilla congue eros
non fermentum. Sed

dapibus pulvinar nibh
tempor porta. Cras ac
leo purus. Mauris quis
diam velit.