

Application of Location Attribute Based on CSS in Web Design

Kui Gao, Xiaocui Fu, Weiyan Li

Information Science and Engineering, Shandong Agricultural University, Tai'an Shandong
Email: gaokui@sdau.edu.cn, fxckey@sdau.edu.cn, liweiyan@sdau.edu.cn

Received: June 6th, 2019; accepted: June 21st, 2019; published: June 28th, 2019

Abstract

In web design, the location of web elements can be set through the location attributes of CSS, and then the layout of the page can be realized. The location attributes of CSS have different attribute values, which determine the different location methods of web elements. This paper compares the positioning attributes of CSS with examples, and analyzes and explains the methods and points for attention of element positioning using different positioning attributes of CSS, so as to achieve the purpose of mastering.

Keywords

CSS, Position, Static, Relative, Absolute, Fixed

基于CSS的定位属性在网页设计中的应用

高 葵, 付晓翠, 李蔚妍

山东农业大学信息科学与工程学院, 山东 泰安
Email: gaokui@sdau.edu.cn, fxckey@sdau.edu.cn, liweiyan@sdau.edu.cn

收稿日期: 2019年6月6日; 录用日期: 2019年6月21日; 发布日期: 2019年6月28日

摘 要

在网页设计中, 可以通过CSS的定位属性实现网页元素的定位设置, 进而实现页面布局。CSS的定位属性由不同的属性值决定了网页元素不同的定位方式。本文通过举例对CSS的定位属性进行比较, 并分析说明, 给出使用不同CSS的定位属性法进行元素定位的使用方法和注意事项, 达到掌握的目的。

关键词

CSS, 定位属性, 静态定位, 相对定位, 绝对定位, 固定定位

Copyright © 2019 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

在网页设计中, CSS 即层叠样式表, 用来控制网页样式并允许样式与内容分离的一种标记性语言[1]。利用 CSS 的定位属性可以实现不同的定位方式, 结合 DIV 等块级元素可以实现页面布局, 而且可以更灵活。本文通过举例并进行分析说明, 给出使用 CSS 的定位属性进行网页布局的使用方法和注意事项, 达到掌握的目的。

2. 使用举例

2.1. 相关概念

1) 定位(position)属性

在 CSS 样式中, position 定位属性定义元素区域的相对空间位置, 可以相对于其上级元素, 或相对于另一个元素, 或相对于浏览器窗口[2]。position 定位属性有 4 种取值, CSS 语法如下:

```
position:static|relative|absolute|fixed
```

参数包括了四种属性值: static、relative、absolute 和 fixed, 它们决定了元素区域的布局方式。

static 静态定位为默认值, 网页元素遵循 HTML 的标准定位规则, 即网页各种元素按照“前后相继”的顺序进行排列和分布[3]。

relative 相对定位, 网页元素也遵循 HTML 的标准定位规则, 但需要为网页元素相对于原始的标准位置设置一定的偏移距离, 可以使用 CSS 的属性如 left、top、right、bottom 来设置位置。

absolute 绝对定位, 位置将依据浏览器左上角的 0 点开始计算, 使用 CSS 属性 left、top、right、bottom 来设置位置。

fixed 固定定位与绝对定位类似, 是绝对定位的子类别, 相对于浏览器窗口进行定位, 当拖动浏览器窗口滚动条时, 该元素位置始终保持不变, 使用 CSS 属性 left、top、right、bottom 来设置位置[4]。

2.2. 使用举例

1) 静态定位

静态定位是 position 属性的默认值, 即该元素出现在文档的常规位置, 遵循 HTML 的标准定位规则, 即按照“前后相继”的顺序进行排列和分布, 不会重新定位。

例题 1: 完成页面, 效果如图 1。

对应的主要代码如下:

```
<!doctype html>
<html>
<head>
```

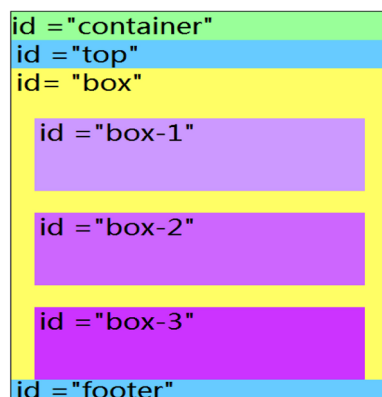


Figure 1. Web layout 1

图 1. 网页布局 1

```

<title>静态定位</title>
<style type="text/css">
#container {                               /*页面顶部容器的 CSS 规则*/
width: 405px;                               /*设置元素宽度*/
margin: 0 auto;                             /*设置元素自动居中对齐*/
font-size: 30px;                           /*设置字体大小*/
background-color: #9F9;                    /*设置背景色浅绿色*/
border: 1px #000 solid; /*设置 1px 黑色 实线边框*/
}
#top {
width: 400px;                               /*设置元素宽度*/
background-color: #6CF;                    /*设置背景色浅蓝色*/
padding-left: 5px;                         /*设置左内边距 5px, 区域总宽度为 405px*/
}
#box {
width: 400px;
background-color: #FF6; /*设置背景色黄色*/
padding-left: 5px;
position: static;           /*静态定位, 默认选项, 可以不设置*/
}
#box-1 {
width: 350px;                               /*设置元素宽度*/
height: 100px;                             /*设置元素高度*/
background-color: #C9F;                    /*设置背景色*/
padding-left: 5px;                         /*设置左内边距 5px*/
margin-left: 20px;                         /*设置左外边距 20px, 区域总宽度为 375px*/
}
#box-2 {

```

```
width:350px;
height:100px;
background-color:#C6F;
margin-left:20px;
padding-left:5px;
}
#box-3 {
width:350px;
height:100px;
background-color:#C3F;
margin-left:20px;
padding-left:5px;
}
#footer {
width: 400px;
line-height: 30px;
background-color: #6CF;
padding-left: 5px;
}
</style>
</head>
<body>
<div id="container"> id="container"
<div id="top"> id="top"</div>
<div id="box"> id="box"
<div id="box-1">
<p>id="box-1" </p>
</div>
<div id="box-2">
<p>id="box-2" </p>
</div>
<div id="box-3">
<p>id="box-3" </p>
</div>
</div>
<div id="footer">id="footer"</div>
</div>
</body>
</html>
```

分析：从上可以看到页面中分别定义了 id="top"、id="box"和 id="footer"3 个 Div 容器，彼此是并列

关系。id="box"的容器又包含 id="box-1"、id="box-2"、id="box-3"这 3 个容器，彼此也是并列关系。题中 div 元素的定位采用定位属性的默认值，即静态定位方式。静态定位的使用场景一般来说不用写，除非是想要覆盖之前设置的定位。

2) 相对定位

相对定位(position:relative;)指的是通过设置偏移量,让这个元素“相对于”它原本的位置进行移动[5]。

例题 2：在例题 1 的基础上完成页面，效果如图 2。

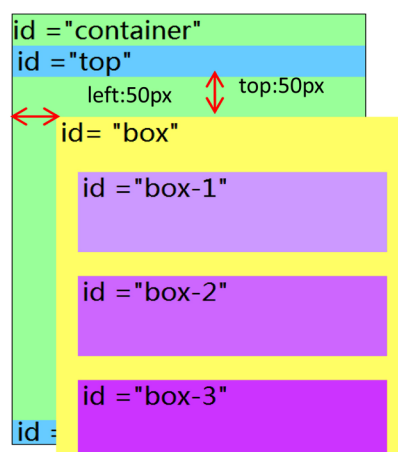


Figure 2. Web layout 2

图 2. 网页布局 2

在例题 1 的基础上，只需将 id="box"的元素的 CSS 定义修改为如下代码即可：

```
#box {
width: 400px;
background-color: #FF6;
padding-left: 5px;
position:relative;    /*设置相对定位*/
top:50px;              /*相对于例题 1 的位置，向下移动 50px*/
left:50px;             /*相对于例题 1 的位置，向右移动 50px*/
}
```

分析：本例中，id="box"的元素进行了相对定位属性设置，相对于在文档流(id="container")中的原来位置向下并且向右各移动了 50px，原来的位置不但没有让 id="footer"的元素占据，反而还将其覆盖了一部分。若此处不设置移动位置即没有 top, left 设置，则相对定位没有效果[6]。相对定位的使用场景和 absolute 一起使用，常用于 logo 的定位。

3) 绝对定位

用“position:absolute;”表示绝对定位，使用绝对定位的对象可以被放置在文档中的任何位置，位置将依据它的包含框(浏览器左上角的 0 点或已经定位的包含框)开始计算。

例题 3：在例题 1 的基础上完成页面，效果如图 3。

与例题 1 相比，主要修改 id="box-1"区域部分，代码如下：

```
#box-1 {
width:350px;           /*设置元素宽度*/
```

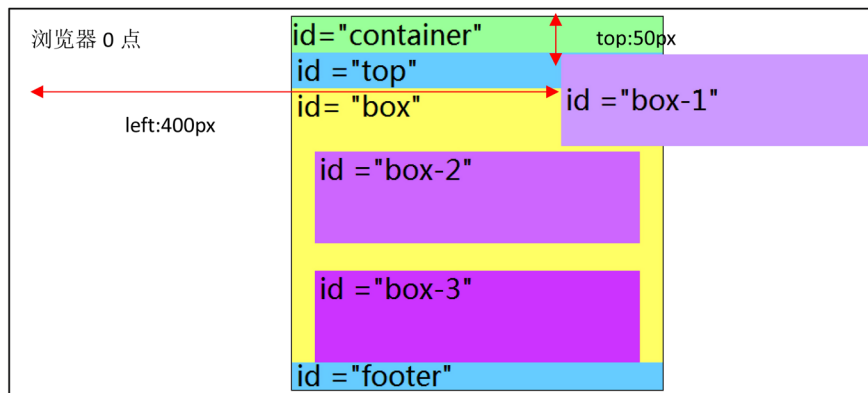


Figure 3. Web layout 3
图 3. 网页布局 3

```

height:100px;          /*设置元素高度*/
background-color:#C9F; /*设置背景色*/
padding-left:5px;      /*设置左内边距 5px*/
margin-left:20px;      /*设置左外边距 20px*/
position:absolute;   /*设置绝对定位*/
top:50px;              /*设置距顶部距离*/
left:400px;            /*设置距左边距离*/
}

```

分析：通过效果图可以看到，当 `id="box-1"` 的元素使用绝对定位后，其位置相对于浏览器 0 点向下和向右移动，页面中的其他元素位置也相应发生变化，`id="box-2"`、`id="box-3"` 和 `id="footer"` 这些元素都上移。由此可见，使用绝对定位元素的位置与文档流无关，且不占据空间，即不再遵循 HTML 的标准定位规则。文档中其他元素的布局就像绝对定位的元素不存在一样[7]。绝对定位的使用场景和 `relative` 一起使用，常用于 logo 的定位。

4) 相对定位与绝对定位的混合使用

例题 4：在例题 1 的基础上结合例题 2 和例题 3 完成页面，使得 `Id="box-1"` 的元素相对于 `Id="box"` 的元素进行绝对定位，效果如图 4。

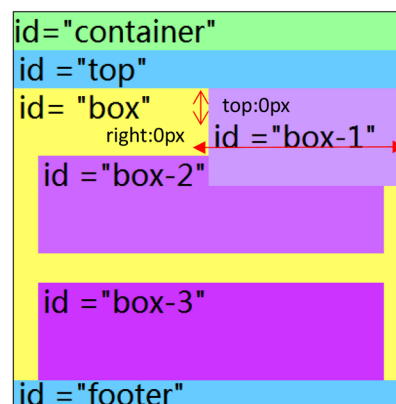


Figure 4. Web layout4
图 4. 网页布局 4

与例题 1 相比，主要修改 box，box-1 区域部分，修改部分代码如下：

```
#box {
width: 400px;
background-color: #FF6;
padding-left: 5px;
position: relative;    /*设置相对定位*/
}
#box-1 {
width: 200px;           /*设置元素宽度*/
height: 100px;         /*设置元素高度*/
background-color: #C9F; /*设置背景色*/
padding-left: 5px;     /*设置左内边距 5px*/
margin-left: 20px;     /*设置左外边距 20px*/
position: absolute;  /*设置绝对定位*/
top: 0px;              /*设置距顶部距离*/
right: 0px;            /*设置距右边距离*/
}
```

分析：相对定位与绝对定位的混合使用。首先对 Id="box" 的元素进行相对定位，使得 Id="box" 中的所有元素成为它的子元素，它的子元素将以它为基准进行定位。设置 Id="box-1" 的元素进行绝对定位，使其相对于 Id="box" 的父元素进行绝对定位。设置 Id="box-1" 元素距离 Id="box" 元素顶部距离 0 px，距离右边距离 0 px，所以 Id="box-1" 元素被置于 Id="box" 元素的右上角，实现了子元素相对于父元素进行定位。若删除 id="box" 的 position 属性，则 box1 元素相对于浏览器窗口进行绝对定位。

5) 固定定位

固定定位(position: fixed;)是绝对定位的子类别，是相对于视窗固定的，就算页面文档发生了滚动，它也会一直停留在固定的位置。

例题 5：固定定位属性的应用，效果如图 5。



Figure 5. Web layout4
图 5. 网页布局 5

对应的主要代码如下：

```
<!doctype html>
```

```
<html>
<head>
<title>固定定位示例</title>
<style type="text/css">
#tt {background-color: #39c;
width: 50%;
position: fixed; //设置固定定位属性
left:200px;
top: 200px;
}
#cc{
width:1024px;
height:768px;
background:url(Lighthouse.jpg);//设置背景图，成为固定定位元素的参照物。
}
</style>
</head>
<body>
<div id="cc">
<div id="tt"><h1>使用 CSS 固定定位</h1></div>
</div>
</body>
</html>
```

分析：将 Id="tt"的元素固定定位，id="cc"的元素的高度让浏览器出现垂直滚动条，当拖动滚动条向下滚动页面时，tt 元素固定定位，相对于浏览器位置不变，如图 5 所示。固定定位的使用场景常用于广告图片。

3. 总结

通过以上的例题和分析可以得出 CSS 的定位属性对页面布局的作用如下：

1) static 静态定位为默认值，网页元素遵循 HTML 的标准定位规则，即网页各种元素按照“前后相继”的顺序进行排列和分布，是最常用的默认定位方式。

2) relative 相对定位，网页元素也遵循 HTML 的标准定位规则，但需要为网页元素相对于原始的标准位置(即使用默认定位的位置)设置一定的偏移距离。可以使用 CSS 的属性 left、top、right、bottom 来设置位置和绝对定位结合常用于网站 logo 的定位。

3) absolute 绝对定位，网页元素不再遵循 HTML 的标准定位规则，将以该元素的上级元素为基准位置或依据浏览器窗口设置偏移量进行定位，可以使用 CSS 属性 left、top、right、bottom 来设置位置。网页元素的位置相互独立，没有影响，相对定位和绝对定位混合使用，使用绝对定位的元素相对于使用相对定位的元素进行绝对定位。

4) fixed 固定定位，相对于浏览器窗口进行定位，当拖动浏览器窗口滚动条时，该元素位置始终保持不变，可以使用 CSS 属性 left、top、right、bottom 来设置位置，常用作广告图片的定位。

5) 特别注意一定要分清每种定位方式的不同, 结合适用场景来合理地使用每种定位方式。

4. 结束语

本文通过例题对 CSS 的定位属性进行了举例和分析, 总结了使用规则, 提出了使用时的注意事项, 特别是通过结合例题对具体定位的概念进行了剖析和学习, 在网页布局中, 可以根据页面布局的需要, 选择合适的定位属性进行布局。

参考文献

- [1] 温谦. CSS 网页设计标准教程[M]. 北京: 人民邮电出版社, 2015: 84-90.
- [2] 刘瑞新. 网页设计与制作 html + css + javascript [M]. 北京: 机械工业出版社, 2013: 112-113.
- [3] 黄玮雯. 网页界面设计[M]. 北京: 人民邮电出版社, 2014: 100-103.
- [4] 庄向泽. 网页制作综合技术教程[M]. 第 2 版. 北京: 人民邮电出版社, 2015: 277-281.
- [5] 李毅. 网页制作与开发教程[M]. 北京: 人民邮电出版社, 2015: 193-196.
- [6] 修毅. 网页设计与制作[M]. 北京: 人民邮电出版社, 2015: 138-139.
- [7] 王海波. 网页设计与制作——基于计算思维[M]. 北京: 机械工业出版社, 2014: 100-105.

知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2325-2286, 即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: sea@hanspub.org