# 第四章 新增弹性模型

1.什么是弹性盒子模型

2.box-orient定义盒子布局取向

3.box-direction属性定义盒子的布局顺序

4.box-ordinal-group属性定义盒子布局位置

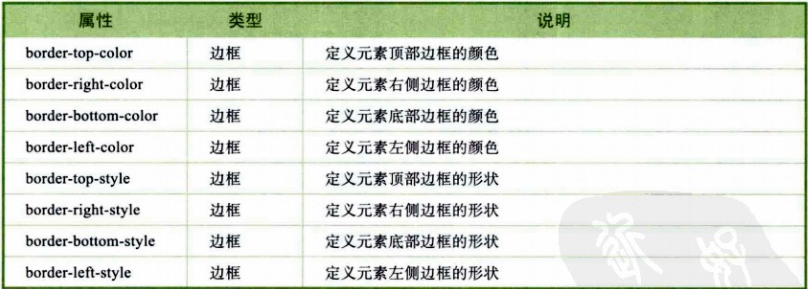
5.定义盒子的弹性空间box-flex属性

CSS3中在CSS2的基础上引入了一些新的盒子模型技术参数,提出了弹性盒子模型的概念,方便于设计师更加灵活的设计页面上各个容器的大小和位置。

CSS1定义的盒形元素



CSS2中新增的盒模型属性



1.什么是弹性盒子模型

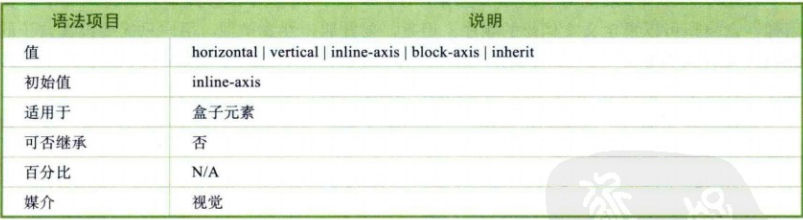
CSS3中引入了新的盒模型处理机制，即是弹性盒模型，该模型用于决定元素在盒子中的分布方式以及如何处理盒子的可用空间。主要目的是用于创建自适应浏览器窗口流动布局或自适应大小的弹性布局。

为了表现弹性盒模型的表现需要，CSS3新增了8个属性。



2.box-orient属性

box-orient属性可用于定义盒子元素内部的流动布局方向，该属性的基本语法如下。注意在使用弹性盒子模型时，用户需要先把父容器的display属性设置为box或者inline-box



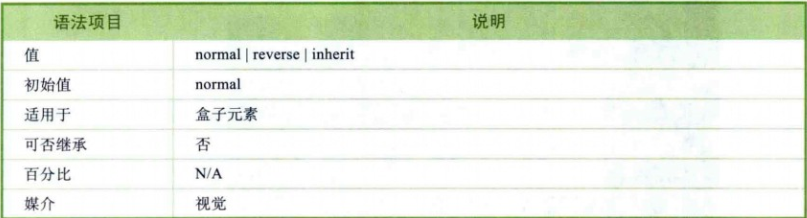
取值简单说明:

horizontal:盒子元素从左到右在一条水平线上显示它的子元素

vertical:盒子元素从上到下在一条垂直线上显示它的子元素

3.box-direction属性定义盒子的布局顺序

box-direction属性可以改变盒子元素中内部元素的流动顺序

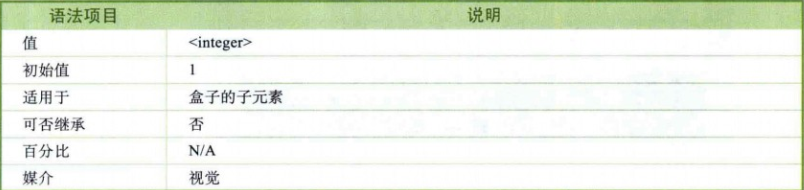


normal:正常线束顺序，即如果盒子元素的box-orient属性值为horizontal,则其包含的子元素按照从左到右的顺序显示。如果盒子元素的box-orient属性值为vertical，则其包含的子元素按照从上到下的顺序显示

reverse:反向显示，盒子所包含的子元素的显示顺序将与normal相反。

## 4.box-ordinal-group属性定义盒子布局位置

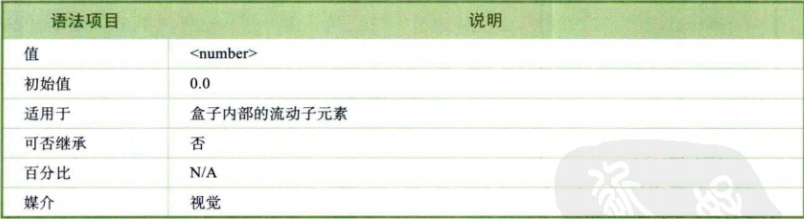
box-ordinal-group属性可以改变盒子内部元素的流动顺序，而box-ordinal-group属性能够设置每个子元素在盒子中的具体显示位置。



box-ordinal-group（）可以设置指定的数值，让元素在规定的位置出现，其序号默认都为1，并且序号相同的元素将按照他们在文档中加载的顺序进行排列。

**注意:值更低的元素会显示在值更高的元素前面显示。盒子内部的元素默认的box-ordinal-group是1**

## 5.定义盒子的弹性空间box-flex属性

****

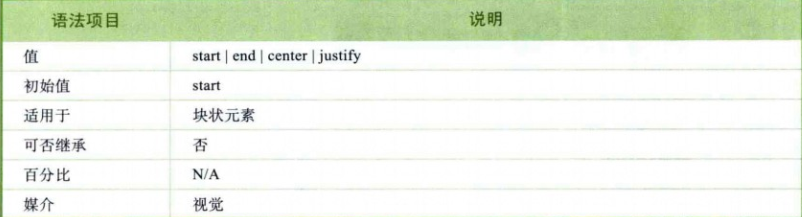
number属性值是一个整数或者小数，当盒子中包含多个定义了box-flex属性子元素时，浏览器将会把这些子元素的box-flex属性相加，然后根据它们各自的值占总值的比例来分配盒子剩余的空间。

**公式:子元素的尺寸=父盒子的尺寸\*子元素的box-flex属性值/所有子元素的box-flex属性值的和**

**练习1：在不使用js的情况下：三列布局：中间固定宽度，两侧屏幕自适应。**

## 6.管理盒子的空间box-pack和box-align属性

**box-pack属性基本语法**

****

取值简单说明

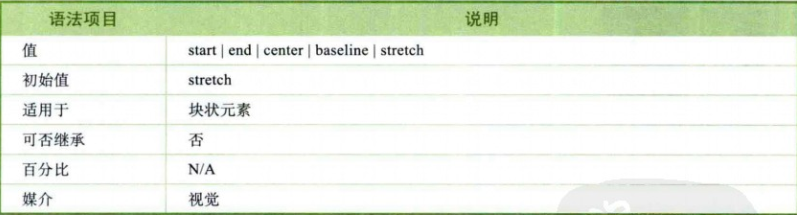
start:所有子元素都显示在盒子的左侧，富余的空间显示在盒子的右侧

end:所有子元素都显示在盒子的右侧，富余的空间在盒子的左侧。

justify:富余空间在子元素之间平均分配。在第一个子元素之前和最后一个子元素之后不分配空间

center:富余的空间在盒子的两侧平均分配

**box-align属性基本语法**

****

取值简单说明:

start:所有子元素沿盒子上边缘排列，都显示在盒子的上部，富余空间显示在盒子的底部。

end:所有子元素沿盒子的下边缘排列，都显示在盒子的底部，富余的空间显示在盒子的上部。

center:富余空间在盒子的上下两侧平均分配。

**练习2：在不使用js的情况下:实现盒子水平和垂直居中**

## 7.盒子模型的阴影

box-shadow属性向框添加一个或者多个阴影

|  |  |
| --- | --- |
| 值 | 描述 |
| h-shadow | 必需。水平阴影位置。 |
| v-shadow | 必需。垂直阴影的位置。 |
| blur | 可选。模糊距离 |
| spread | 可选。阴影尺寸 |
| color | 可选。阴影颜色 |
| inset | 可选。将外部阴影改为内部阴影 |

## 其它新增盒模型的属性

### 8.1） box-reflect倒影

由于box-reflect不是w3c组织标准属性，所以要添加浏览器的私有属性。

**使用方法:**

-webkit-box-reflect:none|<direction><offset>?<mask-box-image>

|  |  |
| --- | --- |
| 属性值 | 描述 |
| none | box-reflect默认值，表示无倒影效果 |
| above | 生成的倒影在对象的上方 |
| below | 生成的倒影在对象的下方 |
| left | 生成的倒影在对象的左侧 |
| right | 生成的倒影在对象的右侧 |

从box-reflect语法中可以得知，其中包括以下几个属性值

* none:此值为box-reflect的默认值，表示无倒影
* direction:此值表示box-reflect生成倒影的方向，主要包括以下几个值
* above:表示生成的倒影（原图）上方
* below:表示生成的倒影（原图）下方
* left:表示生成的倒影（原图）左侧
* right:表示生成的倒影（原图）右侧
* offset:用来设置倒影与对象（原图）之间的距离，其取值可以是固定像素值，也可以是百分比。
* mask-box-image:用来设置倒影的遮盖效果，可以是背景图片，也可以是渐变生成的背景图像。

练习3：模糊水平倒影图片

### 8.2） 诡异的盒子模型 box-sizing

在css2所定义的盒子模型，为标准的盒子模型(content-box)

**content-box定义:**

width/height=border+padding+content

**在css3中新增了一个怪异盒模型的概念(border-box)**

width/height = content

**注：标准盒子模型主要适用于pc端。诡异(border-box)主要适用于了移动端。**

### 8.3） 分栏布局

在css3提供了类似css2中table布局的方式，那就是分栏布局。分栏布局指的是，均分每个容器，每列宽度都一样。分栏布局主要三个参数实现:

* column-count:分列数
* column-width:分列数的宽度
* column-gap:栏间间隙，默认为0
* column-rule:栏间线条，默认为0

### 8.4） CSS3中的响应式布局

在css3中为我们提供了@media查询。此属性是针对于不同媒体来定义不同的样式。

@media可以针对不同的屏幕尺寸设置不同不同的样式，特别是需要你设置响应式的页面，@medai是非常有用的。当你重置浏览器大小的时候，页面也会根据浏览器的宽度和高度重新渲染页面。

**基本用法:**

**1）可以针对于style样式**

**@media screen and|not|only(media feature)**

1. **可以针对与不同的stylesheets**

<link rel=”stylesheet” media=”mediatype and | not|only(media feature)” href=”my.css”/>

@media only screen and (max-width: 500px) {  
    .gridmenu {  
        width:100%;  
    }  
  
    .gridmain {  
        width:100%;  
    }  
  
    .gridright {  
        width:100%;  
    }  
}

作业:

