HTML5新特性

HTML5+CSS3（高级）学习笔记

[一、HTML5新增的语义化标签 2](#_Toc36482703)

[1）视频<video>（一般采用mp4格式的视频） 2](#_Toc36482704)

[2）音频（audio） 3](#_Toc36482705)

[3）一个小的多媒体标签的总结 3](#_Toc36482706)

[三、HTML5的新增的input表单 4](#_Toc36482707)

[2）新增的表单属性 4](#_Toc36482708)

[四、新增的CSS3属性 5](#_Toc36482709)

[1）属性选择器---可以根据元素特定的属性来选择元素 5](#_Toc36482710)

[2）结构伪类选择器—主要根据文档结构来选择元素，常用于根据父级选择里面的子元素 5](#_Toc36482711)

[3）重点掌握以下nth-child（） 5](#_Toc36482712)

[4）伪元素选择器—帮助我们利用css创建新标签元素，从而简化html结构 6](#_Toc36482713)

[5）通过伪元素来清除浮动 7](#_Toc36482714)

[6）双伪元素清除浮动 7](#_Toc36482715)

[7））CSS3新增的盒子模型属性 7](#_Toc36482716)

[8）CSS3滤镜filter—设置模糊或者颜色偏移 8](#_Toc36482717)

[9）calc（）函数---实现简单计算 8](#_Toc36482718)

[10）CSS3实现过渡效果（重点） 8](#_Toc36482719)

[五、2D转换---可以实现元素的位移、旋转、缩放等效果（translate、rotate、scale） 9](#_Toc36482720)

[1）2D转换之移动（translate），可以改变元素在页面中的位置 9](#_Toc36482721)

[2）2D转换之旋转（rotate）--使元素在2维平面内顺时针旋转或逆时针旋转 10](#_Toc36482722)

[3）设置旋转中心点（transform-origin） 10](#_Toc36482723)

[4）2D转换之缩放（scale） 11](#_Toc36482724)

[5）2D转换综合写法 11](#_Toc36482725)

[六、css动画 11](#_Toc36482726)

[1）动画的基本使用（先定义再使用） 11](#_Toc36482727)

[六、BootStrap前段开发框架 11](#_Toc36482728)

[1）bootstrap的使用 11](#_Toc36482729)

[2)内容书写 12](#_Toc36482730)

[七、BootStrap布局容器 13](#_Toc36482731)

[1·）container类---（class=‘container’）使用，属于响应式布局容器，固定宽度 13](#_Toc36482732)

[2）container-fluid类---流式布局容器，百分百宽度 13](#_Toc36482733)

[八、BootStrap栅格系统 13](#_Toc36482734)

[1）简介 13](#_Toc36482735)

## 一、HTML5新增的语义化标签

▲<header>:头部标签

▲<nav>:导航标签

▲<artical>:内容标签

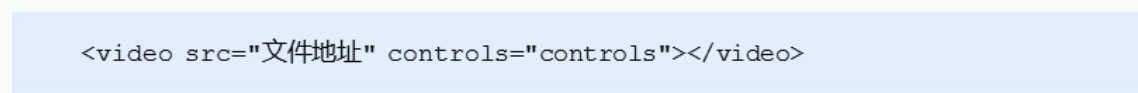
▲<section>:定义文档某个区域

▲<aside>:侧边栏标签

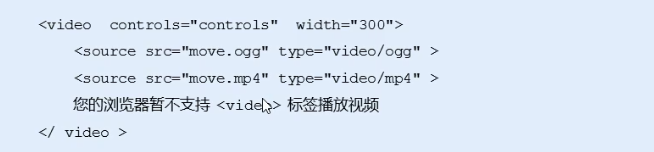
▲<footer>:尾部标签

**二、HTML5新增的多媒体标签（谷歌浏览器默认禁用自动打开，需设置）**

### 1）视频<video>（一般采用mp4格式的视频）



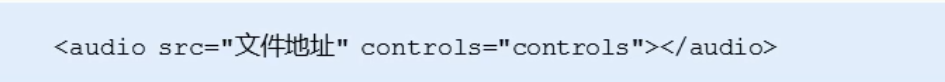
如需考虑兼容性，则如下



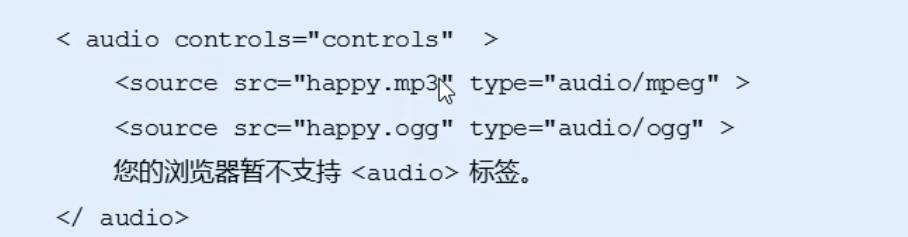
常见属性



### 2）音频（audio）



若需考虑兼容性，则



常见属性



### 3）一个小的多媒体标签的总结

▲谷歌浏览器禁用了音视频的自动播放功能

▲对于视频可通过添加muted属性来静音自动播放，但是音频（可以通过JavaScript解决）

▲视频视频标签是重点，我们常设置自动播放，不使用controls控件，循环和设置大小属性

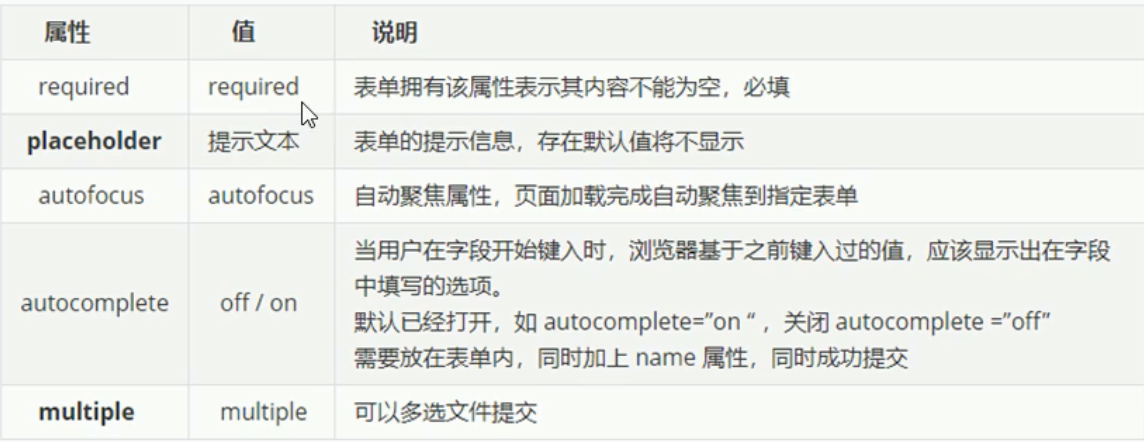
▲视频常见格式（mp4，webm，ogg）

▲音频常见格式（mp3，wav，ogg）

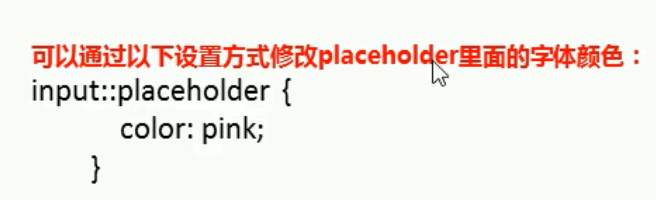
## 三、HTML5的新增的input表单



### 2）新增的表单属性



注，伪类



## 四、新增的CSS3属性

### 1）属性选择器---可以根据元素特定的属性来选择元素

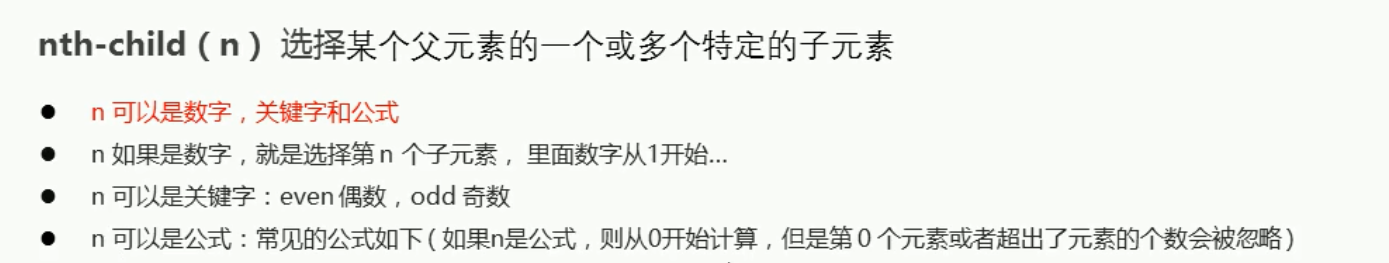


☆类选择器，属性选择器，伪类选择器的权重均 为10

### 2）结构伪类选择器—主要根据文档结构来选择元素，常用于根据父级选择里面的子元素



### 3）重点掌握以下nth-child（）



▲例子1（odd为奇数，even为偶数）

        ul li:nth-child(odd) {

            background-color: #999;

        }

▲例子2（可以采用公式，但是变量必须是n）

        ol li:nth-child(2n) {

            font-size: 20px;

        }

nth-child(n)---选择某个父元素中的一个或多个特定的元素

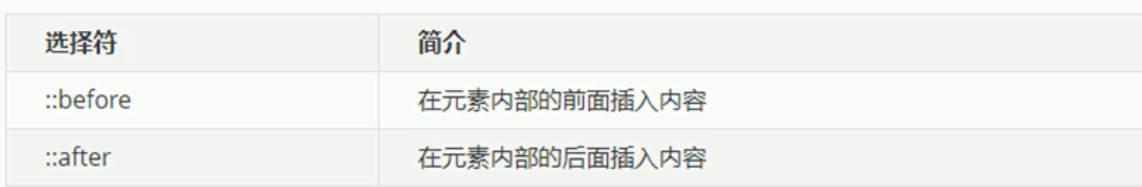


♣nth-child(n)和nth-of-type(n)两者的区别

①nth-child(n)是对父元素里面所有的孩子排序选择（序号是固定的），先找到第n个孩子，然后看看是否和类型E匹配；

②nth-of-type(n)是对父元素里面指定子元素进行排序选择。先去匹配类型E，然后再根据E找第n个孩子

### 4）伪元素选择器—帮助我们利用css创建新标签元素，从而简化html结构



注意 △before和after创建的元素 属于行内元素；

△新创建的这个元素在文档树中是找不到的，所以被称为伪元素；

△实现可参考div：：befor（）；

△需要重点注意的是，before和after中必须要有content属性；（类似于盒子，里面写内容）；

△befor是在父元素父元素内容的前面创建元素；after是在父元素内容的后面插入元素；

△伪元素选择器和标签选择器一样，权重为1

举例↓↓↓

        div::before {

            content: 'hello,我';

            color: yellow;

        }

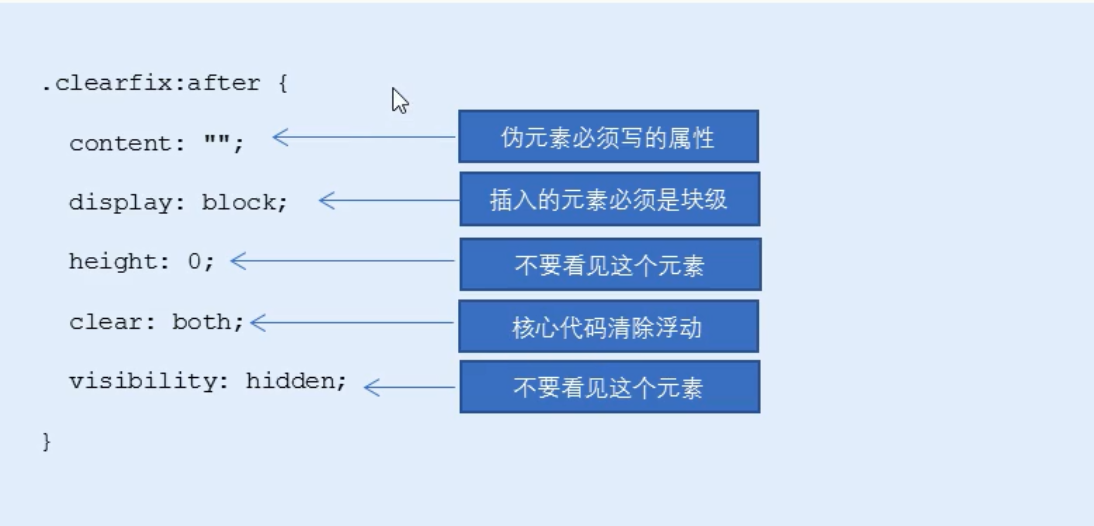
        div::after {

            content: '啥子哦';

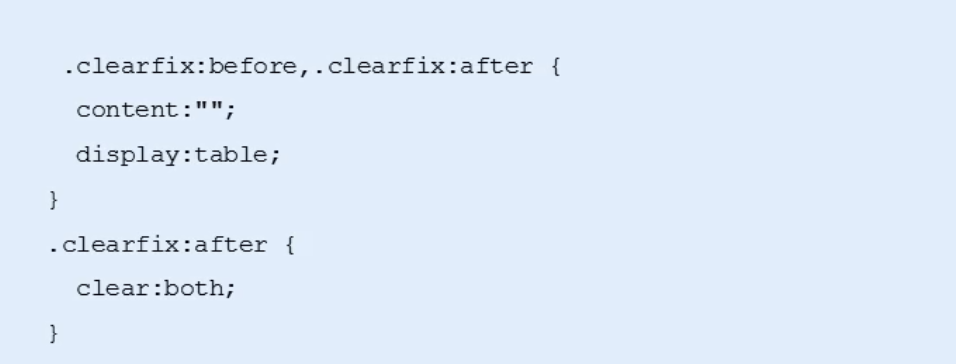
            color: red;

        }

### 5）通过伪元素来清除浮动



### 6）双伪元素清除浮动



### 7））CSS3新增的盒子模型属性

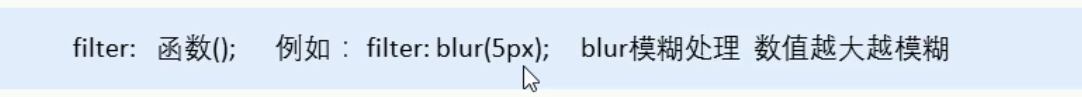
▲可通过box-sizing来指定盒子模型。分为content-box、border-box。（使计算盒子大小的方式改变）

▲box-sizing：content-box盒子大小为width+padding+border（默认类型）

▲box-sizing：border-box盒子大小仅为width

：若盒子模型改为box-sizing：border-box，那padding和border就不会撑开盒子（前提是padding和border不会超过width宽度）

### 8）CSS3滤镜filter—设置模糊或者颜色偏移



例如

        img {

            filter: blur(15px);

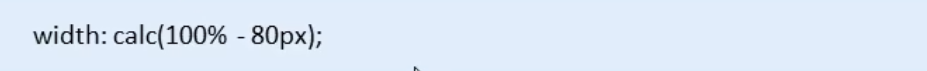
        }

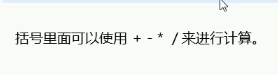
        img:hover {

            filter: blur(0);

        }

### 9）calc（）函数---实现简单计算



运算符两边必须有括号！！！

举例

        .box2 {

            width: calc(100% - 50px);

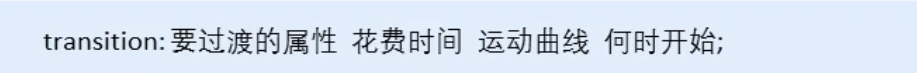
            height: 100px;

            background-color: yellow;

        }

### 10）CSS3实现过渡效果（重点）

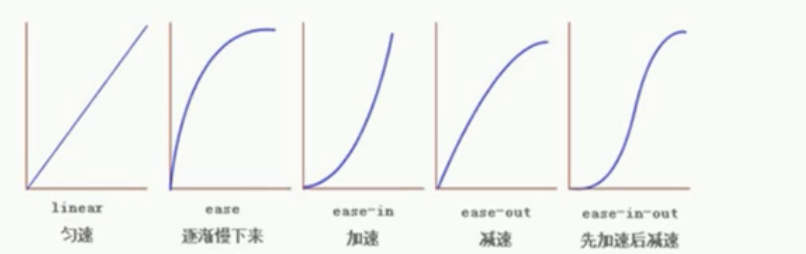
---是属性从一个状态逐渐过渡到另外的状态，常与：hover搭配使用



▲属性：想要变化的CSS属性，宽高内外边距，背景颜色均可。若想要所有的属性都变化过渡，写all即可

▲花费时间：单位是秒（必写），eg 0.5s

▲运动曲线：默认是ease（可省略）



▲何时开始：单位是秒，可以设置延迟触发的时间，默认为0s（可省略）

▲过渡使用口诀：：谁过渡就给谁加

.table .tablepart1 {

    float: left;

    width: 234px;

    height: 614px;

    ▲transition: all .3s;

}

.table .tablepart1:hover {

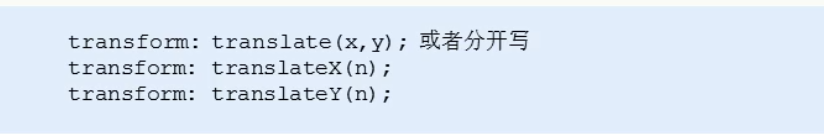
    position: relative;

    top: -5px;

    box-shadow: 5px 12px 10px 5px rgba(0, 0, 0, .3);

## 五、2D转换---可以实现元素的位移、旋转、缩放等效果（translate、rotate、scale）

### 1）2D转换之移动（translate），可以改变元素在页面中的位置

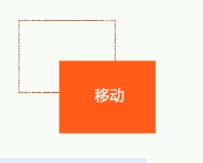


▲定义2D转换中的移动，沿着X和Y轴移动元素；

▲translate最大的优点：不会影响到其他元素的位置；

▲translate中的百分比单位是相对于自身元素的translate：（50%，50%）

▲对行内标签无效；



▲可以根据其百分比特性来实现盒子的水平居中

        .small {

            position: absolute;

            top: 50%;

            left: 50%;

            transform: translate(-50%, -50%);

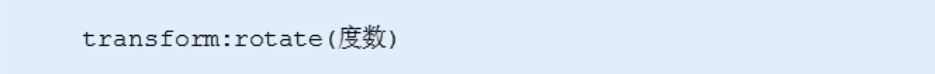
            width: 200px;

            height: 200px;

            background-color: #333;

        }

### 2）2D转换之旋转（rotate）--使元素在2维平面内顺时针旋转或逆时针旋转

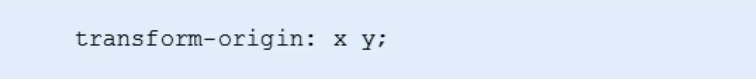


▲rotate里面跟度数，单位是deg，比如rotate（deg）；

▲角度为正时，顺时针转；角度为负时，逆时针转；

▲默认旋转中心是元素的中心点；

### 3）设置旋转中心点（transform-origin）



▲注意参数x和y是用空格隔开；

▲x和y默认转换的中心点是元素的中心点（50% 50%）等价于（center center）

▲x和y的属性值可以是像素也可以是方位名词（top bottom left right center）

        img {

            transition: all .5s;

            transform-origin: right bottom;

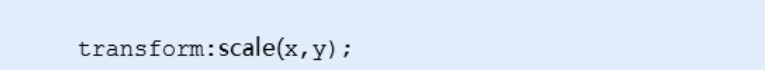
        }

        img:hover {

            transform: rotate(360deg);

        }

### 4）2D转换之缩放（scale）



▲x和y是用逗号分隔；

▲transform：scale（1，1）：宽和高都放大一倍，相当于不变；

▲transform：scale（2，2）：宽和高均放大2倍；

▲transform：scale（2）：若只写一个参数，则第二个参数和第一个参数一样；

▲transfor：scale（0.5，0.5）：是指宽高均缩小一倍；

▲sacle缩放的优势：不会影响其他的盒子；可以自由设置缩放中心点，默认是以中心缩放；

### 5）2D转换综合写法

▲顺序会影响转换的效果（先旋转会改变坐标轴的方向）

▲当我们同时有位移和其他属性时，一定要将位移放到最前！！

        div:hover {

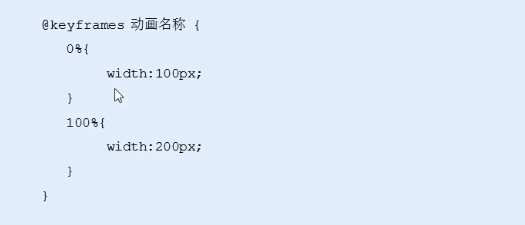
            transform: translate(150px, 50px) rotate(360deg) scale(1.5);

        }

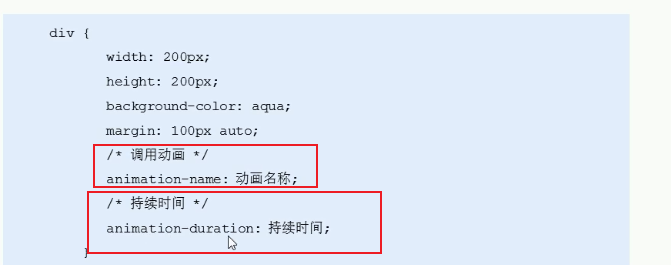
## 六、css动画

### 1）动画的基本使用（先定义再使用）

1）用keyframes定义动画（类似定义类选择器）



2）动画的使用



3）动画序列

①0%是动画的开始，100%是动画的结束，这样的规则就是动画序列；

②在@keyframe中规定某项css样式，就可以创建由当前样式转变为新样式的效果；

③动画是使元素从一种样式逐渐转化为另一种样式的效果。可以改变任意多的样式的任意多的次数；

④用百分比规则表示变化发生的时间：0%、100%或者是form、to

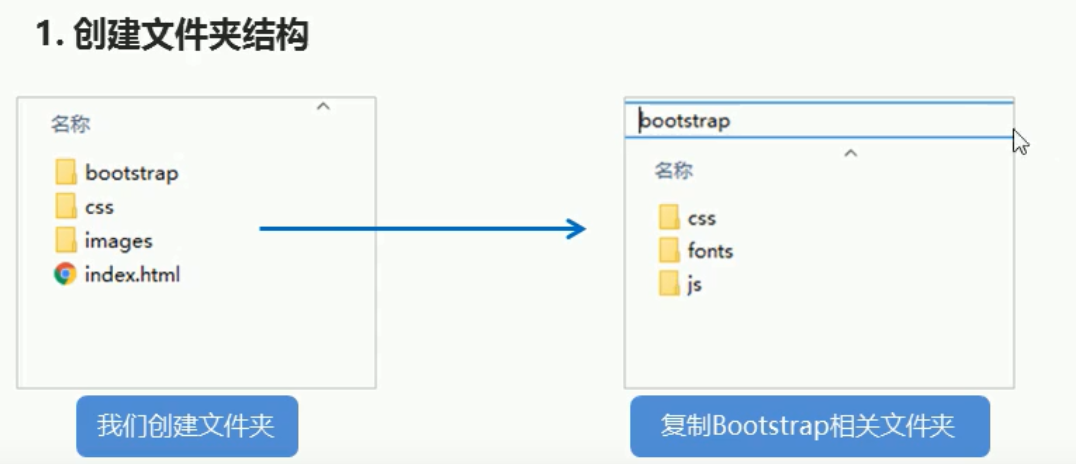
## 六、BootStrap前段开发框架

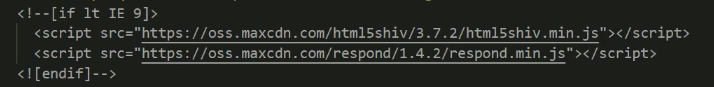
### 1）bootstrap的使用

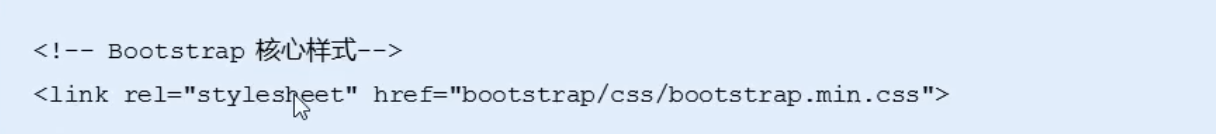
▲控制权在框架本省，使用者要按照框架规定的某种规范进行开发；

▲步骤：1、创建文件夹结构；2、创建html骨架结构；

3、引入引入相关样式文件4、书写内容；







### 2)内容书写

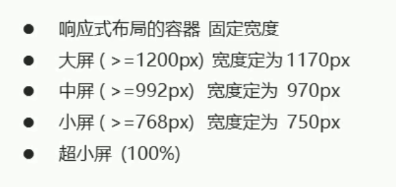
▲可以直接用BootStrap预先定义好的样式来使用;

▲可以自主修改BootStrap原先的样式,但需要注意权重的问题;

▲学好BootStrap的关键在于知道它定义了哪些样式,以及这些样式能实现什么效果;

## 七、BootStrap布局容器

### 1·）container类---（class=‘container’）使用，属于响应式布局容器，固定宽度

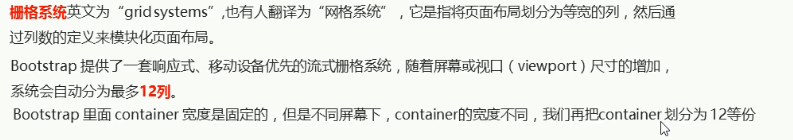


### 2）container-fluid类---流式布局容器，百分百宽度

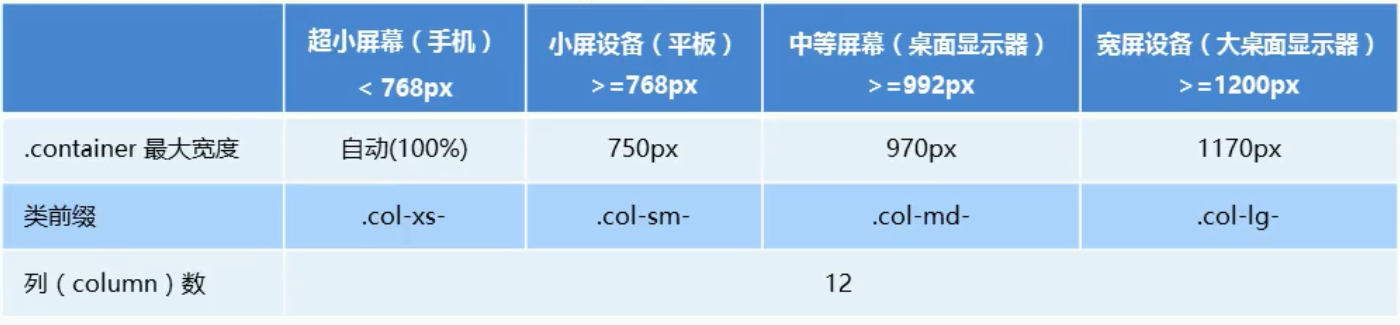


## 八、BootStrap栅格系统

### 1）简介

**2）栅格系统选项参数**

▲栅格系统通过一系列的行（row）和列（column）的组合来创建页面布局。



△行（（row）必须放在container布局容器之中；

△我们实现列的平均划分需要给列添加类前缀；

△xs-extra small：超小；sm-small：小；md-medium：中等；lg-large：大；

△如果列（column）大于·12，则多余的列所在的元素将被作为一个整体另起一行排列；

△每一列默认有左右15像素的padding

△可以同时为一列指定多个设备的类名，以便划分为不同份数，例如class=‘col-md-4 col-sm-6’