标签对<双标签>

<html></html>

<开始标签>内容</闭合标签>

<link/>

<单标签>

页面基本结构：

<!doctype html> --文档声明

<html> --文档的开始

<head> --页面头部信息

<title> </title> --页面的标题

<meta charset=”UTF-8”>

</head>

<body> --页面内容

</body>

</html> --文档的结束

<div>内容</div> --div块标签

1. div如果不设置宽度，就会占满父级的宽度
2. div如果不设置高度，高度就会被内容撑开

id=”idName”

命名：（1）同一个页面id 不能重命 （2）不能以数字命名 （3）不要用中文 （4）尽量可以不要使用特殊符号（\_ - 除外）（5）建议带有含义：

添加样式的方式以及优缺点：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 行内样式 | 内部样式 | 外部样式 | @import |
| 方式 | 标签内部的style属性中引入 | 页面内部专门的style标签中 | 写在专门的css文件中，用link引入页面 | 在样式表中重新引入一个新的样式表 |
| 优点 | 优先级高 | 1.单个页面中的通用性，没有其他页面的样式，废代码少2.代码可读性高 | 通用性极好，方便后期多个页面的修改维护 |  |
| 缺点 | 1.不符合行为、结构的三者分离的标准 2.完全没有复用的可能  3.代码可读性极差 4.不利于后期的维护和扩展 | 1.多个页面不通用 2.不利于后期维护（多个页面进行内部修改） | 1.增加http请求 2.多个页面样式写在一起时，对单个页面来说会有大量的废代码3.用户进入第一个页面，需耗费大量的时间，再进入其他页面会有更快的加载，因为页面已缓存  4.多人协作时，容易代码冲突 | 1.css本身不存在文件之间的依赖关系，所以没有使用import的必要（sass和less这些预编译的除外）2.增加多余的http请求 3.@import是同步请求，会延迟加载时间 |

标签关系：

1. 并列关系（兄弟关系，同级关系）
2. 嵌套关系（父子级关系）

文件路径：

1. 绝对路径 线上绝对路径 本地绝对路径（禁用）
2. 相对路径 相当于当前文件的一个路径（../ 返回上级目录）

添加背景图：background-image: url(图片地址）；repeat-x,repeat-y,no-repeat

设置背景图是否固定于屏幕：background-attachment

--fixed固定在屏幕 --scroll跟随滚动条滚动

（border-width,border-style,border-color）多值写法：

一值（上下左右） 二值（上下）（左右）

三值（上）（左右）（下）四值（上）（右）（下）（左）

font-size 字体大小 font-family 文字字体（首选字体，备选字体，……）

font-weight 文字加粗（normal正常/bold 加粗）

font-style 文字倾斜（italic倾斜/normal正常）

line-height 行高

font: font-style font-weight font-size/line-height font-family;

text-align 文本对齐方式

text-decoration 文本修饰 underline默认值 overline 下划线 line-through 删除线

text-indent 首行缩进

word-spacing 单词间距

letter-spacing 字母间距

white-space 强制不换行 normal pre nowrap pre-wrap pre-line

word-break: break-all; 词内断行（强制单词和数字换行）

padding(内填充)

padding多值写法

一值（上下左右）二值（上下）（左右）

三值（上）（左右）（下）四值（上）（右）（下）（左）

margin（外边距）

一值（上下左右）二值（上下）（左右）

三值（上）（左右）（下）四值（上）（右）（下）（左）

可视高 = 边框+padding+content;

a标签默认不支持宽高

a标签默认有一条下划线

a href 页面地址、文件地址、id名字(锚点) target \_blank新窗口 \_self当前窗口

伪类

:link 未被访问

:visited 已被访问

:hover 鼠标移入

:active 鼠标按下

span 不支持宽高

<ul></ul> 无序列表

<ol></ol> 有序列表

<li></li>

list-style (none)列表样式

<em>倾斜

<strong>加粗

<mark>标记

<dl>

<dt>定义列表的列表项</dt>

<dd>对于dt的描述</dd>

</dl>

<header>页面头部

<main>页面主体

<footer>页面底部

<nav>导航

<article>独立的一篇文章，回帖，新闻等

<aside>作为article附属信息（或整个页面）

<p>段落

<section>板块

标签语义化

1. 要求在写html文档时，根据每块内容的不同含义，去选择具有正确语义的标签
2. 方便机器读懂页面内容
3. 提升代码可读性，添加样式时可以方便区分控制

CSS selector（css选择器，选择符）页面上所有样式

1. id选择器

id=”idName”

#idName{具体样式}(同一个页面内不能重名)

1. \*通配符

\*{具体样式} 基础选择器

1. class选择器（类选择器）

class=”className”

.className{具体样式}

1. 类型选择器（标签选择器）

找到某个标签 tagName(标签名字){具体样式}

高级选择器

包含选择器（后代选择器） 父级 子级{具体样式}（写包含选择器时，找到这个元素，不能影响其他地方）

群组选择器 选择器1，选择器2，选择器3 {具体样式}

选择器优先级一致情况下，后边覆盖前边

选择器优先级：

当使用不同选择器，给元素加样式

类型选择 > 通配符， class选择器 > 类型选择， id选择器 > class选择器，

style > id > class > tag > \*

包含选择器的优先级是一种叠加的关系

群组选择器的优先级不会累加

清除默认样式

body, h1, h2, h3, h4, h5, h6, p, dl, dd, form {margin: 0;}

ul, ol {margin: 0; padding: 0; list-style: none;}

a {text-decoration: none;}

strong {font-weight: normal;}

em {font-style: normal;}

mark {background: none;}

img {border: none; vertical-align: top:}

标签嵌套规范化

1. a标签中不能包含a标签
2. 内嵌标签中，不能包含块标签
3. 部分块标签中，也不能包含块（h1-h6.dt带有标题性质的标签，p标签）
4. ul,ol和li,以及dl和dt,dd标签是固定嵌套关系，不能更改

div块：1.独占一行2.支持所有的样式3.不设置宽度撑满整行

span内嵌： 1.可以在一行显示2.不支持宽高以及上下margin，上下padding也有问题

1. 宽度由内容撑开 4.代码换行产生空格 5.会被字体和字体大小影响

块元素：div figure header footer main section nav article aside p h1 h2 h3 h4 h5 h6 ul ol li dl dt dd

内嵌元素： span a time mark em strong

display block 块 显示为块

inline 内嵌（行内）显示为内嵌

img 只设置宽度或只设置高度，图片会等比缩放

img 同时设置宽高，可能会拉伸图片造成变形

img 1.支持宽高 2.在一行显示 3.代码换行产生空格

inline-block(内联块) （在IE6,7下,不支持给块标签设置inline-block）

1. 支持宽高 2.在一行显示 3.代码换行，解析成空格 4.不设置宽度，宽度由内容撑开

inline-block类型的元素下边会有一个空隙

vertical-align: top; 垂直对齐方式

在表格以外，规定行内（或行内元素）相互之间的对齐方式

sub 下标 sup 上标

在IE10之前浏览器中，a标签包含图片，图片上会多出一个边框

cursor 鼠标指针 url 自定义光标（1）图片格式 .cur（2）必须添加备选方案

text 文本 pointer 一只手

xhtml的三种不同文档声明：

严格型 <!DOCTYPE html PUBLIC “-//w3c//DTD XHTML 1.0 Strict//EN” “http://w3.org/TR/xhtml/DTD/Xhtml-strict.dtd”>

过渡型 <!DOCTYPE html PUBLIC “-//w3c//DTD XHTML 1.0 Transitional//EN” “http://w3.org/TR/xhtml/DTD/Xhtml-transitionalt.dtd”>

框架型 <!DOCTYPE html PUBLIC “-//w3c//DTD XHTML 1.0 Frameset//EN” “http://w3.org/TR/xhtml/DTD/Xhtml-frameset.dtd”>

设置样式表编码 background-position 高级浏览器下有两种解释

基本样式 background-position-x background-position-y 写值时先写x再写y

float:left right none;浮动

1. 支持宽高 2.在一行显示 3.宽度由内容撑开 4.会按照指定的方向移动

脱离文档流（元素在排列的时候，不再占有位置）

元素浮动之后，撑不开父级的高度，清浮动或设置高度

clear: left/right/both/none;

1. 如果在元素的前面有对应的浮动元素，元素会在下一行显示
2. 要加在具有块属性的元素上
3. 只对结构里在它前面的元素起效果

<br> 换行

元素浮动之后，会脱离文档流，脱离了文档流之后浮动元素就撑不开父级

行为、样式、结构三者分离

清浮动 1.clear 2.可以使浮动元素撑开父级的高度

1. 在浮动元素最下边添加<div class=”clearFix></div>

.clearFix {clear: both;}

1. 在浮动元素的最下边添加 <br clear=”all”>
2. 给浮动元素的父级加 class=”clearFix”

.clearFix:after {content: “”; display: block; clear: both;}

1. 给浮动元素的父级触发bfc

block Formatting content 块级格式化上下文

BFC: 布局环境规定改环境内盒子的排列，使环境内的元素不会受到环境外的元素的影响，也不会让环境内的元素影响到外边

1. 包含浮动元素（清浮动）
2. 阻止margin 传递
3. 不被浮动元素覆盖

BFC触发条件：

float的值不为none overflow的值不为visible display的值为inline-block、table-cell、table-caption position的值为absolute或fixed

overflow内容溢出

visible 可见的(默认值) hidden隐蔽 auto超出之后显示滚动条

scroll 显示滚动条

max-width min-width max-width max-height

相对定位

position: relative;

1. 元素添加了定位之后，就可以使用以下几条样式

left right top bottom z-index

1. 相对于自己的初始位置来计算坐标
2. 不脱离文档流（元素移走之后，原始位置会被保留）
3. 提升层级

绝对定位

position: absolute;

1. 脱离文档流
2. 提升层级
3. 触发bfc
4. 根据自己有定位的父级，来计算自己的坐标（如果所有的父级没有定位，就用document来计算坐标）百分比 width/height 100%和父级一样 html的父级是document document的高度和可视区域一致
5. 是元素支持宽高
6. 不设置宽度，由内容撑开宽度

默认情况下，元素定位后，后边层级高于前边

z-index层级 1.数值越大，层级越高 2.层级只能给定位元素设置 3.层级要在同级元素之间比较

resize 大小重置 vertical垂直 horizontal 水平 none inherit 需要配合overflow:auto;

Opacity(0, 1) 0完全透明 1完全不透明

<table></table> 表格（display:table）

<caption></caption> 表格标题（display:table-caption）

<thead></thead> 表头(display:table-header-group)

<tr></tr> 行（display:table-row）

<td></td> 单元格（display:table-cell）

<th></th> 单元格(加粗、居中）（display:table-cell）

<tbody></tbody> 表格主体 （display:table-row-group）

<tfoot></tfoot> 表格底部 （display:table-row-group）

colspan 横向合并单元格 rowspan 纵向合并单元格

cellpadding 单元格填充 cellspacing 单元格间距

表格的每行每列都是对齐的

给单元格设置高度，一横行的高度都会改变

给单元格设置宽度，一竖行的宽度都会改变

thead tbody tr 没有margin和padding

td th 没有marign

border-collapse: collapse 边框合并

table {border-collapse: collapse; border-spacing: 0;}

td, th {padding:0;}

单元格中内容默认垂直居中，如果想要修改位置，可以给单元格设置vertical-align

<from></from>表单 action提交跳转地址

input 表单控件

type=”text”文本输入框 type=”password”密码框 type=”submit”提交按钮 type=”radio”单选框 type=”checkbox”复选框 type=”button” 按钮 type=”reset” 重置按钮

value是 input的值 name 表单控件的数据名称，如果这条数据提交就必须添加name

label 辅助 input label for 属性里写的是关联的input的id

<select></select>下拉菜单

option是select的子项 readonly 只读

<textarea></textarea>文本域

<button></button>按钮 selected disabled 禁止编辑和提交

单复选 checked option默认选中

:focus表单控件获得焦点时时候

块：from option

内联块：select textarea

内联：label

form {margin: 0;} //在IE6,7下有margin

input {margin: 0; padding: 0;}//根据设计图重置边框

button {padding: 0;}//根据设计图重置边框

select {padding: 0;}//根据设计图重置边框

input, button, textarea {padding: 0;}

textarea {padding: 0; resize:none; overflow: auto;}

IE浏览器兼容问题

1. 在IE6下，当高度小于19px的时候，高度就会当作19px来处理

解决方法：给元素加overflow:hidden;

1. 在IE6下，1px的dotted(点线)边框会被显示成虚线

解决方法：用背景图代替

1. 在chrome下，字体小于12px的文字会按照12px显示

解决方法：1.用css3缩放 2.以图换字

1. 在IE6下，如果元素中既包含文字又包含inline-block 类型的标签，元素的行高就会消失 解决方法：把文字单独包起来，单独设置行高
2. 在IE6,7下，给元素加边框，非但不能阻止元素的margin向外消失，还会使子元素的margin消失 解决方法：触发元素的haslayout

haslayout:当haslayout设置为true时，根据当前元素的父级或者元素的自身内容重新计算当前元素的宽高尺寸

haslayout 触发条件:

display: inline-block height: (任何值除了auto)

float: left/right width: (任何值除了auto) writing-mode:tb-rl zoom:(除normal外任意值)

1. 在IE6下，不设置文档声明，浏览器会进行怪异盒模型解析

可视高 = content + padding + border;

怪异盒模型 可视宽/高 = width/height;

content = width/height + padding + content;

标准盒模型 可视宽/高 = width/height + padding + border;

content = width/height;

1. 在IE8及以前的浏览器，html5新增标签不识别

解决方法：document.createElement(‘标签的名字’) 在文档中创建一个标签，默认没有display 注意：在IE浏览器中，不支持main 标签，需要设置main标签

h5标签：header nav footer main article aside time mark figure figcaption

1. IE7及之前的浏览器都不支持（l-v-h-a）之外的伪类

解决方法：用js解决

1. IE6之前的浏览器，只支持a标签的伪类
2. 在IE6,7下，不支持把标签设置成inline-block
3. 在IE6,7下，不允许把其他类型的元素转化成表格属性
4. 在IE6下，块元素有浮动和横向的margin,横向的margin值会被放大两倍

解决方法：转成内嵌

1. 在IE6,7下，一行元素的宽度之和和父级的宽度相差大于等于3px，最后一行的下margin失效 解决方法：无解（不要用下margin）
2. 在IE6,7,下，li里的标签内容浮动，但是li本身没有浮动,li下面会多出一个间隙

解决方法：1.给li加浮动 2.给li加垂直对齐方式

1. 在IE6,7下，浮动的父级，有宽度的情况下，不需要清浮动
2. 在IE6下，两个浮动元素之间有注释或者内嵌元素并且浮动元素的宽度和父级宽度相差小于3px，元素最后的文字会被复制

解决方式：用div把注释和内嵌包起来

1. 在IE6下，浮动元素和绝对定位元素是并列关系，绝对定位元素会消失

解决方法：把定位元素单独包起来

1. 在IE6,7下，元素有相对定位，父级的overflow不起效果

解决方法：给父级加相对定位

1. 在IE6下，通过margin是元素移出父级的部分，会被父级截掉

解决方法：给元素加相对定位

1. 在IE6下，定位父级的宽高是奇数的话，元素的right 和 bottom 会有1px 的偏差

解决方法：无

1. 在IE6下，不支持固定宽高（最大最小宽高）

解决方法：用js模拟

1. 在IE6,7下，不支持inherit属性
2. 在IE8及之前的浏览器下，不支持opacity

解决方法：用filter代替（filter:alpha(opacity = 0~100);）

1. 在IE6,7下，输入类型表单控件，上下各有1px的间隙

解决方法：给元素添加浮动

1. 在IE6下，表单控件设置 border: none 无效

解决方法：1.border:0; 2.重置元素背景

1. 在IE6,7下，输入类型表单控件添加背景图，输入文字时图片会一起移动

解决方法：背景图片加给父级

1. 在IE6下，不支持PNG的透明

border-radius(复合样式，css3，IE9+）圆角半径

border-top-left-radius

border-top-right-radius

border-bottom-right-radius

border-bottom-left-radius

多值写法：一值（左上，右上，右下，左下）

二值（左上，右下）（右上，左下）

三值（左上）（右上，左下）（右下）

四值（左上）（右上）（右下）（左下）

box-shadow 盒模型阴影

(inset[可选] x y 模糊半径 扩展半径[可选] color) css3新增

inset: 内阴影 x: x轴的偏移量 y:y轴的偏移量

模糊半径： 阴影的扩展半径，从阴影显示到完全消失的距离

扩展半径：在元素的轮廓之后，扩展一定像素之后，再消失

color:颜色

元素拥有多层阴影，每层用逗号隔开

元素拥有多层阴影，先写的在上边

X,Y半径不一致 x半径/y半径

百分比（不建议移动端使用）

x半径根据宽度计算百分比，y半径根据高度计算百分比

块标签：header footer nav section article aside main figure

figcaption div p ul ol li dl dd dt form option h1 h2 h3 h4 h5 h6

内嵌：a mark em span strong time label i b

内嵌块：img input textarea select button

宽度由内容撑开：display:inline; display:inline-block; float:left; float:right; position:fixed; position:absolute;

脱离文档流：float:left; float:right; position:fixed; position:absolute;

CSS sprite 有什么优缺点：

优点：减少http请求，减轻服务器压力

压缩文件体积 减少命名 便于风格统一优化

缺点：增加开发成本 更换单图不方便

选择器[属性名称]

从选择器中筛选具有这个属性名称的元素

选择器[属性名称=”属性值”]

从选择器中筛选具有这个属性名称，并且属性值为指定的元素

选择器[属性名称~=”属性值”]

从选择器中筛选出具有这个属性，并且属性值是个词列表，词列表中包含指定值得元素

选择器[属性名称^=”值”]

从选择器中筛选具有这个属性名称，并且属性值以指定值开头的元素

选择器[属性值$=”值”]

从选择器中筛选具有这个属性名称，并且属性值以指定值结尾的元素

选择器[属性值\*=”值”]

从选择器中筛选具有这个属性名称，并且属性值包含在指定值的元素

选择器[属性名称|=”值”]

从选择器中筛选具有这个属性名称，并且属性值为指定值，或者以”指定值-”开头的元素

子类选择器

父类选择器 > 子类选择器

只会往父级的下一级进行查找 不会深入查找下一级内部的子级

兄弟选择器

某元素后面紧跟着的下一个兄弟（下一个兄弟元素）

比如： .title + div

某元素后面的所有兄弟

比如： .title ~ div

.wrap h2:first-child 首先找到wrap内部所有的h2，并且这个h2是h2所在父级的第一个子元素，则找到这个h2

.wrap h2:last-child 首先找到wrap内部所有的h2，并且这个h2是h2所在父级的最后一个子元素，则找到这个h2

.wrap h2:nth-child(2) 首先找到wrap内部所有的h2，并且这个h2是h2所在父级的第二个子元素，则找到这个h2

.wrap h2:first-of-type 首先找到这个h2是和他同级的所有h2中的第一个，则找到这个h2

.wrap h2:last-of-type 首先找到这个h2是和他同级的所有h2中的最后一个，则找到这个h2

.wrap h2:nth-of-type(2) 首先找到wrap下所有的h2，并且这个h2是和它同级所有h2中的第二个，则找到这个h2

.wrap h2:nth-last-of-type(2) 首先找到wrap下所有的h2，并且这个h2是和它同级所有h2中从后往前数的第二个，则找到这个h2

.wrap h2:only-child 首先找到wrap下所有的h2，并且这个h2是它所在父级的唯一一个子元素，则找到这个h2

.wrap h2:only-of-type 首先找到wrap下所有的h2，并且这个h2是它所在父级的唯一一个h2，则找到这个h2

div: empty; 找到所有div,找到其中完全为空的元素，选择到他们

.wrap h2:nth-child(n) --2n和2n+1 --隔行变色效果

background-clip背景图裁切(显示位置)

属性值

border-box 背景被裁剪到边框盒

padding-box 背景被裁剪到内边框盒

content-box 背景被裁剪到内容盒

文字部分背景裁切

-webkit-background-clip: text; 在文字部分显示背景

background-origin 背景图原点（初始位置计算）

border-box 背景图像相对于边框盒来定位

padding-box 背景图像相对于内边距框来定位

content-box 背景图像相对于内容框来定位

css3提供多背景图设置

写法：多背景之间用逗号隔开

先写的在上面，后写的在下面

background-size length具体长度值

设置背景图像的高度和宽度，第一个值设置宽度，第二个值设置宽度。如果只设置第一个值，则第二个值会被设置为“auto”

percentage百分比

以父元素的百分比来设置背景图像的宽度和高度，第一个值设置宽度，第二个设置高度。如果只设置一个值，则第二个值会被设置为“auto”

cover 把背景图片拉伸或缩放值足够尺寸，以使宽度高度安全适应内容区域

background: linear-gradient()背景图：线性渐变

写法：只写入颜色值

例如：background: linear-gradient(red, blue, yellow)

指定点进行渐变

background: linear-gradient(red 0, blue 20px, yellow 30px)

background: linear-gradient(red 0%, blue 10%, yellow 30%)

渐变角度的设置

关键字： 注意：需要加上浏览器的内核前缀

-webkit-linear-gradient(left top, ……);

角度： 注意：加内核前缀和不加内核前缀的角度计算方式不同

渐变角度设置： background:repeating-linear-gradient();

animation

动画帧： @keyframes name {

from {

//起始值

}

to {

//目标值

}

}

调用：animation

animation-name 动画帧名字

animation-duration 动画支持事件（ms/s）

animation-delay 延迟时间（ms/s）动画首次执行时的延迟

animation-iteration-count 动画次数（number/infinite）

animation-direction 规定是否应该轮流反向播放动画（normal、alternate）

animation-timing-function 规定动画的速度曲线

animation相关问题

1. 如果初始值和计算后的样式不一样，在动画执行时，会直接跳到动画的初始位置执行
2. 如果结束值和计算后的样式不一样，在动画执行完成时，会直接跳回计算后的样式
3. 不设置初始值，初始值默认就是计算后的样式
4. 不设置结束值，结束值默认就是计算后的样式

计算后的样式：元素上优先级最高的一条样式（元素最终显示出来的样式）

animation-fill-mode 初始值，结束值和计算后的样式不一样时的处理：

none

forwards 当动画结束时，位置保持在结束值

backwards 在animation-delay 所指定的一段时间内，在动画显示之前，应用开始属性值（在第一个关键帧中定义）

both forwards 和backwards

animation-play-state (running 执行/ paused 暂停)

IE10及IE10+ 支持

animation-play-state (running 执行/ paused 暂停)

-webkit-animation-

浏览器前缀

-webkit- -moz- -ms-

<input type=”range” min=”0” max=”10” step=”3” value=”0” />

range: 数值选择器 step：步数（数值间隔）value：当前数值

datalist：定义input可能出现的值

内部为option

通过list的属性吧input 和 datalist 关联起来

list属性值是datalist 的id

placeholder提示信息

required 设置该项的必填项

autocomplete 是否记录用户呼入的值（on/off）

tabindex 用户按下tab键进行切换时，焦点切换的顺序

autofocus 焦点

pattern 正则校验

formaction 在submit定义一个新的提交地址

formnovalidate 取消验证

视口（可视窗口）：设备可视大小的相关设置

在移动端不设置视口，默认为980；

viewport(视口设置)

width(设置窗口的宽度，number || device-width) 一般不会设置成number，因为部分手机不支持

inital-scale(初始比例缩放 number)

minimum-scale(最小比例缩放 number)

maximum-scale(最大比例缩放 number)

user-scalable(是否允许用户缩放 yes||no)

初始值和最小值一般要保持一致，否则不出效果。缩放不是把可视区缩放了，真正缩放的是可视区中的内容，所以设置了缩小之后反而可视区变大了

//window.devicePixelRatio 像素比（1px的内容放大至npx来显示）

a, button, input {

-webkit-tap-highlight-color: transparent; //清楚默认样式

} 用户按下的高亮颜色

button, input {

-webkit-appearance: none;

border-radius: 0; //清除ios下按钮的圆角

}

body\*{

-webkit-text-size-adjust: 100%; //防止横竖屏切换时文字被缩放

}

body {

font-family: helvetica; //目前只有这一个字体是在各个设备都有

}

当元素有绝对定位并且没有定位父级，那这个时候从body下超出，body的overflow不起作用

解决方法：给body添加相对定位

在body下添加 一个标签，width和body一致，添加overflow和相对定位

自适应+ 百分比布局

手机（<768px 小屏幕） 平板（>=768px 中等屏幕） 桌面（>=992px 大屏幕） 桌面（>=1200px）

平板竖屏 768 横屏 1024 320 375 425

(function(){

toSize();

window.addEventListener('resize',toSize);//屏幕大小改变的时候，重新计算一下 rem的基准值

function toSize(){

var html = document.documentElement;//获取html

var width = html.clientWidth; //获取到html的宽度

var nub = 6.4;//把rem的基准值(也就是html的字体大小)，设置成屏幕宽度的多少分之一

html.style.fontSize = width / nub + "px";//设置html的字体大小

}

})();

text-shadow:x偏移y偏移 模糊半径 阴影颜色

text-stroke: 文字描边 描边的宽度 描边的颜色

-webkit-text-stroke webkit私有样式

radial-gradient 径向渐变 IE10+

radial-gradient([[<shape>||<size>][at<position>]?,|at<position>,]?<color-stop>[,<color-stop>]+);

shape:形状 (circle||ellipse)

size: 大小设置

1) 数值 直接设置 半径 (老版本火狐不支持)

2) 关键字最近端，最近角，最远端，最远角，包含或覆盖 (closest-side, closest-corner, farthest-side, farthest-corner, contain[老版本才有] or cover[老版本才有]

新写法: radial-gradient( [大小或形状] at [中心点坐标],red,blue);

mask 复合样式

mask-image: url();

mask-repeat: repeat|repeat-x|repeat-y|no-repeat;

mask-position: x y;

border-image-source 用在边框的图片的路径。

border-image-slice 图片裁切

border-image-repeat 图像边框是否应平铺(repeated)、铺满(rounded)或拉伸(stretched)。

filter

grayscale灰度

sepia褐色（求专业指点翻译）

saturate饱和度

hue-rotate色相旋转

invert反色

opacity透明度

brightness亮度

contrast对比度

blur模糊

drop-shadow阴影

-webkit-box-reflect 倒影

above/below/left/right

倒影和元素之间的距离

渐变消失

matrix 矩阵 矩阵方程

rotate() 旋转 (deg 角度)

scale() 缩放 (倍数)

scaleX()

scaleY()

skew 斜切 (deg)

skewX()

skewY()

translate 位移

translateX(px)

translateY

默认围绕中心点进行变化

transform-origin 变换远点设置（默认元素的中心点）

1. 关键字

2. 具体数值(从元素的左上角开始计算)

translate 位移

translateX(px)

translateY

transform 有多个操作的时候，每种操作之间以空格隔开

transform 要加载块属性的元素上

在移动端书写 transform时 记得添加前缀

-webkit-transform

transform 有多个值得时候，会先写的后计算

rotate 旋转

rotateX 围绕x轴旋转 （上下翻转）

rotateY 围绕y轴旋转 （左右翻转）

rotateZ 围绕z轴旋转

rotate3d(x,y,z);

translate(x,y)

translateX

translateY

translateZ

translate3d

perspective 在3d中 设置 元素距离视图的距离；

距离越近 近大远小的透视关系 却强烈

transform-style:

preserve-3d; 当元素在进行3d变换时，保留子元素的3d变换效果

flat 当元素在进行3d变换时，把子元素的变成平面的(默认值)

backface-visibility: hidden; 隐藏背面

和父级角度相对的面 叫做背面

弹性盒模型

新版：

display: flex; 弹性盒模型

flex-direction 主轴方向设置(控制子元素的排列方向)

row 从左向右排列(默认值)

row-reverse 从右向左排列

column 从上往下排列

column-reverse 从下往上排列

justify-content: 主轴对齐

flex-start 元素在主轴的开始位置，多余空间在主轴的结束位置

flex-end 元素在主轴的结束位置，多余空间在主轴的结束位置

center 元素在主轴的中心，多余空间在主轴的两侧

space-between 富裕空间在元素之间平均分配

space-around 富裕空间在元素两侧平均分配

align-items: 侧轴对齐

flex-start (默认) 元素在开始位置 富裕空间占据另一侧

flex-end 富裕空间在开始位置 元素占据另一侧

center 元素居中 富裕空间 平分左右两侧

flex-wrap 换行

nowrap (默认)

wrap 换行

wrap-reverse 反向换行 (更主轴方向相反的换行)

align-content 行对齐

flex-start 元素在开始位置 富裕空间占据另一侧

flex-end 富裕空间在开始位置 元素占据另一侧

center 元素居中 富裕空间 平分左右两侧

space-between 富裕空间在元素之间平均分配

space-around(默认) 富裕空间在元素两侧平均分配

老版：

display: -webkit-box; 弹性盒模型老版本写法

-webkit-box-orient: (horizontal||vertical) 控制子元素水平显示还是垂直显示 (主轴方向设置)

-webkit-box-direction: (normal||reverse);

控制子元素正序排列还是倒序排列

-webkit-box-pack 主轴对齐

start 元素在主轴的开始位置，多余空间在主轴的结束位置

end 元素在主轴的结束位置，多余空间在主轴的结束位置

center 元素在主轴的中心，多余空间在主轴的两侧

Justify 富裕空间在元素的两侧平均分布

-webkit-box-align 侧轴对齐

Star 所有子元素在据顶

End 所有子元素在据底

Center 所有子元素居中

portrait竖屏 landscape 横屏

orientation的横竖屏检测时根据页面的宽高比来计算的

当页面的宽度大于高度就会认为是在landscape

当页面的高度大于宽度就会认为是在portrait

手机（<768px） 平板（>=768px） 桌面（>=992px） 桌面（>=1200px）

手机主流页面宽度 320 375 425

平板 竖屏768 横屏 1024

电脑 1280 1366 1440 1920

column-count 规定元素应该被隔开的列数

column-fill 规定如何填充列

column-gap 规定列之间的间隔

column-rule 设置所有的column-rule-\*属性的简写属性

column-rule-color column-rule-style column-rule-width

column-span 规定元素应该跨的列数

column-width