CSS3 Basic

一、html属性所带来的问题

1、相同的效果，不同的标签会通过不同的属性来表示

2、使用属性很不通用，一对一的修改，而不是整体修改

二、使用CSS来解决上述问题

1、什么是CSS

CSS:Cascading Style Sheet(层叠样式表、级联样式表、样式表)

2、作用：

（1）实现内容和表现分离

（2）提高代码的可重用性和可维护性

3、html属性和CSS样式

能用CSS的话，一定使用CSS来实现样式；如果CSS实现不出来效果，则选择使用html属性

三、使用CSS样式表

1、使用CSS的方式

（1）内联方式（行内样式）

将CSS样式定义在某个单独的标签中，例如<div style="color:red;"></div>

（2）内部样式表

将CSS样式定义在网页中的<head></head>中

特点：针对当前网页有效

（3）外部样式表

将CSS样式定义在一个外部的.CSS的文件中，需要时进行引用即可

特点：所有网页公用，是真正意义的实现了重用性和可维护性

\*样式表中的注释：/\*... ... \*/

2、内联方式（行内样式）

将样式内容写在html元素中的style属性中

CSS语法：

常用属性： color，修改当前标签中文本的颜色

background-color，修改当前标签的背景颜色

3、内部样式表

【样式规则】 存放在html文档头部head标签中的style标签内

样式规则： 选择器｛

样式声明

属性：属性值；属性：属性值；...

｝

\*选择器： （1）定义了一组样式的名称

（2）哪些元素能够使用定义好的样式

例如 <style>

p{

color:red;

background-color:blue;

}

</style>

4、外部样式表

将 样式规则 单独存放在 样式表文件中（\*\*.css），需要时直接引用

步骤1，创建一个文本文件，将其后缀修改为.css，在该文件中编写若干样式规则

步骤2，通过<link/>在head中引入外部样式表

\*<link rel="stylesheet" type="text/css" href="样式表文件路径"/>

四、样式表特征

1、继承性

大多数的css样式规则可以被继承

2、层叠性

为同一个元素定义多个样式规则

多个样式规则同时存在时不冲突，最终多个样式规则会合并成一个，\*\*但是，属性重复时以最后一个出现为准

3、优先级

各级样式表冲突时（行内、内部、外部），会按照不同的优先级来应用样式

0（低）：浏览器缺省设置

1（中）：内部样式表或外部样式表

\*内部和外部冲突时：就近原则，谁在下，以谁为主

2（高）：行内样式（内联样式）

4、！important规则（！--> 非）

通过！important显示调整样式规则的优先级，放在属性值之后，用 空格 来区分，由它标识的属性值，优先级别最高

例如： p{

color:red !important;

background-color:blue;

}

五、CSS基础选择器

1、元素选择器（标签选择器、标记选择器）

特点：通过元素名称作为选择器名称，例如p、div、h1等

作用：修改某一元素的默认样式

2、类选择器

特点：通过元素的class属性来进行应用

作用：定义某一组标签的统一样式

语法：.className{...}

\*注意：className不能以数字开头

引用：<div class="className"></div>

\*\*注意：在一个class中可以引用多个类样式，多个类样式之间用 空格 隔开

例如<div class="className username"></div>

3、分类选择器

由类选择器衍生出来的新选择器，可以将类选择器与元素选择器结合使用

目的：为了更精准的定位到页面的元素

语法：元素选择器.类选择器{...}

例如 div.abcd{...}

效果举例： <head>

.abc{color:red;}

div.abc{background-color:pink;}

</head>

<body>

<div class="abc">据阿联接到国家馆</div>

<h2 class="abc">据阿联接到国家馆</h2>

</body>

效果说明：div和h2中字体都为红色，div中背景为粉色

4、通用选择器

作用：用于适配页面上所有的元素，改变他们的样式

语法：\*{...}

5、id选择器

作用：通过页面元素的id值来进行选择器的引用，非常方便的定位到页面上的某一个元素

语法：#id{...}，依然是放在<style></style>中

例如：#abc{color:red;}

<div id="abc"></div>

6、群组选择器

选择器声明是以 ，隔开的选择器列表

作用：定义一组元素的样式

例如 div,p,.abc,div.abc,#aaa{...}

7、后代选择器

根据元素的嵌套关系来定义的样式

根据一个元素去定位它里面的其他元素

语法：选择器1 选择器2{...}，中间用空格连接

例如 <div id="abc">

<div>

<p>

<span></span>

</p>

</div>

<span></span>

</div>

（1）#abc span{...}效果：两个span都是后代元素，都会发生变化

（2）#abc div p span{...}效果：只有上面的第一个span会发生变化

8、子代选择器

要求选择器之间只能存在父子关系

语法：选择器1>选择器2{...}

例如： #abc>.aaa{

修改id为abc里面的class为aaa的元素（下一级）

}

\*\*注意：后代选择器和子代选择器目的是为了精确匹配范围

9、伪类选择器

匹配元素不同状态时候的选择器

语法：选择器1:伪类选择器{...}

分类： 链接伪类

:link 适用于尚未访问的链接

:visited 适用于访问过的链接

动态伪类

:hover 适用于鼠标悬停在元素上的时候的状态

:active 元素被激活瞬间的状态

:focus 适用于元素获取焦点时的状态，比如input（text、password）、textarea

目标伪类

元素状态伪类

结构伪类

否定伪类

六、颜色值

1、rgb(0,0,0)

r：red 0-255

g：green 0-255

b：blue 0-255

rgb（0,0,0）表示黑色 rgb（255,255,255）表示白色

2、rgb(%,%,%)

3、#rrggbb

r: 0-9 A-F

g: 0-9 A-F

b: 0-9 A-F

#000000黑色 #ffffff白色 #ff0000红色 #e4393c京东红

4、#rgb

是#rrggbb的缩写，但是必须是r、g、b值两位一致时使用

七、属性---尺寸

设置元素的宽度和高度

1、宽度

width:

max-width: ↓

min-width:一同定义元素宽度的范围，与width冲突，不能同时定义，如同时出现，以width为主

2、高度

height:

max-height: ↓

min-height:一同定义元素高度的范围，与height冲突，不能同时定义，如同时出现，以height为主

\*注意：只有 块级 以及 行内块（比如img） 元素能够更改width和height属性，行内元素不能修改

3、溢出

作用：当内容溢出元素框时处理

属性： (1)overflow

值： visible 显示（默认值）

hidden 隐藏

scroll 有滚动条（不管溢出与否，都有滚动条{前提是设置了宽和高，如果只设置了宽，则不显示高的滚动条}）

auto 自动，溢出则显示滚动条，不溢出则没有

(2)overflow-x：横向溢出

值：与overflow一致

(3)overflow-y：竖向溢出

值：与overflow一致

八、边框

属性：

简写方式 border:width style color;

width：边框线条的宽度

style：边框线条的样式，有实线（solid）、虚线（dotted）

color：边框线条的颜色

例如 border:1px solid #f00;

单边定义 border-方向：... ... ...

方向值：top bottom left right

单一属性 border-width:...

border-style:...

border-color:...

组合 border-方向-属性:...

例如 border-top-color:#000;

border-left-width:2px;

\*注意：边框颜色除可以设置正常颜色值之外，还可以设置为transparent（透明）

九、边框倒角

元素的四个方向的圆角设置

属性：border-radius

取值： 1、给定四个值，分别表示从左上角开始按顺时针方向的四个角圆角半径

2、给定一个值，表示四个角的圆角半径

给定指定数值： 1、px表示

2、百分比表示，当前元素的宽度的百分比，作为圆角半径

border-radius:100px;或者border-radius:50%;表示圆

单独定义（了解即可）： border-top-left-radius:...

border-top-right-radius:...

border-bottom-left-radius:...

border-bottom-right-radius:...

十、边框阴影

属性：box-shadow 给指定元素边框增加阴影

box-shadow:h-shadow v-shadow blur spread color inset;

h-shadow:阴影水平位置

v-shadow:阴影垂直位置

blur:模糊程度

spread:阴影尺寸

color:阴影颜色

inset:可选，将阴影改为内阴影，默认为外阴影

例如 box-shadow:5px 5px 10px #000 inset;

常用写法：box-shadow:0px 0px 5px #f00;

十一、图片边框

让图片作为元素的边框出现

属性：border-image:source width repeat;

十二、轮廓

在元素边框的周围再绘制一条线

属性：outline:width style color;

其中width style color取值跟border的一样

十三、框模型

1、什么是框模型

框：html中所有的元素都可以认为是框。框中可以盛装内容。

框模型：指的是框与框，以及框与内容之间的关系。

定义了外边距、内边距、边框

\*注意：可以通过width、height定义框的宽度和高度

如果增加了内边距、外边距、边框的时候，不会影响内容区域的尺寸，但会增加框的总尺寸

元素的实际宽度=左外边距+左边框+左内边距+内容区域（width）+右内边距+右边框+右外边距

十四、外边距

1、什么是外边距

在元素边框周围的一些空白区域，该空白区域正常情况下是不允许有其他元素显示的。

2、外边距属性

margin:value;

表示上下左右四个方向的外边距

单边设置 margin-方向:value;

方向：top bottom left right

margin:值1 值2;(两个值中间用空格隔开)

值1表示上下外边距，值2表示左右外边距

margin:值1 值2 值3;

值1表示上外边距，值2表示左右外边距，值3表示下外边距

margin:值1 值2 值3 值4;

数序为 上、右、下、左

取值：px % 负值 auto

\*注意：当元素的左右外边距设置为auto时，那么该元素会在其父层容器内变的水平居中

但是将上下外边距设置auto时，没有任何效果

3、html中，某些元素是具备默认外边距的

h1-h6 p body 等

如果设置以上元素的margin属性值，则会覆盖掉他们的默认值

4、垂直外边距的合并

当两个垂直外边距相遇时，他们将合并形成一个外边距，以最大值为准

例如，上面元素下外边距为50px，下面元素上外边距为50px，则他们相距仍为50px；

上面元素下外边距为50px，下面元素上外边距为80px，则他们相距为80px；

十五、内边距

1、什么是内边距

框内的内容与框之间的距离空间

\*注意：增加内边距之后会扩大元素边框所占的区域

2、语法：

padding:value;

取值：px % 不能取负值 没有auto值

padding:v1;表示四个方向

padding:v1 v2;表示上下和左右

padding:v1 v2 v3;表示上、左右、下

padding:v1 v2 v3 v4;表示上、右、下、左

padding-方向:value;

十六、背景图像

1、属性：background-image

取值：url(路径)

例如 background-image:url(image/1.jpg);

2、背景重复（repeat）

属性：background-repeat:值；

取值： repeat，垂直、水平双方向重复，默认值

repeat-x，水平方向重复

repeat-y，垂直方向重复

no-repeat，不重复

3、背景图片尺寸

属性：background-size:值；

取值： value1 value2，宽和高

value1% value2%，宽和高

cover，覆盖（无限扩大，覆盖当前背景的所有边，能铺满整个背景区域，但不一定能显示完整）

contain，包含（只要有一个边达到容器的边就停止扩大，不一定能铺满整个背景区域，但是能显示完整图片）

5、背景图像固定

属性：background-attachment

取值： scroll，不固定，默认值

fixed，固定

6、背景定位

指的是背景图在元素中的位置

属性：background-position

取值：x y，水平和垂直偏移位置（可以取负值，超出部分不显示）

（1）xy以像素为单位

（2）x以left，center，right取值

y以top，center，bottom取值

x% y%，水平和垂直偏移位置（若都取50%则居中显示）

7、背景裁剪-背景绘制区域

背景从哪开始剪

属性：background-clip

取值： border-box，默认值，背景被裁剪到边框上

padding-box，背景被裁剪到内边距上，不包括边框

content-box，背景被裁剪到内容区域上，不包括边框和内边距

8、背景定位区域

背景从哪开始画

属性：background-origin

取值： border-box，默认值，背景从边框开始画

padding-box，背景从内边距开始画

content-box，背景从内容开始画

9、背景综合属性

background:color url() repeat attachment position;

常用写法： （1）background:#f00;

（2）background:url() no-repeat -100px -100px;

十七、渐变

1、什么是渐变：在多种颜色之间进行柔和的过渡

2、语法

属性：background-image

取值： linear-gradient(渐变值)，表示线形渐变

radial-gradient(值)，表示径向渐变

repeating-linear-gradient()，重复线形渐变

repeating-radial-gradient()，重复径向渐变

3、线形渐变

background-image:linear-gradient(angle,color-point);

值：angle，渐变的方向或角度

方向值：to top，向上

to bottom，向下

to left，向左

to right，向右

角度：以deg结尾，比如30deg

color-point，表示颜色的起始点、各个过渡点、结束点，是由颜色和位置组合起来

比如color-point:red 50%

例如background-image:linear-gradient(to bottom,red 0%,green 50%,blue 100%);表示从上往下，从红色0%到绿色50%到蓝色100%

4、径向渐变

background-image:radial-gradient([size at position],color-point);{【】表示可以省略，省略则为默认值，从元素的正中心开始放射渐变}

值： size，圆的半径

position，圆心出现的位置，默认为元素的中心

比如background-image:radial-gradient(200px at top left,red 0%,green 50%);表示从左上角以200像素为半径从红色到50%处渐变到绿色

5、重复线形渐变

background-image:repeating-linear-gradient(angle,color-point);

color-point中point的取值以像素为单位，不用%

\*重复径向渐变同理

6、浏览器兼容性问题

对不支持的浏览器版本处理：通过加浏览器前缀去匹配不同的浏览器

Firefox，火狐，加-moz-

Chrome、Safari，加-webkit-

Opera，欧鹏，加-o-

例如#div{

background-image:-moz-linear-gradient(...);

}

如果要想所有的都支持，则例如

#div{

background-image:linear-gradient(...);

background-image:-moz-linear-gradient(...);

background-image:-webkit-linear-gradient(...);

background-image:-o-linear-gradient(...);

}

十八、文本格式化

1、控制字体

（1）指定字体

font-family:"微软雅黑","宋体",arial;

（2）字体大小

font-size:...; 以px或pt为单位

常用字体大小12px-14px

（3）字体加粗，相当于<b>标签

font-weight:normal/bold;正常或加粗的写法

:400-900之间表示，400表示normal，900表示bold

（4）字体样式--斜体，相当于<i>标签

font-style:normal/italic;正常或斜体的写法

（5）小型大写字母（只对英文有效）

font-variant:normal/small-caps;

\*效果：将文本中小写字母转换成小型大写字母，原大写字母则保持不变

（6）综合写法

将所有的属性设置在一个font当中，空格隔开

例如font:italic small-caps bold 12px “微软雅黑”,arial;

\*注意font-size另一种写法：font-size/line-height，表示字体大小和行高

2、文本属性

（1）颜色

例如rgba(255,0,0,0.5)表示半透明，a的取值表示透明度

（2）文本排列

text-align:left/right/center;水平对齐，大部分主流浏览器只对行内元素起作用，对块不起作用

vertical-align:top/bottom/middle/baseline（基线对齐）;垂直对齐，主要针对<td>

（3）文字修饰--线条

text-decoration:none/underline

none:去除文本上的线条

underline:下划线

overline:上划线

linethrough:删除线，相当于<s>

（4）行高

line-height

常用方式：

a、如果行高的值高于文本的高度，且设置为等于容器的高度，那么这段文本会在行高范围内垂直居中显示（但只限于只有一行文本的情况下）

b、给文本增加上下的边距

（5）首行文本缩进

text-indent:value;以像素为单位

（6）文本阴影

text-shadow:h-shadow v-shadow blur color;

3、文本溢出处理

（1）处理空白

属性 white-space:normal/nowrap;

normal，自动换行，默认

nowrap，不换行

一般配合overflow:hidden;（溢出隐藏）使用

（2）文本溢出

属性text-overflow:

取值： clip，裁剪、切割

ellipsis，隐藏溢出内容并用...表示出来

\*\*注意：【必须】配合overflow:hidden;使用，因为必须先隐藏，才能处理隐藏内容，如果都没隐藏，处理什么？

（3）长单词换行（一般用于英文）

属性word-wrap

取值： normal，

break-word，分开英文单词强制换行

（4）文本换行（段落）

属性word-break

取值： normal，

break-all，强制一切，换行，保证不超出元素界限

keep-all，保持英文完整度的前提下才换行（通过空格和标点来识别）

十九、表格

1、常用属性（标准属性）

（1）padding

（2）width，height

（3）border

（4）background

（5）文本格式化属性，如font、color等

（6）对齐text-align，vertical-align

2、表格【特有】的样式属性

（1）边框合并

属性：border-collapse

取值： separate，分离边框，默认值

collapse，边框合并

（2）边框间距，相当于table中的cellspacing

属性：border-spacing

取值： i 指定一个值，表示水平和垂直统一间距

ii 制定两个值，第一个表示水平间距，第二个表示垂直间距，中间用空格隔开

\*注意：使用的前提是border-collapse值必须为separate，即分离状态下

（3）标题位置，即<caption>出现的位置

属性：caption-side

取值： top，默认值，表格上端

bottom，表格下端

（4）显示规则，即如何控制表格中单元格的算法

属性：table-layout

取值： auto，列的宽度由单元格的内容来决定，默认值，称为自动表格布局

fixed，列宽度由列的width属性来决定，不受内容的影响，称为固定表格布局

\*自动表格布局：处理复杂表格时速度会稍慢，不适合确定列大小时使用

固定表格布局：速度略快，适合用于确定列宽度时使用

二十、浮动

1、什么是定位：元素该出现的位置

分类： （1）普通流定位（文档流定位）（默认）

（2）浮动定位

（3）相对定位

（4）绝对定位

（5）固定定位

2、普通流定位

块级元素：从上往下依此排列

行内元素：从左到右排列

3、浮动定位

（1）什么是浮动定位

元素会脱离默认文档流，在页面上不会占据空间；

浮动定位的元素会放置在包含框的左右或右边，但是浮动的元素依然在包含框内；

当浮动元素碰到其他浮动元素时，就会停止浮动；

元素一旦浮动起来，那么都将成为块级元素

（2）能够解决：实现特殊的定位方式，比如让多个块级元素在同一行内显示

（3）属性：i float

取值：left，左浮动

right，右浮动

none，无浮动，默认值

ii clear，清除浮动所带来的影响

取值： left，清除左边元素带来的影响

right，清除右边元素带来的影响

Both

4、浮动元素对父层元素带来的影响

一个元素内如果包含了浮动元素，那么该元素的高度可能会变成0，如果里面还有未浮动的元素，那么高度就会变成自适应

原因：浮动元素脱离了文档流，理论上讲就不在父层容器内了

解决方案： （1）显示设置父层元素的高度

（2）通过overflow:hidden来撑起父层容器的高度

二十一、显示方式

属性：display，改变元素的显示方式，比如由行内改变成块级，或者块级改变成行内

取值： 1、none，隐藏，让生成的元素没有框

特点：隐藏之后在页面上不显示，不占据页面空间

2、block，由行内元素变成块级元素

常用：想显示修改行内元素的高度和宽度的时候，可以将其显示形式设置为block

3、inline，由块级元素变成行内元素

4、inline-block，行内块

特点：本身为行内元素，但是具备块级元素的特点，多个行内块在同一行内显示，但是可以改变其宽度和高度

常用：想改变行内元素的高度和宽度的时候

二十二、显示效果

属性：visibility，真正实现元素显示与隐藏

取值： 1、visible，默认值，元素可见

2、hidden，隐藏，元素隐藏，但是空间依然保留

3、collapse，用在表格，删除一行或一列，不影响表格布局

\*\*visibility与display的区别：

1、visibility是真正实现元素的显示与隐藏的，而display是改变元素的显示方式

2、visibillity：hidden能够实现元素的隐藏，但会占据页面空间；display：none没有显示方式，能实现元素的隐藏，并且不会占据空间

二十三、透明度

属性：opacity

取值：0-1之间的小数，值越大越不透明，值越小越透明

二十四、cursor，更改鼠标形状

取值： default，默认值

pointer，小手

crosshair，十字、加号

text，I样

wait，等待，沙漏、圆圈，根据浏览器不同

help，鼠标右下角带小问号

二十五、列表

1、列表项标致，list-style-type

取值： （1）无序列表

none，disc，circle，square

（2）有序列表

none，decimal，lower-roman，upper-roman

2、列表项图像，list-style-image

取值：url(...)

例如 list-style-type:none;

list-style-image:url(...);

3、列表项位置

属性：list-style-position

取值： outside，在文本区域之外

inside，在文本区域之内

4、列表综合属性

属性：list-style:type url() position;

常用值：list-style:none;

5、常用用法

（1）css重写，也就是将一些标签的默认样式修改掉

\*{

margin:0px;

padding:0px;

list-style:none;

}

（2）导航

二十六、定位

1、定位方式

（1）浮动定位

（2）相对定位、绝对定位、固定定位、静态定位

2、属性

（1）定位方式相关属性 -- position

取值： static，默认，静态定位

relative，相对定位

absolute，绝对定位

fixed，固定定位

（2）与定位位置相关属性

top、bottom、left、right

取值以像素为单位

（3）堆叠顺序 -- z-index

取值为无单位的数值，值越大，越靠近用户

3、相对定位

相对于当前元素本身出现的位置而实现定位的一种方式

实现方式：position:relative;

常用： （1）进行位置的微妙的调动

（2）配合绝对定位使用

例如 position:relative;

left:-10px;

bottom:-10px;

表示向左下方偏移10px

4、绝对定位

脱离文档流，不占据页面空间

定位位置：相对于最近的【已定位】的祖先元素；如果祖先元素没有进行定位，它的位置就相对于最初的包含块（body）

已定位：非static，指relative、absolute、fixed

实现方式：position:absolute;

取值与relative一致

常用：（1）弹出菜单的位置

5、固定定位

将元素固定在页面上的某个位置，不随滚动条的滚动而发生改变

语法：position:fixed;

top、bottom、left、right相对于body取值