# 第一天

1、CSS概述

1、CSS概述

1、问题

1、相同的效果用不同的属性

<body text="blue">

<font color="red">

2、可维护性和可重用性不高

2、CSS

1、什么是CSS

Cascading Style Sheets

层叠样式表、级联样式表、样式表

2、作用

1、实现 内容与样式 相分离

2、提高代码的可重用性和可维护性

3、CSS与HTML之间的关系

HTML ：人 - 搭建网页结构

CSS ：衣服装扮 - 构建网页样子

4、CSS与HTML使用原则

尽量使用css属性 取代 HTML 属性

能取代：

html ：align

css ：text-align

html：color

css：color

不能取代：

colspan

rowspan

2、使用样式表

3、使用CSS样式表

1、使用样式表方式

1、内联方式

将 样式定义在 元素的style属性中

语法：

<标记 style="样式声明;样式声明"></标记>

样式声明：样式属性:样式值

ex:

color:red; 当前元素文本为红色

background-color:blue;背景颜色

font-size:20px;字体大小

弊端：

1、没有体现出 可重用性和可维护性

2、内部样式表

1、what

将样式定义在 head 元素中的 style 元素中

2、语法

<head>

<style>

若干"样式规则"

</style>

</head>

样式规则组成：

1、选择器

规定了页面中[哪些]元素能够使用定义好的样式

2、若干样式声明

样式属性:值;

选择器{

样式属性:值;

样式属性:值;

}

h1{

color:red;

font-size:12px;

}

特点：

1、提升了样式的可重用性和可维护性

3、外部样式表

1 、作用

将样式声明在独立的样式文件中(\*\*.css),并且独立于任何 HTML文档

2、步骤

1、创建css文件并声明样式规则

2、在页面中对 css 文件进行引入

<head>

<link rel="stylesheet" href="css样式表文件url">

</head>

2、CSS样式表特征

1、继承性

大部分的CSS样式属性是可以被继承

2、层叠性

可以为一个元素定义多个样式规则

样式属性不冲突时，可以将多个样式规则中的属性 层叠为一个

3、优先级

样式属性定义冲突时，按照不同样式表或样式规则的优先级来应用样式

浏览器缺省设置(UA) 低

外部样式表与内部样式表 中

-就近原则 ：后定义的优先

内联样式 高

4、!important 规则

1、作用

调整样式属性的优先级

2、用法

选择器{属性:值 !important;}

3、建议

少用

1、IE6 及一下了浏览器不支持

2、打破了原有优先级规则

5、排错

1、Unknown property name

属性名称 写错了

2、Invalid property value

属性值 写错

3、选择器写错

什么样式都没有

2、CSS语法(重点)

1、选择器

1、作用

规范页面中哪些元素能够使用定义好的样式

匹配页面元素

2、详解

1、通用选择器

1、作用

匹配页面所有的元素

2、语法

\*{}

2、元素选择器

1、作用

匹配 页面中指定的元素的元素

2、语法

元素名{}

div{}

p{}

h1{}

span{}

3、类选择器

1、作用

允许元素使用附带的class属性进行引用

2、语法

.类名{}

注意：不能以数字开始

引用

<标记 class="类名"></标记>

3、多类选择器

允许一个元素 引用 多个类选择器

<标记 class="类名1 类名2"></标记>

4、分类选择器

将类选择器和元素选择器结合使用，以便实现对某种元素中不同样式的细分控制

语法：元素选择器.类选择器{}

.redBack{} //匹配页面中所有class 为 redBack 的元素

div.redBack{}//匹配页面中所有class为redBack的div元素

4、id选择器

1、作用

只匹配页面指定id值得元素

2、语法

#idValue{}

3、ex

<div id="d1"></div>

<div id="d2"></div>

#d1{color:red;}

5、群组选择器

1、what

声明时，以 逗号 隔开的选择器列表

2、语法

选择器1,选择器2,选择器3,..{

}

6、后代选择器

1、作用

根据元素间的后代(出现在该元素中的所有元素)关系来匹配元素

2、语法

选择器1 选择器2{}

7、子代选择器

1、作用

根据元素间的子代(一层层级关系)关系来匹配元素

2、语法

选择器1>选择器2{}

8、伪类选择器

1、作用

匹配元素不同状态时的样式

2、分类

1、链接伪类

2、动态伪类

3、目标伪类

4、元素状态伪类

5、结构伪类

6、否定伪类

3、语法

:伪类{}

选择器:伪类选择器{}

1、链接伪类

1、:link 适用于尚未访问的超链接

2、visited 适用于访问过的超链接

2、动态伪类

1、:hover 适用于鼠标悬停在HTML元素时

2、:active 适用于元素被激活时的状态

3、:focus 适用于html元素获取焦点时的状态(text,password)

3、优先级

选择器 是通过 权值 表示优先级

元素选择器 1

类选择器 10

伪类选择器 10

ID选择器 100

内联样式 1000

选择器冲突时，将权值加到一起，大的优先。权值相同时，后定义优先

3、尺寸与边框

1、CSS单位

1、尺寸单位

% : 相对单位,属性值会占据父层容器对应属性的 百分比

<div style="width:500px;">

<p style="width:50%;"></p>

</div>

in : 英寸 1in=2.54cm

cm : 厘米

mm : 毫米

pt : 磅 point 1pt=1/72in,多数用于 字体大小

px : 像素(计算机屏幕上的一个点)

em : 1em当前字体尺寸，2em，当前字体尺寸2倍

注意：有些属性值是有单位的，单位不能省略。0除外。

2、颜色单位

1、rgb(r,g,b)

r:red 0-255

g:green 0-255

b:blue 0-255

color:rgb(0,0,0); 黑色

background-color:rgb(255,255,255); 白色

2、rgb(r%,g%,b%)

3、#rrggbb

由6位16进制的数字组成的一种颜色

每位由 0-9 A-F 组成

#000000 : 黑色

#111111 ：深灰色

#EEEEEE : 浅灰色

#FFFFFF : 白色

color:#E64916;

4、#rgb

简写方式，每两位数字相同时，可用该方式表示

#000000 -> #000;

#ff55ee -> #f5e;

#fe55cd 无替换

5、表示英文颜色的单词

6、rgba(r,g,b,alpha)

alpha : 透明度，0-1之间的数字

0 完全透明

1 完全不透明

rgba(255,0,0,0.5);

2、尺寸属性

1、作用

主要用于设置元素的宽和高

2、属性

1、宽度

width

max-width

min-width

2、高度

height

max-height

min-height

3、哪些元素可以设置尺寸属性

1、块级元素 可以设置

div,p,h1~h6,ul,ol

2、行内块

input

type = text ， password ...

3、行内元素

不能修改尺寸

4、本身具备 width 和 height 属性的 html元素 可以修改

<img><table>

4、溢出处理

1、什么是溢出

使用尺寸属性限制元素大小时，如果内容所需空间大于元素大小本身，会导致内容溢出

2、属性

overflow

overflow-x : 横向溢出处理

overflow-y : 纵向溢出处理

取值：

1、visible

默认值，溢出可见

2、hidden

溢出隐藏

3、scroll

右边，底边出现滚动条，溢出时可用

4、auto

自动，溢出时显示滚动条并且可用

3、边框属性

1、边框属性

1、简写方式

border:width style color;

width : 宽度 ，取值px的数值

style : 边框线条样式

取值：

1、solid ：实线

2、dotted ：虚线

3、dashed ：虚线

color : 边框线条颜色

作用：元素的 上下左右 四条边的 宽度，样式，颜色

border:1px solid black;

2、单边定义

1、语法

border-方向:width style color;

方向：

top : 上

right : 右

bottom : 下

left : 左

3、单属性定义

1、语法

border-属性:值;

属性：

1、width ：宽度

2、style ：样式

3、color ：颜色

2、作用

设置四条边框某一属性值

4、单边单属性定义

1、语法

border-方向-属性:值;

2、ex

border-top-color :上边框颜色

border-left-style:左边框样式

5、注意

1、边框颜色 可取值为 transparent(透明色)

2、边框特殊用法

border:none;

或

border:0;

2、边框倒角

1、什么是倒角

将边框四个直角 倒成 圆角

2、属性

border-radius

取值：

1个值 ：设置四个角的倒角半径

可以是 绝对数值 或 百分比

单角定义:

border-top-left-radius:左上角

border-top-right-raidus:右上角

border-bottom-right-radius:右下角

border-bottom-left-radius:左下角

3、边框阴影

1、属性

box-shadow

2、取值

是由多个值所组成的值列表，多个值之间用 空格 隔开

1、h-shadow : 必须，阴影的水平偏移距离，取值为正，向右偏，取值为负，则向左偏

2、v-shadow : 必须，阴影的垂直偏移距离，取值为正，向下偏，取值为负，则向上偏

3、blur : 可选，模糊距离

4、spread : 可选，阴影尺寸

5、color : 可选，阴影颜色

6、inset : 可选值，将默认的外部阴影改为内部阴影

3、常用模式

box-shadow:0 0 blur color;

4、轮廓

1 、属性

outline:width style color;

outline-width:

outline-style:

outline-color:

2、常用方式

outline:0;或outline:none;

# 第二天

1、框模型

1、框模型

1、框

页面元素皆为"框"

2、框模型

框模型(Box Model)定义了元素框处理元素内容、内边距、外边距以及边框的方式

width和height 指定内容区域的宽和高

增加了内外边距、边框 不会影响内容区域的尺寸，有可能会增加元素框的总尺寸

元素实际宽度=左右外边距 + 左右边框 +左右内边距 + width;

元素实际高度=上下外边距 + 上下边框 +上下内边距 + height;

2、外边距

1、什么是外边距

围绕在元素边框周围的空白区域，默认是不允许被其他元素所占据的。

2、作用

主要拉开元素与元素之间的距离

3、语法

1、属性

margin:value;

单边设置

margin-top/right/bottom/left:value;

取值单位：

1、px 值

2、%

3、auto ：左右方向有效

4、负值 ：目的为移动元素

2、取负值

让元素向反方向移动

通常只设置

margin-top : 负值

margin-left: 负值

3、auto

能够自动计算外边距(左右)

特点：控制非 行内元素 水平居中时使用

前提：必须设置元素宽度

4、外边距的简便写法

margin:value;(四个方向外边距相同)

margin:v1 v2;v1 上下外边距 v2 左右外边距

margin:v1 v2 v3;v1 上外边距，v2 左右外边距，v3 下外边距

margin:v1 v2 v3 v4;上 右 下 左

5、有默认外边距的元素

body,h1,h2,h3,h4,h5,h6,p,ul,ol,pre,dl,dd

具备默认外边距

通过 css重写(CSS Reset) 方式，改变默认外边距为 0

6、特殊情况

1、外边距合并

当两个垂直外边距相遇时，它们将形成一个外边距

合并后的外边距的高度等于两个外边距中高度较大者的值

2、外边距溢出

子债父还

在某些特定情况下，给子元素设置外边距时，效果会作用在父元素上。

特定情况：

1、上下外边距 能产生溢出

2、父元素无边框，并且设置的是第一个元素的上外边距或最后一个元素的下外边距

解决方案：

1、为父元素增加边框

2、由父元素的内边距来取代子元素的外边距

4、注意

上下外边距 对行内元素 是不起作用的

上下外边距 对行内块元素 有特殊效果

3、内边距

1、什么是内边距

内容区域与边框之间的距离

会扩大元素边框所占的区域

2、语法

1、属性

padding:value;

取值 可以为 像素px，百分比%

padding-top/right/bottom/left:val;

2、简写方式

padding:value;(四个方向内边距)

padding:v1 v2;(上下 左右)

padding:v1 v2 v3;(上 左右 下)

padding:v1 v2 v3 v4(上右下左)

3、注意

内边距的效果只能影响元素本身，并不会给其他元素带来影响效果

2、背景属性

1、作用

背景颜色以及背景图像

注意：背景是从边框的位置处开始绘制的

2、背景属性

1、背景色

语法：background-color:value;

取值：

任意合法颜色值

transparent

2、背景图片

语法:background-image

取值:url(图片URL);

3、背景重复

语法:background-repeat

取值:

1、repeat ：默认值，即垂直又水平方向的重复

2、no-repeat : 不重复

3、repeat-x : 仅在水平方向重复

4、repeat-y ：仅在垂直方向重复

4、背景图片尺寸

语法：background-size

取值：

1、value1 value2

宽度 高度

2、value1% value2%

百分比

3、cover

覆盖，将背景图像等比缩放，直到背景能够覆盖到元素的所有区域

4、contain

包含，将背景图像等比缩放，直到背景图像碰到右边或下边时，停止缩放。

5、背景图片固定

语法：background-attachment

取值：

1、scroll，默认值，背景图会随着文档而滚动

2、fixed，背景图像固定，背景图不会随着文档而发生滚动

6、背景定位

1、作用

改变背景图像在元素中的位置

2、语法

属性：background-position

取值：

1、x y

x 表示 水平偏移距离,取值为正，图像向右偏移，取值为负，图像向左偏移

y 表示 垂直偏移距离，取值为正，图像向下偏移，取值为负，图像向上偏移

2、x% y%

0% 0% : 原始位置

100% 100% :右下角

50% 50% : 正中间

3、关键字

x ：left center right

y ：top center bottom

3、CSS Sprites

CSS精灵

将若干幅小图像 拼合成 一幅大图像

7、背景属性

属性：background

取值：color url() repeat attachment position;

ex

background:red url(a.jpg) no-repeat fixed center center;

常用：

background:url() no-repeat position;

background:red;

3、渐变

1 、作用

多个颜色之间，逐渐变换的效果

2、渐变分类

1、线性渐变

2、径向渐变

3、重复渐变

1、重复线性

2、重复径向、

3、重要元素

1、色标

每种颜色值 及其 出现的位置

4、渐变语法

1、属性

background-image

取值：

1、linear-gradient() : 线性渐变

2、radial-gradient() : 径向渐变

3、repeating-linear-gradient()

重复线性渐变

4、repeating-radial-gradient()

重复径向渐变

2、线性渐变

1、语法：

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

1、angle

方向 或 角度

取值：

1、to top --> 0deg

从下向上填充颜色

2、to right --> 90deg

从左向右填充颜色

3、to bottom --> 180deg

从上向下填充颜色

4、to left --> 270deg

从右向左填充颜色

2、color-point

color position

ex:

red 0px

green 50%

注意：省略 位置

3、径向渐变

1、语法

background-image:radial-gradient([size at position],color-point,color-point);

1、size at position

size ：半径大小

position ：圆心所在位置

0px 0px：圆心在元素左上角

50% 50%: 圆心在元素中间处

left bottom:圆心在元素坐下角

ex:50px at center center

4、重复渐变

repeating-linear-gradient();

repeating-radial-gradient();

注意：colo-point 颜色的位置 必须给 px为单位的数值。

repeating-linear-gradient(to top,red 0px,yellow 20px,blue 50px);

5、浏览器兼容性

主流浏览器的新版本 肯定支持渐变

不支持的版本，尝试着加浏览器前缀

Firefox : -moz-

Chrome & Safari : -webkit-

Opear : -o-

IE : -ms-

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

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

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

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

# 第三天

1、文本格式化属性

1、文本格式化

字体样式：加粗，斜体 ，大小

文本格式：排列方式，缩进，颜色

建议：尽可能使用 css 属性 取代 html 标记

2、控制字体

1、字体系列

属性：font-family

取值：value1,value2,....,valuen;

注意：字体值中包含特殊符号(空格)、中文时，需要用 "" 引起来

ex ：font-family:"microsoft yahei",Arial,simsun;

2、字体大小

属性：font-size

取值：px 或 pt 或 em

3、字体加粗

属性：font-weight

取值：

1、normal 正常

除 i ，strong，h1~h6 之外 ，其他都是 normal

2、bold

加粗

3、value

无单位的数值

100~1200

常用值：400 ~ 900

4、字体样式

1、作用

斜体 --> <i></i>

2、属性

font-style

3、取值

1、normal 正常

2、italic 斜体

5、小型大写字母显示

Aa Bb Cc Ee

1、作用

英文，按照大写字母显示，个头和小写字母一样高

2、属性

font-variant

3、取值

1、normal

2、small-caps

6、字体属性 font

font:style variant weight size family;

注意：font 属性值中，如果没有 family 的值， 则无效

font:12px; 无效

font:12px "微软雅黑"; 有效

3、文本属性

1、文本颜色

color

2、文本排列

文本的水平对齐方式

1、属性

text-align

2、取值

left/center/right/justify

3、文字修饰(线条)

1、属性

text-decoration

2、取值

1、none

无线条样式

除 a、s、u 其余都是none

2、underline

下划线

3、overline

上划线

4、line-through

删除线

4、首行文本缩进

1、属性

text-indent

2、取值

以 px 为单位的数值

5、文本阴影

1、属性

text-shadow

2、取值

h-shadow v-shadow blur color;

6、行高

1、作用

设置一行文本的高度。

如果 行高的高度值 大于 文本本身的高度，那么 该行文本将在指定的行高高度内 呈现出 垂直居中的效果

2、表格属性

1、表格常用属性

1、内边距属性 ： padding

2、尺寸属性 ：width height

3、文本格式化/字体属性

font-\*

text-\*

4、背景属性

5、border属性

6、vertical-align

1、允许使用 vertical-align属性的元素

1、table td

2、img

3、行内块

2、作用

垂直对齐方式

3、取值

top/middle/bottom

2、表格特有属性

1、边框合并属性

属性：border-collapse

取值：

1、separate

默认值，分离边框

2、collapse

边框合并

2、边框边距

属性：border-spacing

作用：设置相邻单元格的边框间的距离

要求：border-collapse 必须为 separate

取值：

1、取一个数值 ：水平和垂直间距相同

2、取两个数值 ：第一个值 表示水平间距，第二个值 表示垂直间距

两个值之间用 空格 隔开

3、标题位置

<caption></caption>

属性：caption-side

取值：

1、top，默认值

2、bottom

4、显示规则

1、作用

规定 单元格 的宽度 和 高度的处理方式

2、属性

table-layout

取值：

1、auto

默认值，自动表格布局，单元格宽度由单元格内容决定

2、fixed

固定表格布局。单元格宽度 由表格和单元格设置的值为主。

3、对比

1、自动表格布局

table-layout:auto

1、单元格大小会适应内容的大小

2、加载复杂的表格式比较慢

3、适用于不确定每列大小时使用

2、固定表格布局

table-layout:fixed

1、列宽度取决于表格宽度，列宽度，与单元格内容无关

2、会加速显示表格

固定表格布局 ，算法较快，不够灵活

自动表格布局 ，算法较慢，传统表格

3、浮动(重点)

1、定位概述

1、什么是定位

元素应该出现在网页中的哪个位置处

2、分类

1、普通流定位

2、浮动定位

3、相对定位

4、绝对定位

5、固定定位

3、普通流定位

1、what

又称为 文档流定位

每个块级：占用一行的空间，并且上下排列

每个行内：从左到右排列，不单独占一行

2、浮动定位

1、什么是浮动 & 特点

1、会将元素排除在文档流之外-脱离文档流

2、元素不占据页面空间

3、其余在文档流中的元素要上前补位

4、浮动元素会停靠在父元素的左边或右边，或者停靠在已经浮动起来的元素的边缘上

2、处理问题(作用)

1、让块级元素在一行内显示

3、属性

属性：float

取值：

1、none

默认值，即没有浮动效果

2、left

左浮动

3、right

右浮动

4、特殊处理

1、如果父元素的宽度已经装不下所有的已浮动子元素，那么最后一个将会换行

2、浮动元素如果不设置宽度的话，那么元素的宽度将自适应

3、所有的元素浮动起来都将变成块级元素

块级元素：允许修改 尺寸

4、浮动元素 对 文字、行内元素、行内块元素的影响

5、清除浮动

1、属性

clear

作用：清除左右两端浮动元素给当前元素带来的影响

取值：

1、left

清除 当前元素 左边的浮动元素带来的影响

2、right

清除 当前元素 右边的浮动元素带来的影响

3、both

清除 当前元素 两边的浮动元素带来的影响

4、none

默认，不清除

6、浮动元素对父元素高度所带来的影响

1、父元素浮动

2、设置父元素的高度

3、设置父元素 overflow 值为 auto 或 hidden

4、父元素中，追加一个 空的 块级 子元素，并且 设置其 clear 属性为 both

# 第四天

1、显示

1、显示方式

1、什么是显示方式

决定了元素在页面中如何来排列

2、属性

display

取值：

1、none

隐藏，让生成的元素不显示

特点：脱离文档流

2、block

让 元素 变成块级

3、inline

让 元素 变成行内

注意：以后不要把块级元素改变成行内

4、inline-block

让 元素 变成 行内块

位置摆放如同行内元素，但是会具备块级元素的特点(允许修改宽和高)

2、显示效果

1、visibility 属性

作用：控制元素可见性

取值：

1、visible

默认值，显示的

2、hidden

隐藏，占据页面空间

3、collapse

用在表格元素。动态删除一行或一列时，不会影响表格整体布局

display:none 与 visibility:hidden 的区别

1、display属性时控制显示方式

display:none;让生成的元素不显示，并且不占据页面空间,脱离文档流

2、visibility

专门改变元素可见性，取值为hidden的时候，会隐藏元素，但是不会脱离文档流

2、opacity属性

作用：改变元素透明度

取值：0.0(完全透明) - 1.0(完全不透明) 之间的数字

3、vertical-align

1、作用

垂直对齐方式

2、允许作用的元素

1、td

2、行内块中允许使用

允许设置 行内块，img ，左右两端文本 的垂直对齐方式

取值：

1、baseline

默认值，基线对齐

2、top/middle/bottom

Aa Bb Cc Jj Gg Yy

3、光标

属性：cursor

取值：

1、default

2、pointer

小手

3、crosshair

+

4、text

I

5、wait

等待

6、help

?

2、列表属性

1、列表项标志

属性：list-style-type

取值：

none

disc

circle

square

2、列表项图像

属性：list-style-image

取值：url()

3、列表项位置

属性：list-style-position

取值：

1、outside

默认值，列表项标志位于内容区域之外

2、inside

将标志放在内容区域中

4、列表属性 - list-style

取值：type url() position;

常用方式：list-style:none;

5、列表使用场合

1、从上到下，从左到右，并且具备相同特征的数据，全部使用列表。

3、定位

1、属性

1、position

作用：规定元素的定位类型

取值：

static : 默认值，静态

relative : 相对定位

absolute : 绝对定位

fixed : 固定定位

2、偏移属性

作用：配合 上边 三种定位方式，实现元素的位置移动

top:上偏移

bottom:下偏移

left:左偏移

right:右偏移

3、堆叠顺序

属性：

z-index

只有 relative,absolute,fixed的时候才能使用

2、定位-相对定位

1、什么是相对定位

相对于元素原来的位置偏移某个距离

2、设置相对定位

position:relative

通过 偏移属性 完成位置的移动

3、定位-绝对定位

1、什么是绝对定位

1、脱离文档流

不占据页面空间

2、相对于 离它最近的已定位的祖先元素去实现定位

最近：

已定位：relative，absolute，fixed

祖先：无限的父层元素

3、如果没有 已定位的祖先元素，那么它的位置就相对于最初的包含框(body或html)实现定位

2、属性

position:absolute;

通过 偏移属性 实现位置移动

3、相关操作

1、元素修改为 绝对定位后 将变成块级元素

2、margin

margin 可以用，但 auto 会失效

4、堆叠顺序

属性：z-index

取值：无单位的数字，值越大越考上

注意：

父子关系没办法使用 z-index 调整，永远都是 子压在父上

只有已定位元素才能使用relative,absolute,fixed

5、固定定位

1、作用

将内容固定在页面的某个位置处

2、属性

position:fixed

通过 偏移属性 实现位置移动

3、注意

固定定位会脱离文档流