

WEB前端系统学习——课程笔记

课程链接：[WEB前端系统学习 - 网易云课堂](#)

笔记链接：[Web-Learn-Note](#)

一、课程目录

二、各章节笔记

一、课程目录

章节1 前端的认知

章节2 CSS引入

章节3 CSS盒模型

章节4 文本样式及样式继承

章节5 常见标签

章节6 浮动样式和特性

章节7 浏览器兼容性

章节8 表格与表单

章节9 整站规划

章节10 CSS3常用图片处理

章节11 CSS盒阴影及文字操作

章节12 媒体查询

章节13 HTML新标签使用及CSS选择器

章节14 服务器字体及弹性盒模型

章节15 CSS3动画过渡

章节16 JS认知

章节17 JS属性操作

章节18 For循环

章节19 数据

章节20 传参

章节21 变量作用域

章节22 函数

章节23 JS动画

章节24 JS之DOM

章节25 JS事件

章节26 JS正则表达式

章节27 JS面向对象

章节28 jQuery

章节29 WEB前端框架—bootstrap

二、各章节笔记

章节1 前端的认知

- 课时1 Html5及行业介绍 23:07
- 课时2 课程介绍及工具安装PS使用 27:02

Sublime Text3安装、相关使用插件介绍
PS安装

- 课时3 HTML文档的结构 14:23

章节2 CSS引入

- 课时4 CSS引入之行间样式与内部样式、标签名 21:10

1. 行间样式的引入

写法：在标签中，写一个style的属性 比如：style="....." 在引号当中，写相应的css样式

缺点：不利于维护，不利于代码重用

优点：优先级最高

2. 内部样式表的引入

写法：在head标签里，写一个style的标签，在标签中，通过选择器来控制样式选择器？

标签名选择器

写法：

```
div {  
    ... css的样式  
}
```

ID选择器

写法:

首先在相应标签中设置一个ID的属性

如: `id="id名"`

在样式表中, 通过

```
# + id名 {  
    ... css的样式  
}
```

温馨小提示:

id名要以英文字母开头

id名不能重复, 是唯一的

class选择器

写法:

首先在相应标签中设置一个class的属性

如: `class="class名"`

在样式表中, 通过

```
. + class名 {  
    ... css的样式  
}
```

温馨小提示

class名以英文字母开头

优先级

标签名选择器 < class选择器 < id选择器 < 行间样式

优点: 加载速度快, 不需要去请求服务器

缺点: 不利于代码重用

3. 外部样式表的引入

写法: 在head标签里, 写一个link标签, 用来关联一个css的文件, 在css文件中, 通过选择器来控制样式

温馨提示.

```
<link rel="stylesheet" href="css文件的路径">
```

优点: 利于代码重用

缺点: 需要加载服务器

- 课时5 class选择器和外部样式表, 选择器优先 19:18

标签名选择器 < class选择器 < id选择器 < 行间样式

- 课时6 css宽、高、背景设置 26:35

width 宽
height 高
background 背景
 背景色
 background-color: 颜色值 英文单词 十六进制 **rgb**
 背景图
 background-image: **url**("图片路径")
 背景图平铺
 background-repeat: **no-repeat** **repeat-x** **repeat-y**
 背景图定位
 background-position: 第一个值(X轴的位置) 第二个值(Y轴的位置);
 第一个值: **left** **center** **right** **30px**;
 第二个值: **top** **center** **bottom** **100px**;

复合写法
background: **color** **image** **position** **repeat**;

背景图滚动
background-attachment: **fixed**;

背景图尺寸
background-size: 第一个值(X轴的比例) 第二个值(Y轴的比例);

温馨提示: **css3**的样式 不兼容

章节3 CSS盒模型

- 课时7 **css**的基础样式-**border** 17:20

border 边框
复合写法
border: **border-width** **border-style** **border-color**;

border-width 边框宽度
border-style 边框样式 **solid** 实线 **dashed** 虚线 **dotted** 点线 (点线IE6不认)
border-color 边框颜色 英文单词 十六进制 **rgb**

border-radius 圆角

透明 **transparent**

- 课时8 **css**的基础样式-**padding** 18:08

padding 内边距 内填充

padding-top 上内边距

`padding-right` 右内边距
`padding-bottom` 下内边距
`padding-left` 左内边距

`padding:10px;` 一个值代表 上右下左都是10px
`padding:10px 20px;` 两个值 第一个值代表上下 第二个值代表左右
`padding:10px 20px 40px;` 三个值 第一个值代表上 第二个值代表左右 第三个值代表下
`padding:10px 20px 40px 80px;` 4个值 第一个值代表上 第二个值代表右 第三个值代表下 第四个值代表左

- 课时9 css的基础样式-margin 18:33

`margin` 外边距

`margin-top` 上外边距
`margin-right` 右外边距
`margin-bottom` 下外边距
`margin-left` 左外边距

`margin`的复合写法

`margin:10px;` 一个值代表 上右下左都是10px
`margin:10px 20px;` 两个值 第一个值代表上下 第二个值代表左右
`margin:10px 20px 40px;` 三个值 第一个值代表上 第二个值代表左右 第三个值代表下
`margin:10px 20px 40px 80px;` 4个值 第一个值代表上 第二个值代表右 第三个值代表下 第四个值代表左

- 课时10 盒模型 14:02

空间大小 :

盒子的总宽度 : `border-left + padding-left + width + padding-right + border-right`

盒子的总高度 : `border-top + padding-top + height + padding-bottom + border-bottom`

位置

`margin`

章节4 文本样式及样式继承

- 课时11 文本样式上 17:19
- 课时12 文本样式中 15:47
- 课时13 文本样式下 29:20

line-height、text-align、text-indent、font-weight、font-style、decoration、letter-spacing、word-spacing、font.....

章节5 常见标签

- 课时14 常见标签 语义 title div h 权重 20:39

语义：你说的话到底是什么意思

权重：优先先考虑谁

title标签

写法:<title></title>

双标签

语义：网页标题

权重最大

div标签

写法: <div></div>

双标签

语义：无意义

默认样式：无

h标签

h1~h6

写法：

<h1></h1>

<h2></h2>

<h3></h3>

<h4></h4>

<h5></h5>

<h6></h6>

双标签

语义：标题

默认样式：font-size font-weight margin

权重：h1 最大 h2 次之 h3更次之。。。。。。。

h1页面中最好不要超过一个

h2页面中最好不要超过12个

h3之后的h标签就随便了

样式初始化文件 reset.css

群组选择器，例：

```
h1,h2,h3,h4,h5,h6{  
    margin:0;  
}
```

- 课时15 语义p ul li ol 22:44

p标签

写法: `<p></p>`

双标签

语义: 段落

默认样式: `margin`

p标签在嵌套时不要包含块元素的标签

ul标签

写法: ``

双标签

语义: 无序列表

默认样式: `margin padding-left list-style-type`

li标签

写法: ``

双标签

语义: 列表项

默认样式: `list-style-type`

ol标签

写法: ``

双标签

语义: 有序列表

默认样式: `margin padding-left list-style-type`

ul,ol和li 是组合标签

ul,ol的第一层子级一定是li li的父级一定是ul或ol

```
<ul>
  <li>
    <h2></h2>
  </li>
</ul>

<div>
  <ul>
    <li></li>
  </ul>
</div>

<ol>
  <li>
    <ul>
      <li></li>
    </ul>
  </li>
</ol>
```

```
<div>
  <ol>
    <li></li>
  </ol>
</div>
```

- 课时16 dl dt dd 块属性标签特性 14:40

dl标签

自定义列表 字典标签

dl dt dd

写法: <dl></dl>

双标签

语义: 自定义列表

默认样式: margin

dt

写法: <dt></dt>

双标签

语义: 列表标题

默认样式: 无

dd

写法: <dd></dd>

双标签

语义: 列表说明

默认样式: margin-left

块属性的标签 特性:

1. 默认是父级100%的宽
2. 支持所有CSS样式
3. 独占一行

- 课时17 span 内联属性标签特性 23:08

内联属性的标签

内联标签

特性:

1. 同属性的标签排在同一排
2. 内容撑开宽度
3. 不支持宽高, 不支持上下的padding和margin
4. 代码换行被解析 间距大小取决于父级的font-size的大小

span标签

写法:

双标签
语义：无意义
默认样式：无

strong标签

写法：
双标签
语义：强调
默认样式：font-weight
权重比较高

em标签

写法：
双标签
语义：强调
默认样式：font-style
权重比较高

a标签

写法：

双标签
语义：超链接
默认样式：color text-decoration cursor
温馨提示：

href属性中

1. 如果要链接一个网站地址的话，

2. # 代表链接到当前页

target

_self 默认 在当前窗口打开
_blank 在新窗口打开

一定要加 http://

内联块标签

内联块属性

特性：

1. 同属性的标签排在同一排
2. 内容撑开宽度
3. 支持所有的css的样式
4. 代码换行被解析，间距大小取决于父级的font-size

大小

img标签

写法：
单标签
语义：图片
默认样式：低版本浏览器中有默认的边框

温馨小提示：

1. 路径地址一定要正确
2. alt属性一定要写

- 课时18 内联块特性 img, 标签类型转换 23:34

标签类型的转换

显示为无: `display:none;`

块属性 `display:block;`

内联属性

内联 `display:inline;`

内联块 `display:inline-block;`

章节6 浮动样式和特性

- 课时19 浮动样式和特性介绍 25:51

`display:inline-block;`

浮动 `float`

浮动的特性

- 1.浮动的元素排在同一排
- 2.浮动的元素内容撑开宽度
- 3.浮动的元素支持所有的css样式
- 4.浮动的元素脱离文档流
- 5.浮动的元素提升层级半级

- 课时20 BFC常见方法及优缺点 24:55

1.overflow 溢出

会重新计算元素的空间

`hidden` 溢出隐藏

`auto` 溢出出现滚动条

`scroll` 出现滚动条的位置

2.元素的居中

`margin:0 auto;`

3.元素的伪类

伪类: 就是CSS一些元素身上的特殊属性

`:hover` 鼠标停留

`:after` 在元素内容之后插入一些内容

浮动: 其实就是使元素脱离文档流, 按照一定的方式排列, 遇到相邻的浮动元素或者父级的边界停下来。

BFC 就是清浮动 就是用来处理浮动元素脱离文档流的问题

1.父级也浮动

弊端: 左右的`margin:0 auto;` 会失效;

2.父级加`display:inline-block`

弊端：左右的margin:0 auto; 会失效;
(如果需要让元素居中可以给父级加text-align:center)

3. 给父级加高

弊端：扩展性不好

4. br 标签

写法：

作用：换行

5. clear

clear 元素的某一方向不允许出现其他的浮动元素

left

right

both 推荐使用

one

温馨小提示

不符合W3C规范 违反结构 样式 行为 三者分离原则

6. 伪类清除浮动

```
:after{
    content:"";display:block;clear:both;
}
```

温馨提示：目前主流方法。建议使用

- 课时21 BFC常见方法及优缺点 提升层级 21:16
- 课时22 定位特性相对定位 绝对定位 固定定位 24:08

定位： 把一个元素 按照一定的方式 定到页面的某一个位置
position

相对定位 relative

针对自己本身的位置进行定位

绝对定位 absolute

针对有定位的父级的原点进行定位，如果父级没有定位，会找父级的父级身上有没有定位，如果有，针对父级的父级的原点进行定位，以此类推，如果都没有定位，针对document进行定位

温馨小提示：

绝对定位即使没有初始值，也一定要设置值

left:0px;

top:0px;

固定定位 fixed

针对页面窗口进行定位

温馨提示：

IE6 不支持固定定位

偏移量

left

top

right

bottom

温馨提示

left top 比 **right bottom** 的优先级要高

- 课时23 三种定位的特性 15:46

相对定位 **relative**

1. 不影响元素本身的特性
2. 不使元素脱离文档流
3. 提升层级
4. 无法触发BFC
5. 针对自己本身进行定位

绝对定位 **absolute**

1. 会使元素完全脱离文档流
2. 内容撑开宽度和高度
3. 使元素支持所有的CSS样式
4. 提升层级
 - z-index**: 数值; 定位层级设置
 - 数值越大, 层级越高
5. 绝对定位要和相对定位配合使用
6. 如果有定位父级, 针对定位父级发生偏移, 没有定位父级, 针对document进行偏移
7. 如果绝对定位的子级有浮动, 可以省略清浮动的操作

固定定位 **fixed**

1. 不兼容IE6
2. 针对窗口进行定位
3. 如果固定定位的子级有浮动, 可以省略清浮动的操作

- 课时24 定位特性的应用及派生选择器的应用 24:09

派生选择器是由 **id选择器** **class选择器** **标签名选择器** 组合而成

```
#box2 span {  
    .....  
}
```

章节7 浏览器兼容性

- 课时25 兼容性1 17:11
- 课时26 兼容性2 16:36
- 课时27 兼容性3 18:31
- 课时28 兼容性4 20:50
- 课时29 兼容性5 31:30

兼容性：页面在不同的浏览器中可能会显示不同。

在IE6下，子级的宽度会撑开父级设置好的宽度

温馨小提示：盒模型的计算一定要精确，否则IE浏览器可能会显示不同

ie6中，元素浮动，如果宽度需要内容撑开，需要给里面的块元素都添加浮动才可以

在IE6.7下 元素要通过浮动排在同一排，就需要给这行元素都加浮动

在IE6下元素的高度如果小于19px的时候，会被当成19px来处理

解决方法：overflow:hidden;

在IE6下不支持1px的dotted边框样式

解决方法：切背景平铺

在ie6下，块元素有浮动有横向的margin的值的时候，横向的margin的值会扩大两倍

margin-left 一行中左侧的第一个元素有双边距

margin-right 一行中右侧的第一个元素有双边距

在IE6,7下，li本身没有浮动，li里面的内容有浮动，li下会产生一个间隙

解决方法：1.给li加浮动 2.给li加vertical-align:top

在IE6,7下，li本身没有浮动，li里面的内容有浮动，li下会产生一个间隙

解决方法：

1.给li加浮动

2.给li加vertical-align:top

在IE6下最小高度的bug 和 li的间隙问题共存的时候，给li加浮动，vertical-align不好使

当一行子级元素宽度之和和父级的宽度相差超过3px，或者子级元素不满行的情况的时候，最后一行的子级元素的margin-bottom会失效

在ie6下的文字溢出bug

子元素的宽度和父级的宽度如果相差小于3px的时候，两个浮动元素中间有注释或者内联元素，就会出现文字溢出，内联元素越多，溢出越多

解决办法：用div把注释或者内联元素包起来

在IE6下，当浮动元素和绝对定位元素是兄弟关系的时候，绝对定位会失效

解决方法：

不让浮动元素和绝对定位元素是兄弟关系，用div或者其他标签把a标签包起来

在IE6.7下，子级元素有相对定位，父级overflow包不住子级元素

解决方法：给父级也加相对定位

在IE6下，如果绝对定位的父级宽高是奇数的时候，子级元素的right和bottom值会有1px的偏差

在IE6.7下 输入型的表单标签控件上下会有1px的间隙。

处理方法：给input加浮动

```
css hack:
  \9 ie10之前的ie浏览器解析的代码
+或者* 表示ie7包括7之前的ie浏览器
_表示 ie6包括6之前的ie浏览器
```

章节8 表格与表单

- 课时30 表格嵌套规则 样式初始化 14:04

```
<table></table> 表格
<thead></thead> 表格的头部
<tbody></tbody> 表格的身体
<tfoot></tfoot> 表格的尾部
<tr></tr>        表格行数
<th></th>        表格的标题
<td></td>        表格的单元格

table {
  border:1px solid red;
  border-collapse: collapse;
}
td,th {
  border:1px solid red;
  padding:0;
}
```

- 课时31 表格标签特性 单元格合并 20:13

table标签的特性

1. 单元格会默认平分整个**table**的宽度
2. **th**的内容默认加粗，并且水平和垂直都居中
3. **td**的内容默认不加粗，并且垂直居中
4. **table**的宽度决定整个表格的宽度，如果**td**，**th**设置的总宽度比**table**的宽度大，不好使
5. 表格的同一列继承最大宽度 （**th**,**td**最好都设置上宽度）
6. 表格的同一排继承最大高度 （**th**,**td**最好都设置上高度）

单元格合并

```
colspan="数值"
rowspan="数值"
```

数值对应需要合并的单元格的数量，

温馨提示：

被合并的单元格要删除，否则会多出一个单元格

- 课时32 表单 表单控件input类型 29:22

```

form
    action="需要传送到地址"
    method="传送的方式"
        get 在地址栏做内容的拼接，用户可以看到的
        post 把表单元素里面的内容做拼接，并且打包传送的action的地址，用户看不到
    target
        _blank 在新窗口打开
        -self 在当前窗口打开

input的type类型
    文本 text
    密码 password
    radio 单选
    checkbox 多选
    file 文件上传
    button 按钮
    hidden 隐藏
    reset 重置
    submit 提交

温馨提示：
    input里面标签的值都体现在value中
    name属性一定要设置，如果不设置，无法实现上传

```

- 课时33 表单其他控件 样式初始化 09:35

```
select
```

章节9 整站规划

- 课时34 服务器 路径 命名 19:32

服务器：可以提供服务的机器 可以提供数据服务的机器

命名法

驼峰式命名

```
id="mainLeft"
```

```
id="sideBar"
```

下划线 php

```
id="dl_header"
```

中划线 命名法

```
id="dl-header"
```

(1)页面结构

- 容器: container
- 页头: header
- 内容: content/container
- 页面主体: main
- 页尾: footer
- 导航: nav
- 侧栏: sidebar
- 栏目: column
- 页面外围控制整体布局宽度: wrapper
- 左右中: left right center

(2)导航

- 导航: nav
- 主导航: mainbav
- 子导航: subnav
- 顶导航: topnav
- 边导航: sidebar
- 左导航: leftsidebar
- 右导航: rightsidebar
- 菜单: menu
- 子菜单: submenu
- 标题: title
- 摘要: summary

(3)功能

- 标志: logo
- 广告: banner
- 登陆: login
- 登录条: loginbar
- 注册: regsiter
- 搜索: search
- 功能区: shop
- 标题: title
- 加入: joinus
- 状态: status
- 按钮: btn
- 滚动: scroll
- 标签页: tab
- 文章列表: list
- 提示信息: msg
- 当前的: current
- 小技巧: tips
- 图标: icon
- 注释: note
- 指南: guild
- 服务: service
- 热点: hot
- 新闻: news
- 下载: download
- 投票: vote
- 合作伙伴: partner

- 友情链接: `link`
- 版权: `copyright`

- 课时35 网站加载速度和用户流失比率 13:07

网站的加载速度

文件大小

`CSS`

提取公共样式 把相同的`CSS`样式提出 尽量重用 减少代码量

尽量用复合样式减少代码量

尽量发现公共样式 使代码重用减少代码量

相同样式用群组选择器 减少代码量

`CSS`的维护

划分好区域 方便维护代码

文件数量

图片

- 课时36 背景图合并 整站思路 26:37

图片的命名不要用中文，尽量用英文或者拼音 不要随意加数字 不要大小写切换 要么全大写 要么全小写

章节10 CSS3常用图片处理

- 课时37 圆角 多背景 14:19

`border-radius` 圆角

分为水平半径和垂直半径 详情请见 `img`文件夹中`radius.png`图

水平半径和垂直半径中间用 `/` 分开

`/`左边代表水平半径

`/`右边代表垂直半径

多背景

在`CSS3`中，可以设置多背景，中间通过 `,` 隔开，不只可以设置基础样式，还可以设置背景尺寸，同样用 `,` 隔开

- 课时38 背景尺寸 19:41

背景大小

`background-size` 设置背景图像的大小

1. 具体的数值

100px 200px 第一个值表示X轴的大小（宽度） 第二个值表示Y轴的大小（高度）如果只设置了一个值，表示X轴的背景大小（宽度），第二个值会自动缩放或者拉伸

2. 百分比

80% 100% 第一个值表示X轴针对自己本身盒子的宽度的比例（宽度） 第二个值表示Y轴针对自己本身盒子高度的比例的大小（高度）如果只设置了一个值，表示X轴针对自己本身盒子的宽度的比例（宽度），第二个值会自动缩放或者拉伸

3. cover

把背景图尽量扩展至最大，使背景图覆盖整个盒子区域，至于X轴和Y轴是否超出，不予考虑

4. contain

把背景图尽量扩展至最大，使背景图尽量显示完整，当X轴或者Y轴其中之一贴近盒子边缘，另一方停下来，至于X轴和Y轴是否有填不满的情况，不予考虑

在CSS3中，所有的颜色值都可以用rgba的值来写，方法：**border:50px solid rgba(255,255,255,0.9);**

a表示透明度，数值在0~1之间，数值越大，越不透明

- 课时39 背景裁切 浏览器内核前缀 背景裁切 22:44

背景裁切: **background-clip**

1. border-box

从边框开始显示背景

2. padding-box

从padding区域开始显示背景

3. content-box

从内容区域显示背景

4. text

火狐不支持 ie9,10不支持
从文本区域显示背景

温馨提示:

需要rgba的配合 （需要rgba的透明度进行配合）

浏览器内核前缀

w3c组织

```
-webkit-transition: 1s; chrome safari ....
-moz-transition: 1s;      火狐
-ms-transition: 1s;       ie 360
-o-transition: 1s;        Opera
transition: 1s;           w3c标准样式
```

- 课时40 线型渐变 20:24
- 课时41 线型渐变笔记 16:56

线型渐变 ie9以下不支持 渐变属于背景下的一个属性

方法 `linear-gradient()`;

写法:

比如: `background: linear-gradient(90deg,red,yellow,green,blue);`

括号里的第一个值,是角度

第一个值可以模糊的值,也可以是具体的数值

模糊的值:

`to right`
`to left`
`to top`
`to bottom`

具体的角度数值

角度是指水平线和渐变线之间的角度,按逆时针方向计算

比如:

标准

`0deg` 从下到上

`90deg` 从左到右

非标准

`0deg` 从左到右

`90deg` 从下到上

标准和非标准的换算公式

$$90 - x = y$$

x为标准角度

y为非标准角度

(温馨提示: 参考图: 渐变角度参考图)

从第二个值开始是颜色值,可以通过具体的数值确定颜色值渐变的区域,比如

```
background: linear-gradient(90deg,  
    red 100px,  
    yellow 180px,  
    green 280px,  
    blue  
);
```

(温馨提示: 见图: 颜色值区域参考图)

章节11 CSS盒阴影及文字操作

- 课时42 径向渐变 重复渐变 19:54

径向渐变 ie9以下不支持

渐变都是背景下一个属性

`background: radial-gradient(at center,red,yellow,blue);`

径向渐变是从圆心往进行渐变

括号里的第一个值，是圆心位置

可以使模糊的值，也可以是具体的数值

模糊的值

`at left`

`at center`

`at right`

`at top`

`at bottom`

具体的数值

`at 100px 200px`

`100px`为X轴的偏移量

`200px`为Y轴的偏移量

从第二个值开始，是颜色值，颜色值可以无限的写

颜色值，可以通过具体的数值来规定，渐变区域或者范围

具体的数值可以是

1, `100px`

2, 百分比

重复渐变

```
background: repeating-linear-gradient(to right, red 10%, yellow 20%, blue 30%);
```

```
background: repeating-radial-gradient(at center, red 10%, yellow 20%, blue 30%);
```

• 课时43 盒阴影 文字阴影 18:36

盒阴影

`box-shadow`

第一个值：默认省略或者`inset` 可以省略，省略即为默认

处理盒子阴影在盒子的里面还是外面

默认省略，就是外面

`inset` 就是里面

第二个值：

具体的数值 可以是正数，也可以是负数 为盒子阴影X轴的大小

第三个值：

具体的数值 可以是正数，也可以是负数 为盒子阴影Y轴的大小

第四个值：

具体的数值 可以是正数，不可以是负数 可以省略 省略之后，没有模糊面积为盒子阴影模糊面积

第五个值：具体的颜色值，英文单词 十六进制 `rgb rgba`

盒阴影值可以不断的添加，中间用逗号隔开

文字阴影

第一个值：

具体的数值 可以是正数，也可以是负数 为文字阴影X轴的大小

第二个值：

具体的数值 可以是正数，也可以是负数 为文字阴影Y轴的大小

第三个值：

具体的数值 可以是正数，不可以是负数 可以省略 省略之后，没有模糊面积 为文字阴影模糊面积

第四个值：具体的颜色值，英文单词 十六进制 `rgb rgba`

文字阴影值可以不断的添加，中间用逗号隔开

• 课时44 蒙版 倒影 19:48

蒙版

目前只有`-webkit-`内核的浏览器支持，其他不支持
需要PNG的透明度遮罩
和背景设置一样，可以设置多个png为蒙版

倒影只有`-webkit-`内核浏览器支持

`-webkit-box-reflect: below;`

第一个值

控制倒影出现的位置

上 `above`

下 `below`

左 `left`

右 `right`

第二个值

倒影和本体之间间隙

为具体的数值： 比如 `10px`

`-webkit-box-reflect: below 10px;`

第三个值

可以透明度的渐变，也可以一个png透明的图片链接
透明度的渐变

```
-webkit-box-reflect:below 10px linear-
gradient(to top,
                                rgba(255,255,255,1) 10px,
                                rgba(255,255,255,0.3) 80px,
                                rgba(255,255,255,0) 200px,
                                rgba(255,255,255,0),
                                rgba(255,255,255,0)
                                );
png透明的图片链接
-webkit-box-reflect:left 10px
url("img/7.png");
```

章节12 媒体查询

- 课时45 媒体查询 13:35

例子:

```
@media screen and (max-width: 460px){
    div {
        width:300px;
        height:600px;
        border:30px solid yellow;
        background: #000;
    }
}
```

当屏幕发生变化，根据设置的最大值宽度或者最小值宽度时，读取大括号中的代码 比如：以下

```
@media screen and (max-width: 960px){
    会执行以下代码
    div {
        width:300px;
        height:600px;
        border:30px solid yellow;
        background: #000;
    }
}
```

- 课时46 viewport 布局百分比 18:47

```
<meta name="viewport" content="width=device-width,initial-scale=1.0,minimum-scale=1.0,maximum-scale=1.0,user-scalable=no" />
```

width :控制viewport的大小，可以制定一个数值或一个特殊的值
如: **device-width**就是设备的宽度（单位为缩放100%的CSS像素）

initial-scale:

初始缩放，就是页面开始缩放的程度，是一个浮点数

例如: **initial-scale=1.0**，页面开始的时候，就是1: 1的比例

maximum-scale=1.0，**minimum-scale=1.0**

最大缩放和最小缩放，允许放大或缩小多少倍

user-scalable=no

是否允许用户调整页面

如果为**YES** 用户可以改变

如果为**NO** 用户不可以改变

如果为**NO**，用户不能缩放，**maximum-scale=1.0**，**minimum-scale=1.0** 会失效

响应式布局

百分比布局

布局百分比

width
padding-left
padding-right
margin-left
margin-right
left
right

需要牢记的公式

目标元素的宽度 / 父级元素的宽度 = 百分比宽度

温馨提示:

高度的增加或者减少需要配合媒体查询

高度:

height
padding-top
padding-bottom
margin-top
margin-bottom
top
bottom

@media screen and () {}

图片百分比

图片的宽度和高度比较弹性化

图片如果不设置宽度和高度，会原始显示，当设置了图片的宽度百分比时，宽度发生变化，高度也会等比缩放

温馨提示，图片有时候太大，会超出，最好设置一个上限值，max-width 或者 max-height

文字百分比

em 和 rem

em父级的文字大小的倍数 父级大小是多少，决定子级文字em的大小

rem 就是针对HTML文字字体大小

一般响应式布局都用rem 通过媒体查询改变html的文字大小，就可以整体改变所有的页面中的文字大小

- 课时47 图片文字百分比 设备宽度自定义 23:19

图片的宽度和高度比较弹性化

图片如果不设置宽度和高度，会原始显示，当设置了图片的宽度百分比时，宽度发生变化，高度也会等比缩放

温馨提示，图片有时候太大，会超出，最好设置一个上限值，`max-width` 或者 `max-height`

`em` 和 `rem`

`em`父级的文字大小的倍数 父级大小是多少，决定子级文字`em`的大小

`rem` 就是针对HTML文字字体大小

章节13 HTML新标签使用及CSS选择器

- 课时48 H5新标签 语义 22:35

`<header></header>`

语义：文档的头部或者页面的头部

`<nav></nav>`

语义：导航

`<footer></footer>`

语义：文档的脚部或者页面的脚部，通常用来包含文档的版权信息 或者授权 或者友情链接

`<aside></aside>`

语义：侧边栏 或者附属信息

`<hgroup></hgroup>`

语义：标题的集合，可以用来包含多个`h1~h6`的标签

`<figure></figure>`

语义：图片区域，可以包含多个图片或者一个`figcaption`，`figcaption`用来表示图片标题

`<figcaption></figcaption>`

语义：图片区域的标题，一般写在`figure`

`<section></section>`

语义：分区，页面的或者文档的一部分区域，有独立的内容，但结构相近，就可以用`section`

可以包含 `header h1 ~ h6`等等

`<article></article>`

语义：独立的内容，可以是文章 `blog` 帖子 短文或者回复 评论

- 课时49 css3子代 兄弟 同级通用 属性选择器 15:04

子选择器：>

作用范围：`div`下的`span`会有享受样式，`p`里的不享受，原因，`p`里的`span`是孙子


```
div>span {
    border:1px solid red;
    background: yellow;
}
```

相邻兄弟选择器: +

作用范围 : 可以选择相邻的两个元素, 元素一定要有同一个父级, +后面的元素享受样式

```
p+span {
    border:1px solid red;
    background: yellow;
}
```

同级元素通用选择器: ~

作用范围: 所有相同父级中, 位于 P 标签之后的同级的 span 标签 享受样式

```
p~span {
    border:1px solid red;
    background: yellow;
}
```

属性选择器

第一种

选择带有某种属性的所有元素

```
a[属性名]{.....}
```

第二种

选择带有某种属性, 并且规定属性值的元素

```
input[属性名="属性值"]{
    .....
}
```

- 课时50 A标签的伪类 伪类选择器 18:50

伪类选择器

选择属于其父级下的第n个元素, 并且必须是p标签, 直到匹配位置, 否则不生效

```
p:nth-child(n){
    background: red;
}
```

温馨提示: n可以用来计算

选择属于其父级下的第1个元素, 并且必须是p标签, 否则不生效

```
p:first-child{
    background: red;
}
```

选择属于其父级下的最后1个元素, 并且必须是p标签, 否则不生效

```
p:last-child{
    background: red;
}
```

```
}
```

选择属于其父级下的第n个元素，并且一定会匹配p标签

```
p:nth-of-type(n){
    background: red;
}
```

温馨提示：n可以用来计算

选择属于其父级下的第1个元素，并且一定会匹配p标签

```
p:first-of-type{
    background: red;
}
```

选择属于其父级下的最后1个元素，并且一定会匹配p标签

```
p:last-of-type{
    background: red;
}
```

伪类：是一种特殊性的属性， 可以理解为表示一种状态 love hate 爱恨记忆法
after
before

章节14 服务器字体及弹性盒模型

- 课时51 服务器字体 18:22

```
@font-face {
    font-family:"瞎编的 自定义的字体名字";
    src:
        url("字体图片的路径地址"),
        url("字体图片的路径地址"),
        url("字体图片的路径地址"),
        url("字体图片的路径地址");
    font-weight;;
    font-style;;
}
@font-face {
    font-family: 'b_surfersplane';
    src: url('bsurp___-webfont.eot');
    src: url('bsurp___-webfont.eot?#iefix') format('embedded-opentype'),
        url('bsurp___-webfont.woff2') format('woff2'),
        url('bsurp___-webfont.woff') format('woff'),
        url('bsurp___-webfont.ttf') format('truetype'),
        url('bsurp___-webfont.svg#b_surfersplane') format('svg');
    font-weight: normal;
    font-style: normal;
}
.ttf
```

```
.otf
.woff
.eot
.svg
http://www.dafont.com/ 找字体的网站
https://www.fontsquirrel.com/tools/webfont-generator 修改字体图片格式的线上工具
```

- 课时52 字体图标 怪异盒模型 17:57

```
www.iconfont.cn
@font-face {
  font-family: "aaaa";
  src: url('fontImg/1/iconfont.eot?t=1478759787428'); /* IE9*/
  src: url('fontImg/1/iconfont.eot?t=1478759787428#iefix')
  format('embedded-opentype'), /* IE6-IE8 */
  url('fontImg/1/iconfont.woff?t=1478759787428') format('woff'), /*
chrome, firefox */
  url('fontImg/1/iconfont.ttf?t=1478759787428') format('truetype'), /*
chrome, firefox, opera, Safari, Android, iOS 4.2+*/
  url('fontImg/1/iconfont.svg?t=1478759787428#iconfont') format('svg'); /*
iOS 4.1- */
}

.iconfont {
  font-family:"aaaa" ;
  font-size:40px;
  font-style:normal;
  width:300px;
  height:300px;
  border:1px solid red;
  color:#2EEF64;
  text-indent: 30px;
  line-height: 300px;
}

.iconfont:before { content: "\3432"; }
```

box-sizing: 盒模型的设置

标准盒模型 content-box

盒子的宽: border + padding + width

盒子和高: border + padding + height

怪异盒模型 border-box

盒子和宽: 就是设置的width 会包含 border + padding

盒子的高: 就是设置的height 会包含 border + padding

- 课时53 弹性盒模型 分栏 31:28

弹性盒模型，不兼容，w3c目前还没有标准化，需要加浏览器内核前缀

如果想要搭建弹性盒模型，首先要在父级建立弹性盒模型的空间
方法：

```
display:-webkit-box;
display:-moz-box;
display:-ms-box;
display:-o-box;
display:box;
```

排列方式 有两种 水平 和 垂直

`-webkit-box-orient:vertical;` 垂直

`-webkit-box-orient:horizontal;` 水平 默认值

子级的排列位置，可以给父级加`-webkit-box-pack` 来实现
方法：

```
-webkit-box-pack:
-moz-box-pack:
-ms-box-pack:
-o-box-pack:
box-pack:
```

相应的值有：

`start` 左边 默认值

`center` 中间

`end` 右边

`justify` 均分

子级自己本身可以均分父级的宽度，根据自身的分数进行均分

设置分数的方法

给子级加

`-webkit-box-flex:几份;`

比如：

`-webkit-box-flex:1;` 根据一份的宽度来计算 分一份

`-webkit-box-flex:5;` 根据一份的宽度来计算 分五份

`column-count:3;` 分几栏

`column-gap:100px;` 分栏之间的间隙 数值越大 间隙越大

`column-rule: 5px solid #25AB12;` 类似border的设置，可以设置分栏之间的分界线

ie9以下不支持 火狐需要加前缀

`!important` 不推荐使用

一般写在想要的样式后面，把相应的样式的优先级提升至最高

章节15 CSS3动画过渡

- 课时54 过渡 18:10

过渡，就是从一個值到另一个值之间的过程

transition

比如：

transition:1s 1s all ease; 后三个值可以省略

第一个值：过渡的总时间 单位是s或ms

第二个值：延迟过渡的时间 单位是s或ms

第三个值：需要过渡的样式

第四个值：过渡的方式

linear 匀速

ease 慢速开始，然后慢慢变快

ease-in 慢速开始

ease-out 慢速结束

ease-in-out 开始和结束都慢

cubic-bezier(1,0,1,0) 贝瑟尔曲线

贝瑟尔曲线线上工具

<http://xuanfengge.com/easeing/ceaser/>

- 课时55 2D变换 20:42

变换 **transform**

位移

translate 当中接受的值就是正常的px值 ru: 200px

translateX() 正值往右走 负值往左走

translateY() 正值往下走 负值往上走

translate(X轴的位移量,Y轴的位移量)

第一个值X轴位移量

第二个值Y轴的位移量

中间用逗号隔开

旋转

rotate 当中接受的是一个旋转角度 如: 45deg

rotate() 正值是顺时针 负值逆时针

缩放

scale 当中接受的是一个数值，可以使浮点数 如: 1.2 或者 0.3 或者 2

scale() 大于1 是放大效果 小于1 是缩小效果

斜切

skew 当中接受的也是一个旋转角度 如: 40deg

skewX()

skewY()

skew(30deg,30deg)

第一个值X轴斜切角度

第二个值Y轴斜切角度
中间用逗号隔开

变换原点

`transform-origin`

`left`

`top`

`right`

`bottom`

`center`

默认是`center` 当设置相应的值后，会按照相应的设置的值发生变换

- 课时56 3D变换 27:12

3d效果，首先需要给需要进行3d变换的元素的父级，添加3d环境 和 景深
方法：

`transform-style:preserve-3d;` 添加3d环境
`perspective:数值;` 如: `300px;`

`transform`

`rotateX` 跳水 或者 后手翻 前空翻

`rotateY` 扫堂腿 或者 红人 芭蕾舞

`rotateZ` 大风车 或者 风火轮

`translateX` 向左右 向右走

`translateY` 上天 入地

`translateZ` 冲着你来 或者 冲着你倒退

兼容 ie9以下不支持

- 课时57 animation 关键帧 17:14

动画

`animation`

首先要先定义一套关键帧

方法：

```
@keyframes 动画名称{
    0%{
        相应的样式
    }
    10%{
        相应的样式
    }
    20%{
        相应的样式
    }
    100%{
```

相应的样式

```

    }
}

```

(温馨提示, 可以从0%一直设置到100%)

然后调用 动画名称
方法:

数 `animation:1s 2s 动画名称 运动方式 动画执行的次数`

第一个值: 运动的总时间
第二个值: 延迟的运动时间
第三个值: 动画名称
第四个至: 运动方式
`linear` 匀速
`ease` 慢速开始, 然后慢慢变快
`ease-in` 慢速开始
`ease-out` 慢速结束
`ease-in-out` 开始和结束都慢
`cubic-bezier(1,0,1,0)` 贝瑟尔曲线
贝瑟尔曲线线上工具
<http://xuanfengge.com/easeing/ceaser/>

第五个值: 动画执行的次数
可以具体的数值 也可以是无限次
`infinite`

章节16 JS认知

- 课时58 JS认知 18:06
- 课时59 显示隐藏 29:45

下拉列表的显示隐藏
变量声明
方法:

`var` 变量名
`var`变量声明的关键字
变量名就是你自己瞎编的 但是不能是关键保留字 比如: `var`

`function`

关键字 保留字的网址 <http://www.runoob.com/js/js-reserved.html>

章节17 JS属性操作

- 课时60 JS属性操作读写 13:59

属性操作

属性 分为属性名 和 属性值

在标签中设置属性 方法: 属性名="属性值"

在CSS中设置属性 方法: 属性名:属性值;

操作

读操作 获取 查找

方法: 元素.属性名 比如: `oDiv.style.width`

写操作 添加 替换 修改

方法: 元素.属性名 = 新要设置的值 比如:

```
oDiv.style.width = '400px'
```

= 一个等号就是赋值

会把等号右边的给左边

- 课时61 字符串拼接 15:43

字符串的拼接

字符串就是JS数据类型的一种

数字 可以进行运算

字符串 进行拼接

布尔

+

数字类型相加可以运算 就是相加

字符串类型和任何类型相加都是字符串, 都会进行拼接

- 课时62 if 16:13

- 课时63 innerHTML className 22:11

innerHTML就是元素里面所有的东西 文字 换行 标签

innerHTML可以进行读操作 也可以进行写操作

`oDiv.innerHTML`

读取元素里面的内容

`oDiv.innerHTML = 新的值;`

替换

用新的值 代替 旧的值

+=

在自身的基础上 累加一个新的值 可以用加等
比如

```
a = a + 2;
```

```
a += 2;
```


在比如：

```
oDiv.innerHTML = oDiv.innerHTML + '诺手';  
oDiv.innerHTML += '诺手';
```

- 课时64 “-” 33:15

在JS中的属性操作中，不允许出现“-”，用驼峰式命名代替

数组就是用来存数据的大仓库

1. 每一个数据中间用逗号隔开
2. 通过数组下标读取 下标是从0开始

比如

```
var arr = [1,2,'img/1.jpg'];  
alert(arr[2]);
```

3. 数组的长度 length

章节18 For循环

- 课时65 for 20:25
- 课时66 for添加事件处理函数 18:57
- 课时67 this 28:25
- 课时68 自定义属性 20:58
- 课时69 开关 22:55

章节19 数据

- 课时70 数据类型 16:44
- 课时71 数据类型转化 19:51
- 课时72 隐式类型转化 17:11
- 课时73 NaN 20:11

章节20 传参

- 课时74 函数传参 12:05
- 课时75 传参例子布局 05:52
- 课时76 传参例子讲解上 12:25
- 课时77 传参例子讲解下 14:50

章节21 变量作用域

- 课时78 变量作用域（上） 18:22
- 课时79 变量作用域（中） 12:31
- 课时80 变量作用域（下） 06:05

章节22 函数

- 课时81 字符串函数（上） 12:21
- 课时82 字符串函数（下） 18:54
- 课时83 数组函数（上） 16:19
- 课时84 数组函数（下） 19:43
- 课时85 日期函数（上） 15:30
- 课时86 日期函数（下） 05:11
- 课时87 计时器 19:26

章节 23JS动画

课时88获得计算后样式上12:48 课时89获得计算后样式下11:02 课时90大图滚动117:13 课时91大图滚动210:36 课时92大图滚动313:09 课时93样式获得10:46 课时94样式获得练习及随机数17:23 课时95Tweenjs12:08

章节24JS之DOM

课时96节点操作112:28 课时97节点操作214:58 课时98DOM添加标签113:10 课时99DOM添加标签209:39 课时100DOM常用属性115:45 课时101DOM常用属性213:48 课时102DOM常用属性练习19:37

章节25JS 事件

课时103事件对象上14:30 课时104事件对象下14:53 课时105事件对象练习及鼠标事件上19:03 课时106事件对象练习及鼠标事件下13:45 课时107鼠标事件练习122:01 课时108事件冒泡17:41 课时109事件监听20:01 课时110移入移出效果练习18:17 课时111阻止默认事件及键盘事件上13:18 课时112阻止默认时间及键盘事件下13:56 课时113滚轮事件练习24:42 课时114自定义滚动条115:42 课时115自定义滚动条212:27 课时116碰撞检测118:10 课时117碰撞检测208:44 课时118FlappyBird讲解117:17 课时119FlappyBird讲解210:18 课时120FlappyBird讲解320:34 课时121FlappyBird讲解415:13 课时122FlappyBird讲解509:24 课时123表单事件14:48 课时124移动端事件一20:46 课时125移动端事件二14:20 课时126移动端事件三10:01 课时127移动端事件四08:02 课时128移动端事件五-摇一摇12:46 课时129TouchJS安装及使用10:44 课时130TouchJS实现拖拽13:26 课时131TouchJS实现旋转11:16 课时132TouchJS缩放08:07 课时133Swiper介绍16:57 课时134Swiper使用120:42 课时135Swiper使用219:26 课时136Swiper使用318:08 课时137Swiper使用413:59 课时138Swiper使用517:09

章节26JS正则表达式

课时139正则表达式的创建13:57 课时140正则及字符串常用函数17:56 课时141正则量词与元字符117:12 课时142正则量词与元字符210:03 课时143正则量词与元字符315:15 课时144正则量词与元字符413:13 课时145正则量词与元字符509:20 课时146正则量词与元字符611:20 课时147正则转译字符17:41 课时148正则QQ号验证14:45 课时149正则电话号验证11:07 课时150正则HTML标签验证17:08

章节27JS面向对象

课时151面向对象概述及创建上17:39 课时152面相对象概述及创建下15:14 课时153面向对象创建上11:57 课时154面相对象创建中13:44 课时155面向对象创建下23:58 课时156面向对象案例讲解上17:06 课时157面向对象案例讲解下11:53 课时158面向对象原型17:14 课时159面向对象封装案例18:20

章节28jQuery

课时160jQuery介绍14:20 课时161jQuery介绍212:43 课时162jQuery选择器115:19 课时163jQuery选择器217:45 课时164jQuery选择器316:24 课时165jQuery选择器413:11 课时166jQueryDOM操作117:06 课时167jQueryDOM操作215:38 课时168jQuery事件121:11 课时169jQuery事件228:00 课时170jQuery样式设置1 课时171jQuery样式设置217:37 课时172jQuery动画及插件32:08 课时173jQuery盖房子练习121:07 课时174jQuery盖房子练习217:20 课时175jQuery盖房子练习319:27 课时176jQuery盖房子练习416:05

章节29WEB前端框架—bootstrap

课时177HBuilder和Less的变量使用19:54 课时178Less使用10:44 课时179BootStrap全局样式20:36 课时180BootStrap表格和按钮15:54 课时181BootStrap表单和图片20:15