

HTML

基础标签

- h1 - h6 标题标签 `<h1></h1>`
- p 段落标签
- `
` 换行 (单标签)
- div 块级标签
- span 行内标签
- img 图片 (单标签) 属性：`src` --图片路径 `alt` --图片无法显示时的替换标签 `title` --鼠标放上去的图片名
- 路径 相对路径 绝对路径 同一级直接写，下一级用 `/`，上一级用 `../`
- a 超链接 属性：`href` --必须属性，用于跳转 `target` --链接打开方式 `_self`表示再当前页面打开，是默认值 `_blank`表示在新标签页内打开
- 空链接 `href="#"` 如果链接地址是文件或者压缩包，会直接下载 锚链接：`id="id" href="#id"`

表格

`<table>`表示表格，`<tr>`表示行 `<td>`表示列

`<th>`表示表头标签，写在`<tr>`里面

- `colspan="合并的单元格数"` --在一行内合并单元格
- `rowspan="合并的单元格数"` --在一列内合并单元格
- 单元格合并代码写在`<td>`里面
- 表格一般在CSS里面定义样式

列表

- `ul>li` 无序列表
- `ol>li` 有序列表
- `dl>dt dd` 自定义列表
- 列表里面直可以写 `li` 不可以写其他代码

表单

表单内容写在`<form>` 里面 属性包括：`action` --表单接收的地址 `method` --提交方式，值可以是`get/post` `name` --表单的名称，可能存在很多表单

input 属性type

type属性包括

- `button` 按钮
- `text` 文本
- `password` 密码
- `radio` 单选按钮 在input里面加id，用`<label for = "id">` 实现点击内容选中该按钮
- `checkbox` 复选按钮
- `submit` 提交按钮
- `reset` 重置按钮 点击清空表单内容

- file 上传文件

其他属性

- name input的名字
- value input元素的值
- checked 表单被加载的时候被选中
- placeholder 提示性的文本，点击之后会消失，不作为正式文本

select 下拉菜单 属性 option

select 中必须包含一个以上的option，如果想加载出来的默认某个，在option 里面加 selected = "selected"

textarea 文本域

属性rows 行数 cols 每行最多字符

CSS

基础选择器

标签选择器 类选择器 id选择器 通配符选择器

基础属性

- font : font-size font-family font-weight font-style
- text
 - color
 - line-height
 - text-decoration 一般清除链接的下划线
 - text-align 文本居中对齐
 - text-indent 首行缩进
- background
 - background-color
 - background-image:url(url)
 - background-repeat:repeat/no-repeat/repeat-x/repeat-y 背景平铺
 - background-position: x y
 - background-attachment:fixed(固定)/scroll(滚动)
 - background : rgba(0,0,0, .3) 透明写法

复合选择器

- 后代选择器 ul li
- 子代选择器 选出第一层子代的所有元素
- 并集选择器
- 链接伪类 a:hover

显示模式

块级元素 h1 p div ul li ol

- 独占一行
- 宽高，内外边距可设定
- 宽度默认是父级宽度
- 里面可以放块级或行内元素，文本元素只放文本 h p

行内元素 a span strong b em i del 等

- 一行显示
- 宽高设置无效
- 宽度默认内容宽度，没有内容就是0
- 行内元素只能放文本或其他行内元素
- 链接里面不能放链接，链接里面可以放块级元素，但转换一下块级元素是安全的

行内块元素 img input td

- 行内块和行内元素，可以一行显示，中间会有空白间隙
- 宽度默认自身内容宽度
- 高度，行高，内外边距可控

转换

- display:block
- display:inline
- display:inline-block

盒子模型

border

- border:1px solid rgba() (border-size border-style border-color)
- border-collapse:collapse 边框合并，常用在表格当中
- 边框会影响盒子的大小

padding 内边距

- 5px 上下左右
- 5px 10px 上下 左右
- 5px 10px 15px 上 左右 下
- 5px 10px 15px 5px 上 右 下 左

margin

- 用法同上
- 清除内外边距
- 上下两个盒子都有margin 会向大的塌陷 只给一个设边距
- 父子都有上外边距，会向大的塌陷 给父级加上内边距或上边框，或父级加overflow:hidden

其他

- border-radius 圆角边框
- box-shadow:h-shadow(水平位置) v-shadow(垂直位置) blur(模糊距离) spread(阴影尺寸) color(颜色) inset --盒子阴影
- text-shadow

浮动 float

浮动可以让多个块级盒子一行显示

- float : left
- float: right
- 任何元素都可以添加浮动，添加之后有行内块元素的特性，中间不会有空白间隙
- 块级盒子添加浮动之后，宽度不随父级盒子，而是根据内容定
- 行内元素同上
- 浮动元素脱离标准流，不会占有原来的位置
- 浮动元素一行显示并且顶端对齐
- 浮动元素一般有一个父级标准流的盒子
- 一浮全浮，一个元素浮动，理论上他的兄弟盒子也要全部浮动
- 浮动盒子只影响后面的标准流，不会影响后面的标准流

清除浮动

浮动的父级盒子给了指定的高度后，下面的内容不会占用原来的位置。如果没有指定高度，由于浮动盒子不占有原来位置，所以下面的元素会占用原来位置，下面的元素就会顶上来。此时需要清除浮动。

- **额外标签法** ---在浮动元素的末尾添加一个空的标签，例如<div style = "clear:both"></div> 或者</br>
- 父级添加 overflow:hidden/auto/scroll 缺点是溢出的部分会被隐藏
- 父级添加**after**伪元素

```
.clearfix:after {  
    content:"";  
    display:block;  
    height:0;  
    clear:both;  
    visibility:hiddden;  
}
```

- 父级添加双伪元素

```
.clearfix:before,.clearfix:after {  
    content:"";  
    display:table;  
}  
  
.clearfix:after {
```

```
clear:both;
}
```

定位

定位可以让盒子自由的在盒子内移动或者固定屏幕中的某个位置，并且可以压住其他盒子

定位 = 定位模式 + 偏移量（记住一定要加偏移量，否则定位不起作用）

- 偏移量：定义元素相对于父级元素的边线距离 top left right bottom

relative 相对定位

相对定位是元素移动的时候，相对它原来的位置来说的

- 相对原来的位置
- 原来位置的标准流继续占有
- 相对定位没有脱标

absolute 绝对定位

绝对定位移动的时候是相对与其祖先元素而言

- 没有祖先元素或者祖先元素没有定位，则相对于浏览器准定位
- 如果祖先元素有定位（包括相对定位，绝对定位，固定定位），以最近一级的父级元素作为参考点
- 绝对定位不占有原来的位置
- 脱标

fixed固定定位

固定定位是相对于浏览器的可视区域

- 以浏览器的可视区域伪参考点移动元素
- 与父元素没有关系
- 不随滚动条滚动
- 不占原来的位置（脱标）
- 固定定位的盒子 left:50% 走到浏览器的一半位置 然后margin-left：版心的一般大小 便可在侧边固定住盒子

sticky 粘性定位

粘性定位是相对定位和固定定位的混合

- 以浏览器为可视窗口来移动元素
- 粘性定位占有原来的位置
- 必须加一个偏移量
- 跟页面滚动配合使用

拓展

浮动元素只会压住下面标准流的盒子，不会压住盒子里面的文字

绝对定位和固定定位会压住下面的所有内容

- 绝对定位的盒子不可以用`margin: 0 auto`的方式实现居中对齐，`left: 50% margin-left: -自身盒子的一半`内容
- 脱标的盒子不会引发外边距塌陷 浮动 绝对定位 固定定位
- 行内元素添加绝对定位 固定定位，可直接设置宽高
- 块级元素添加绝对定位 固定定位，没有指定宽高的话，宽高默认等于他的内容的大小

z-index 定位的叠放顺序

- 数值可以是正数，负数或0，默认是`auto`，数值越大盒子越靠上
- 属性值相同，后来者居上
- 数字后面不加单位
- 定位的盒子才有该属性

子绝父相

由于绝对定位的偏移量是相对父级定位盒子，所以一般给加绝对定位的盒子指定一个相对定位的父级盒子

元素的显示和隐藏

- `display` 显示和隐藏元素 不保留原来的位置
- `visibility` 显示和隐藏元素 保留原来的位置
- `overflow` 溢出显示隐藏 只对溢出来的部分做处理

display

- `display:none` 隐藏对象
- `display:block` 除了转换成块级元素，还有显示元素的意思
- 不占有原来的位置（用处更多）

visibility

- `visibility:visible` 元素可视
- `visibility:hidden` 元素隐藏
- 隐藏后继续占有原来的位置

overflow

该属性规定如果内容超出了一个元素的框，会发生什么

- `visible` 不剪切内容 不加滚动条
- `hidden` 不显示超出对象的内容 超出的部分隐藏
- `scroll` 不管是否超出内容，总显示滚动条
- `auto` 超出自动显示滚动条 不超出不显示
- 定位盒子慎用，会不显示超出部分

H5

新增语义化标签

- <header> 头部标签
- <nav> 导航标签
- <article> 内容标签
- <section> 定义文档某个区域
- <aside> 侧边栏标签
- <footer> 尾部标签
- 标签可以使用多次

多媒体标签

音频 audio 视频 video

可以方便的在页面中插入音频和视频，不再去使用flash和其他浏览器插件

视频 video

视频的格式有 MP4 WebM Ogg 不同格式浏览器支持情况不一样

```
<video src="文件地址" controls = "controls"></video>
```

```
<video controls="controls" width = "300">  
  <source src = "move.ogg" type="video/ogg">  
  <source src = "move.mp4" type="video/mp4">  
  您的浏览器暂不支持<video>标签播放视频  
</video>
```

属性

属性	值	描述
autoplay	"autoplay"	视频就绪自动播放（谷歌浏览器加muted来解决自动播放问题）
controls	"controls"	向用户显示播放控件
width	pixels	播放器宽度
height	pixels	播放器高度
loop	"loop"	播放完之后是否循环播放
preload	"auto"(预先加载视频)/"none"(不应加载视频)	规定是否预加载视频 如果有了 autoplay 忽略该属性
src	"url"	视频url地址
poster	"imgurl"	加载视频时候的画面图片

属性	值	描述
muted	"muted"	静音播放

音频 audio

音频格式 Mp3 Wav Ogg H5 在不使用插件的情况下可以插入原生音频

```
<audio src="文件地址" controls = "controls"></audio>
```

```
<audio controls="controls">
  <source src = "happy.mp3" type="audio/mpeg">
  <source src = "happy.ogg" type="audio/ogg">
  您的浏览器暂不支持<audio>标签播放视频
</audio>
```

属性

属性	值	描述
autoplay	autoplay	出现该属性，音频自动播放
controls	controls	出现该属性，向用户显示控件
loop	loop	出现该属性之后，结束之后自动播放
src	src	播放音频的url

谷歌浏览器把音视频的自动播放给禁止了

多媒体标签总结

- 音视频使用方式基本一致
- 浏览器支持情况不同
- 谷歌浏览器把音视频的自动播放给禁止了
- 视频加muted静音播放，音频不可以（可以通过javascript解决）
- 视频标签是重点，经常设置自动播放，不使用controls控件，循环和设置大小

新增 input 属性

新增 input 类型

属性值	说明
type="email"	限制输入类型为Email
type="url"	限制输入类型为URL
type="date"	限制输入类型为日期

属性值	说明
type="time"	限制输入类型为时间
type="month"	限制输入类型为月
type="week"	限制输入类型为周
type="number"	限制输入类型为数字
type="tel"	手机号码
type="search"	搜索框
type="color"	生成一个颜色选择表单

常用 number tel search

属性值	说明
required="required"	拥有该属性表示内容不能为空
placeholder="提示文本"	提示信息，表单有默认值则不显示
multiple="multiple"	可以多选文本提交

```
input::placeholder {
  color: pink;
}
```

CSS

新增选择器

属性选择器

选择器	简介
E[att]	选择具有att属性的E元素
E[att="val"]	选择att属性等于val的E元素
E[att^="val"]	选择att属性以val开头的E元素
E[att\$="val"]	选择att属性以val结尾的E元素
E[att*="val"]	选择att属性是以val为中间值的E元素

结构伪类选择器

选择符	简介
-----	----

选择符	简介
E:first-child	匹配父元素中的第一个子元素 E
E:last-child	匹配父元素中的最后一个E元素
E:nth-child(n)	匹配父元素中的第n个子元素
E:first-of-type	指定类型 E 的第一个
E:last-of-type	指定类型 E 的最后一个
E:nth-of-type(n)	指定类型 E 的第 n 个

区别 nth-child : 先将里面所有子元素排序, 再找第 n 个看能不能匹配的上 **nth-of-type** : 对里面的指定的子元素E 排序, 然后根据E找到第n个子元素

- n 可以是数字, 关键字和公式
- n 是数字, 选择第 n 个, 里面数字从 1 开始
- n 是关键字的话 even 偶数 odd 基数
- n 是公式的话
 - 2n 偶数
 - 2n+1 奇数
 - 5n 5 10 15...
 - n+5 从第五个开始
 - -n+5 前五个 (包含第五个)

伪元素选择器

伪元素选择器可以帮我们利用CSS创建新标签, 而不需要HTML标签, 从而简化HTML结构

- :: before 在元素内部前面插入内容
- :: after 在元素内部的后面插入内容
- 插入的元素属于行内元素
- before 和 after 里面必须要有 content 属性
- 伪元素选择器和标签选择器一样, 权重为1

CSS3

盒子模型

- box-sizing : content-box/border-box
- content-box: 内边距和边框会撑开盒子大小, 盒子大小等于width+padding+border
- border-box: 内边距和边框不会撑开盒子大小, 盒子大小等于width
- calc : 可以执行一些宽度的计算 包括 + - * /

- width:calc(100% - 80px);

过渡

从一种样式到另一种样式时为元素添加效果

目前和：hover 搭配使用

谁做过渡给谁加这个属性

- transition: 要过渡的属性 花费时间 运动曲线 何时开始；
- 属性：想要变化的css属性，宽高度 背景颜色 内外边距都可，如果想要所有元素都变化过渡，加一个all
- 花费时间：0.5s
- 运动曲线：默认ease
- 何时开始：可以设置延时触发时间 默认0s

2D

2D标签是改变二维平面上的位置，类似定位

移动 translate

```
transform: translate(x,y);
transform: translateX(n);
transform: translateY(n);
```

- 不会影响其他元素的位置
- x,y是百分比的话，是相对于自身元素而言的
- 对行内标签没有效果

旋转 rotate

- transform: rotate(度数)；
- 度数单位是deg，正数是顺时针，负数是逆时针
- 默认旋转的中心是元素的中心点

转换中心点

- transform-origin: x y;
- x y中间用空格隔开
- 默认元素的中心点 (50% 50%)
- 还可以设置元素或者是 方位名词

缩放 scale

- transform:scale(x,y);参数中间有逗号隔开
- transform:scale(2,2)，宽高都放大2 两倍

- 默认中心点缩放，不影响其他盒子

综合写法的时候，先写位移，避免二维坐标被改变

动画

- 先定义动画
- 再调用动画

```
定义动画
@keyframes 动画名称 {
  0% {
    width:100px;
  }
  100% {
    width:200px;
  }
}
```

```
调用动画
div {
  /*动画名称*/
  animation-name: 动画名称;
  /* 持续时间 */
  animation-duration: 持续时间;
}
```

动画常见属性

属性	描述
@keyframes	规定动画
animation	所有动画属性的简写属性，除了animation-play-state属性
animation-name	规定@keyframes动画的名称。（必须的）
animation-duration	规定动画完成一个周期所花费的秒或毫秒，默认是0。（必须的）
animation-timing-function	规定动画的速度曲线，默认是“ease”
animation-delay	规定动画何时开始，默认是0
animation-iteration-count	规定动画被播放的次数，默认是1，还有infinite
animation-direction	规定动画是否在下一周期逆向播放，默认是“normal”,alternate逆播放
animation-play-state	规定动画是否正在运行或暂停。默认是“running”,还有“paused”
animation-fill-mode	规定动画结束后状态，保持forwards回到起始backwards

简写属性

- `animation` : 动画名称 持续时间 运动曲线 何时开始 播放次数 是否反方向 动画起始或者结束的状态;
- `animation: myfirst 5s linear 2s infinite alternate;`
- 简写属性里面不包括`animation-play-state` 暂停动画
- `animation-play-state: paused;`
- 经常和鼠标经过等其他配合使用
- 想要动画走回来 , 而不是直接跳回来 : `animation-direction : alternate`
- 盒子动画结束后 , 停在结束位置 : `animation-fill-mode : forwards`

速度曲线调节

`animation-timing-function` : 规定动画的速度曲线 , 默认是“ease”

值	描述
linear	动画从头到尾的速度是相同的。匀速
ease	默认。动画以低速开始 , 然后加快 , 在结束前变慢。
ease-in	动画以低速开始。
ease-out	动画以低速结束
ease-in-out	动画以低速开始和结束。
steps()	指定了时间函数中的间隔数量 (步长)

3D 转换

三维坐标系

- x轴 : 水平向右 注意 : x 右边是正值 , 左边是负值
- y轴 : 垂直向下 注意 : y 下面是正值 , 上面是负值
- z轴 : 垂直屏幕 注意 : 往外面是正值 , 往里面是负值

主要知识点

- 3D位移: `translate3d(x,y,z)`
- 3D旋转: `rotate3d(x,y,z)`
- 透视: `perspective`
- 3D呈现 `transform-style`

translate3d

3D移动在2D移动的基础上多增加了一个可以移动的方向 , 就是z轴方向
因为z轴是垂直屏幕 , 由里指向外面 , 所以默认是看不到元素在z轴的方向上移动

- `transform:translateX(100px)`：仅仅是在x轴上移动
- `transform:translateY(100px)`：仅仅是在Y轴上移动
- `transform:translateZ(100px)`：仅仅是在Z轴上移动（注意：`translateZ`一般用px单位）
- `transform:translate3d(x,y,z)`：其中 x、y、z 分别指要移动的轴的方向

透视 perspective 单位是px

在2D平面产生近大远小视觉立体，但只是效果二维的

- `transform:translateZ(100px)`：仅仅是在Z轴上移动。有了透视，就能看到`translateZ`引起的变化了
- `translateZ`：近大远小
- `translateZ`：往外是正值
- `translateZ`：往里是负值

rotate3D 3D旋转

3D旋转指可以让元素在三维平面内沿着 x轴，y轴，z轴或者自定义轴进行旋转。语法

- `transform:rotateX(45deg)`：沿着x轴正方向旋转 45度
- `transform:rotateY(45deg)`：沿着y轴正方向旋转 45deg
- `transform:rotateZ(45deg)`：沿着Z轴正方向旋转 45deg
- `transform:rotate3d(x,y,z,deg)`：沿着自定义轴旋转 deg为角度（了解即可）

左手准则

- 左手的手拇指指向 x轴的正方向
- 其余手指的弯曲方向就是该元素沿着x轴旋转的方向

3D呈现 transform-style

- 控制子元素是否开启三维立体环境。。
- `transform-style: flat` 子元素不开启3d立体空间 默认的
- `transform-style: preserve-3d`; 子元素开启立体空间
- 代码写给父级，但是影响的是子盒子
- 这个属性很重要，后面必用

3D转换的中心在元素的中心点位置

浏览器私有前缀

浏览器私有前缀是为了兼容老版本的写法，比较新版本的浏览器无须添加。

私有前缀

- `-moz-`：代表 firefox 浏览器私有属性
- `-ms-`：代表 ie 浏览器私有属性
- `-webkit-`：代表 safari、chrome 私有属性
- `-o-`：代表 Opera 私有属性

提倡的写法

- -moz-border-radius: 10px;
- -webkit-border-radius: 10px;
- -o-border-radius: 10px;
- border-radius: 10px;