# 第一天

1、复杂选择器

1、兄弟选择器

兄弟：相同父元素并且是同级别的元素可称为兄弟元素

1、相邻兄弟选择器

作用：匹配指定元素的相邻兄弟元素(只能向后找，不能向前找)

语法：selctor1+selecotr2

2、通用兄弟选择器

作用：匹配指定元素后面所有的兄弟元素

语法：selector1~selector2

2、属性选择器

作用：能够将元素所附带的属性及其值用于选择器中。

<input type="" maxlength="" name="" id="">

语法：[]

1、[attr] : 元素中 附带 attr属性的元素

[id] : 匹配页面中所有附带id属性的元素

[type] : 匹配页面中所有 附带type属性的元素

2、element[attr]:匹配 附带 attr 属性的 element 元素

p[class] : 匹配 附带 class 属性的 p 元素

3、[attr=value]:匹配 附带的属性名称为 attr 并且值为 value的 所有元素

[type=text]:获取页面中所有的文本框

[type=checkbox] : 获取页面中所有的复选框

获取页面中 name 属性值为 txtUserName 的input元素

input[name=txtUserName]

[id=d1]

[id='d1']

[id="d1"]

4、[class~=value]

class属性值，是以空格 隔开的值列表，value是值列表中的一个单词

<div class="important danger normal redBack">XXXX</div>

div[class~=danger] : 匹配 class 值列表中 包含 danger 属性值 的 div元素

5、[attr^=value]

^ : 匹配开始

匹配 attr 属性值 是以 value 字符开始的元素

<div class="font\_24"></div>

<div class="font\_32"></div>

[class^="font\_"]{

font-weight:bold;

}

6、[attr$=value]

匹配 attr 属性值 以 value作为结尾的元素

7、[attr\*=value]

匹配 attr 属性值 中包含 value 字符的元素

3、伪类选择器

1、目标伪类

作用：用于匹配 活动的 html锚元素。

定义锚点：

1、<a name></a>

2、<ANY id=""></ANY>

语法： :target

a:target :

2、元素状态伪类

元素状态：

状态：启用，禁用，被选中

语法：

:enabled ，匹配每个已启用的元素(表单控件使用)

disabled ，匹配每个已禁用的元素(表单控件使用)

:checked ，匹配每个被选中的元素(只用于 radio 和 checkbox)

3、结构伪类(重点)

1、:first-child

作用：匹配属于其父元素中的首个子元素

td:first-child

2、:last-child

作用：匹配属于其父元素中的最后一个子元素

3、:empty

匹配没有子元素的指定元素

div:empty

<p>Hello</p> 非empty

<p> </p> 非empty

<p>

</p> 非empty

<p></p> : :empty

4、:only-child

作用：匹配属于其父元素中的唯一子元素

5、:nth-child(n)

作用：匹配属于其父元素中的第 n 个子元素

:nth-child(5)

4、否定伪类

作用：将指定选择器的元素排除在外

语法： :not(selector)

4、伪元素选择器

作用：用于匹配元素中的某块内容

1、:first-letter

::first-letter

作用：用于选取指定选择器的首字符。

2、:first-line

::first-line

作用：用于选取指定选择器的首行内容

3、::selection

作用：用于匹配用户选取的部分

2、内容生成

1、什么是内容生成

通过 css 向某元素的 内容区域之前 或 之后 ，增加一部分内容

2、伪元素选择器

1、:before 或 ::before

匹配 某元素 的内容区域之前

2、:after 或 ::after

匹配 某元素 的内容区域之后

注意：以上两个伪元素选择器只能配合着内容生成一起使用

3、属性 content

属性：content

作用：配合着 :before 或 :after，来插入生成的内容

常用取值：

字符串 : 纯文本

url : 图像

计数器 ：... ...

3、计数器

1、什么是计数器

通过 css 生成一些有序的数字，再通过内容生成将数字插入到某元素中去

2、属性 counter-reset

属性：counter-reset

作用：初始化(声明)/复位 一个计数器

注意：

1、初始值默认为 0

2、初始值可以为正也可以为负

语法：

xxx{

counter-reset:名称 初始值;

}

counter-reset 可以声明/复位多个计数器，中间用 空格 隔开

counter-reset:名称1 名称2;

counter-reset:名称1 10 名称2 20;

p{

counter-reset:c1;

}

<p>张无忌</p>

<p>张翠山</p>

<p>殷素素</p>

10 张无忌

20 张翠山

30 殷素素

计数器声明的位置：不能放在要使用计数器的元素中进行声明。

3、属性 counter-increment

作用：设置某个计数器使用时的增量，默认为1

注意：

1、可以设置正数(递增)或负数(递减)

声明位置：

哪个元素要使用计数器，就在哪个元素中声明增量

语法：

counter-increment:名称 增量值;

counter-increment:名称1 增量1 名称2 增量2;

body{

counter-reset:c1 0;

}

p{

counter-increment:c1 10;

}

<p>张无忌</p>

<p>张翠山</p>

<p>殷素素</p>

10 张无忌

20 张翠山

30 殷素素

4、函数 counter()

语法：counter(计数器名称);

必须配合:before 或 :after 以及 content 属性 一起使用

body{

counter-reset:c1 0;

}

p{

counter-increment:c1 10;

}

p:before{

content:"第"counter(c1)"名";

}

<p>张无忌</p>

<p>张翠山</p>

<p>殷素素</p>

10 张无忌

20 张翠山

30 殷素素

练习：

1、HTML

1.1 基本标签

1.2 结构标记

1.3 表单元素

2、CSS3

2.1 基础选择器

2.2 边框属性

2.3 高级选择器

4、多列

1、分隔列

作用：将文本分成竖排几列

属性：column-count

column-count:3;

2、列间隔

作用：设置列与列之间的间隔距离

属性：column-gap

column-gap:10px;

3、列规则

作用：设置每两列之间的间隔线的宽度 样式 颜色

属性：column-rule

column-rule:1px solid #f00;

4、浏览器兼容性

IE10+ 以及 Opera 支持多列的

Firefox ： -moz-

Chrome 和 Safari ：-webkit-

-webkit-column-count:3;

-webkit-column-gap:10px;

-webkit-column-rule:1px solid #ddd;

5、CSS Hack

1、解决问题

浏览器兼容性(IE)

2、CSSHack 原理

选择器和属性的优先级

3、CSSHack实现方式

1、CSS类内部Hack

属性名称前加前缀或在属性值后加后缀来适配不同的浏览器

2、选择器Hack

在选择器前加前缀，适配不同的浏览器

3、HTML头部引用Hack

通过 HTML 条件注释 解决问题

<!--[if 条件 IE 版本]>

可选择性被注释内容

在以上条件满足时，该位置代码可以被浏览器解析执行。

如果条件不满足的话，那么该位置的代码，就会被注释掉，不会执行。

<![endif]-->

条件：

gt : 大于条件版本以上版本的

gte : 大于等于条件版本以上的 <!--[if gte IE 8]>

lt : 小于条件版本以下

lte : 小于等于条件版本以下

! : 选择条件版本以外的其他版本

<!--[if !IE 8]>

<![endif]-->

# 第二天

1、转换

1 、转换简介

1、什么是转换

使元素改变形状、尺寸、或位置的一种效果

可以向元素应用 2D 和 3D方向的转换。能够对元素实现 旋转、缩放、位移以及倾斜的效果。

2D ：使元素在x轴和y轴上发生变化

3D ：在2D基础上增加对 z轴的变化

2、转换属性

1、属性

transform

2、取值

1、none

无转换效果，默认值。

2、transform-function

transform-funciton表示的是一个或者多个转换函数。如果是多个转换函数的话中间用 空格 分开

3、转换的原点

1、什么是转换原点

指的是转换操作所围绕的中心点。默认情况下，原点是在元素的中间位置处。x ：50%，y:50%

2、属性

transform-origin

取值：

两个值：表示x轴 和 y轴的位置。

三个值：表示原点在x轴，y轴和z轴上的位置。

方式：

1、数值 0px 0px 表示的是元素的左上点

2、百分比 原点位置占元素的宽或高的占比。

3、关键字

top:

bottom:

center:

left:

right:

2、2D转换

1、位移

位置移动

实现元素位置的改变。

1、属性 和 函数

属性：transform

函数：translate(x)

translate(x,y)

单向位移函数：

translateX(x)

translateY(y)

2、取值

数值 / 百分比

x ：取值为正，则向右移动，取值为负，则向左移动

y ：取值为正，则向下移动，取值为负，则向上移动

2、缩放

1、什么是缩放

改变元素的大小。可以让元素实现x轴缩放(改变宽度)以及实现y轴的缩放(改变高度)

2、属性 和 函数

属性：transform

函数：scale(value)

scale(x,y)

注意：如果只给一个值的话，那么第二个值与第一个值是相等的。

取值：

取的值表示的是缩放的倍数。

1 ：原始大小，即默认值。

大于 1 ：放大

0-1 ： 缩小

单向函数：

scaleX(x)

scaleY(y)

3、旋转

1、什么是旋转

使围绕着一个原点，使角度发生变化。

2、属性 和 函数

属性：transform

函数：rotate(ndeg)

n ：表示旋转的角度

n为正，则按顺时针旋转

n为负，则按逆时针旋转

3、注意

旋转操作是连同坐标轴一起都跟着转换。

4、倾斜

1、什么是倾斜

是的元素 向左右（x轴）或上下（y轴）的方向倾斜

2、属性和函数

属性：transform

函数：

skew(x) : x轴倾斜，向左右方向产生倾斜变换，实际上使得y轴按逆时针的角度产生倾斜效果。

skewX(x) : 同上

skewY(y) : y轴倾斜，向上下方向产生倾斜角度，实际上使得x轴按顺时针的角度产生倾斜效果。

skew(x,y)

3、3D转换

1、什么是3D转换

在2D的基础上，增加了对 z轴(空间轴)上的转换变化

2、perspective 属性

1、什么是perspective

假设 人眼 到投射平面的距离

2、注意

1、为元素定义 perspective属性时，其子元素会呈现3D转换的效果，而不是元素本身。

2、浏览器兼容性

Chrome 和 Safari 支持替代的 -webkit-perspective

3、3D-位移

1、什么是3D位移

在2D位移的基础之上，又增加了元素在 z轴上的位移距离。

2、属性 和 函数

属性：transform

函数：translateZ(z)

translate3D(x,y,z);

4、3D-旋转

1、什么是3D旋转

允许让元素以 x轴、y轴或z轴为中心进行的旋转操作。

2、属性 和 函数

属性：transform

函数：

rotateX(xdeg);

rotateY(ydeg);

rotateZ(zdeg);

rotate3D(x,y,z,ndeg)

x,y,z :

任何一个值，如果取值为 0 的话，那么该轴则不参与到选操作中。

如果大于0的话，指定的该轴要参与到旋转中，并且旋转的角度为 n 度

rotate3D(1,1,0,30deg)

5、3D-缩放

函数：

scaleZ(z)

scale3D(x,y,z)

6、transform-style 属性

作用：如何在3D空间中呈现被嵌套的元素。（子元素如何去摆放(2D或3D方式摆放)）

取值：

1、flat ：子元素将不保留其3D位置。（子元素按照2D的方式进行摆放），默认值

2、preserve-3d ：子元素将保留其3D位置（子元素允许按照3D的位置实现位置摆放）

2、过渡

1、什么是过渡

使得CSS属性值，在一段时间内的平滑过渡。

2、过渡 4要素

1、指定过渡属性

指定 哪个 CSS属性值在变化时使用过渡的效果

2、指定过渡所需时间

一个过渡效果在多长时间内完成

3、指定过渡函数(速度曲线函数)

4、指定过渡延迟

3、触发过渡

鼠标移入时 --> hover

鼠标点击时

4、过渡子属性

1、过渡属性

1、作用：指定哪个属性值在变化的过程中使用过渡进行显示。

2、属性

transition-property

值：

none ：不使用过渡效果

all : 任何一个属性值在发生变化时，都使用过渡效果

property : 具体的css属性名称

3、允许设置过渡的属性

1、颜色

2、取值为数值的属性

3、转换属性 : transform

4、渐变属性

5、visibility

6、阴影属性

eg :

p{

/\*宽度发生变化时，就采用过渡效果进行显示\*/

transition-property:width;

}

div{

/\*元素发生旋转时，采用过渡效果进行显示\*/

transition-property:transform;

}

2、过渡时间

1、作用：

一个过渡效果在多长的时间内完成。以秒(s) 或 毫秒(ms)作为单位

1000ms=1s;

2、属性

transition-duration

注意：默认值为 0，意味着没有过渡效果。所以过渡操作中，必须设置 transition-duration ,否则没有任何的显示效果。

3、过渡函数

属性：transition-timing-function

取值：预定义的函数值或贝塞尔曲线

预定义函数值：

ease:默认值，慢速开始，快速变快，以慢速结束

linear : 匀速

ease-in : 慢速开始，加速效果

ease-out : 快速开始，慢速结束

ease-in-out:慢速开始和结束，中间先加速再减速

transition-timing-function:linear;

4、过渡延迟

属性：transition-delay

取值: s | ms 为单位的数值

5、指定多个过渡效果

transition-property:background,width;

transition-duration:5s,3s;

transition-timing-function:linear;

transition-delay:1s;

6、过渡的简写属性

transition:property duration timing-function delay;

transition:width 2s linear,height 3s ease;

3、动画

1、什么是动画

指的是 使元素从一种样式逐渐变化为另一种样式的过程

注意：

浏览器的兼容性

Chrome Safari ：-webkit-

Firefox : -moz-

Opera : -o-

2、动画的使用步骤

1、声明动画

声明一个动画，并且指定 动画在指定 【时间段】 上的 【状态】 是什么

2、调用动画

指定 调用 哪个已声明好的动画，并且同时指定动画执行参数。

3、声明动画

语法：

@keyframes 动画名称{

/\*时间点 ，状态\*/

0% | from{

开始时的CSS样式;

}

...

25%{

动画执行 1/4 时 ，元素的状态定义;

}

...

100% | to{

结束时的CSS样式;

}

}

兼容性：

Chrom Safari

@-webkit-keyframes name{}

Firefox

@-moz-keyframes name{}

4、调用动画(动画属性)

动画：animation

1、animation-name

指定调用动画的名称

2、animation-duration

动画完成一个周期所需要用的时间 s | ms 为单位

3、animation-timing-function

执行动画的时间速度曲线函数

ease,linear,ease-in,ease-out,ease-in-out

4、animation-delay

播放之前的延迟

5、animation-iteration-count

动画播放次数，默认为1

取值：

1、具体数值

2、infinite(无限次播放)

6、animation-direction

动画播放方向

取值：

1、normal

默认值，正常播放，从 0%~100%

2、reverse

反向播放，从100% ~ 0%

3、alternate

轮流播放

奇数次数 ：为正向播放

偶数次数 ：为反向播放

7、animation 属性

通过该属性设置动画的所有子属性

animation:name duration timing-function delay iteration-count direction;

8、animation-fill-mode

动画在播放前以及播放后的填充模式

取值：

1、none

默认值

2、forwards

当动画完成后，保持在最后一个属性状态上(100%)

3、backwards

动画播放前，显示最开始的属性值。

注意：在动画播放延迟时，有效。

4、both

播放前后 均采用填充模式

9、animation-play-state

动画的播放状态

取值：

paused : 动画暂停

running : 动画播放(默认值)

注意：以上属性，在使用过程中，根据不同浏览器添加不同前缀。