# CSS的继承非常简单，比如



其中.btn 是一个css类

.btn-primary是一个css类

然后后面的realname是自己定义的css类就行了 他就会一步一步继承下去，很实用

# 经典的选择器：

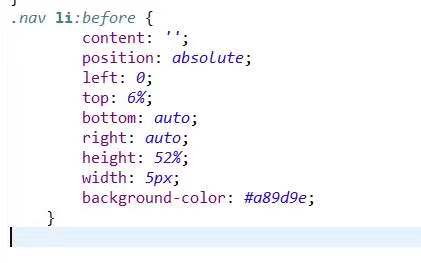
nth-child(x){ }

# 伪类



伪类设置出来的border很实用（自己体会，很简单）：

要定义放在那一方，就把那一方设置成0，对应方设置成auto



# 3.css3规定字体方法：

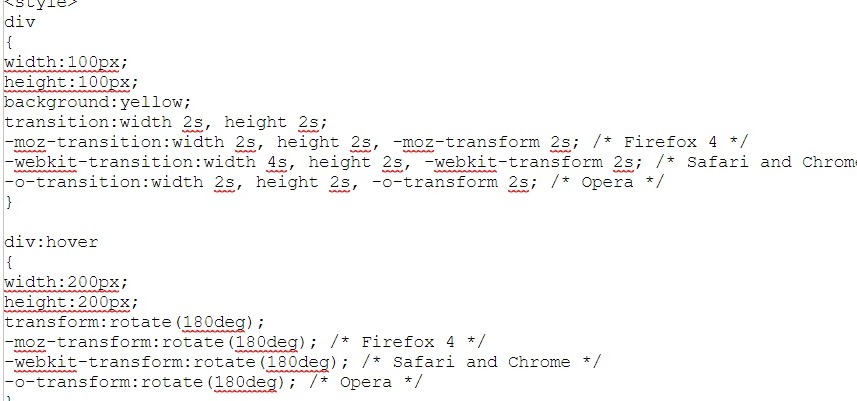


# 4.2D改变的总结



# 5.动画开始！！！

## 1.过渡（transition）：



就是属性的过度 这个过程就是动画

常用的过度效果：



### 实例

一个最简单的实例：



-webkit-transition:width 1s linear 2s;

第一个是变换的属性名称 第二个是过渡时间 第三个是过渡的方式 第四个是delay延迟时间

其中第三个很重要 过渡样式的选择提供的有5种 可以自定义

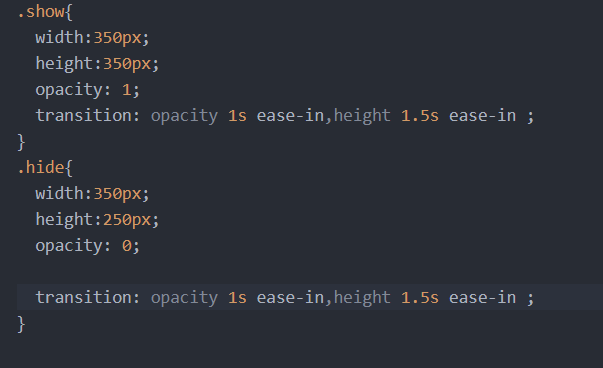
### 多项同时改变（长宽等同时）：



transition里面定义的是变什么，变多久

至于变成什么样子那是js或者另一个css干的事（自己的demo里有）

transition要多重比如opacity和height的话要写在一个transition里面如下图



## 2.函数式包裹动画（animation： from to）



这个就是定义动画的from to 和jq差不多（定义在css中）

@-webkit-keyframes xxx 是动画名称

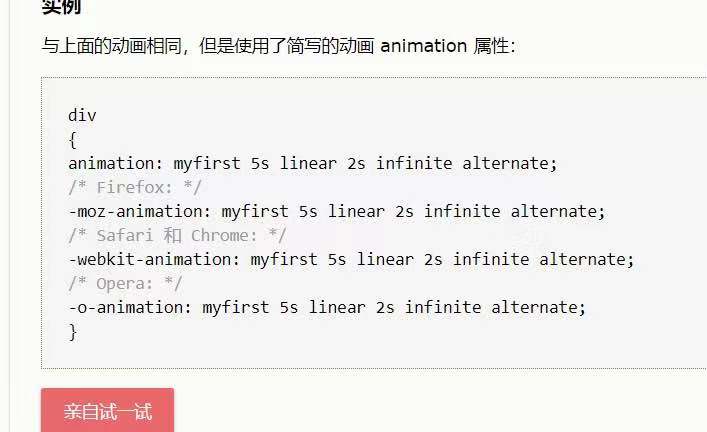
### 实例使用：



同样 定义在原本div上要说明执行的是哪个动画 以及执行的时间

时间必须要加 不然默认为0 不起作用

### 拓展效果：（效果，重复，延时）



-webkit-animation: myfirst 5s linear 2s infinite alternate;

第一个是指定运行的动画名称 第二个是过渡时间 第三个是过渡样式 第四个是delay延迟播放时间 第5个是动画的播放次数

次数默认是1 或者自己设置 或者设置成infinite

infinite无限次数播放

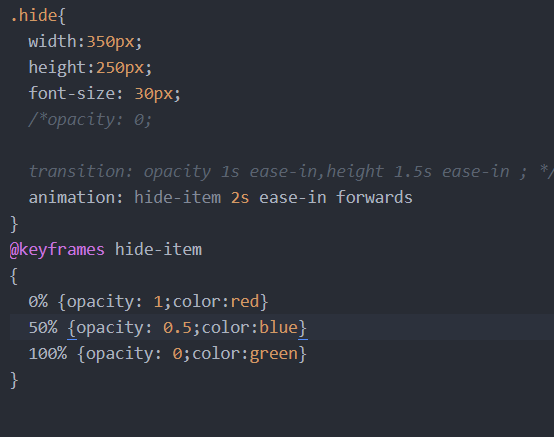
第六个是是否播完一次后逆向播放 类似回退 两个选项 一个是none 一个是alternate

一个是alternate一个是normal

Alternate是逆向播放，normal是正常播放

### 多步骤组合使用：





动画的含义是从xxx的选中样式改变到xxx

上一个CSS是不变的

变化到当前CSS时经过animate的过渡改变才成型

### 动画结束后保持状态：



animation-fill-mode:forwards;  这句话写在css中

原文：<https://blog.csdn.net/lp2659359879/article/details/52523888>

# 6.Div外轮廓



这个轮廓可能有点用

轮廓可能是圆形哦

# 7.Overflow：

overflow把多余的部分隐藏 然后鼠标放上去整个div往上移，

就变成了类似轮播效果

自己demo里有

C:\Users\MACHEN~1\AppData\Local\Temp\WeChat Files\96cd238ed8e5279fee30fa9c2bb2ce8.jpg

# 8.样式重叠：



# 9.Img自适应DIV容器大小



img自适应容器

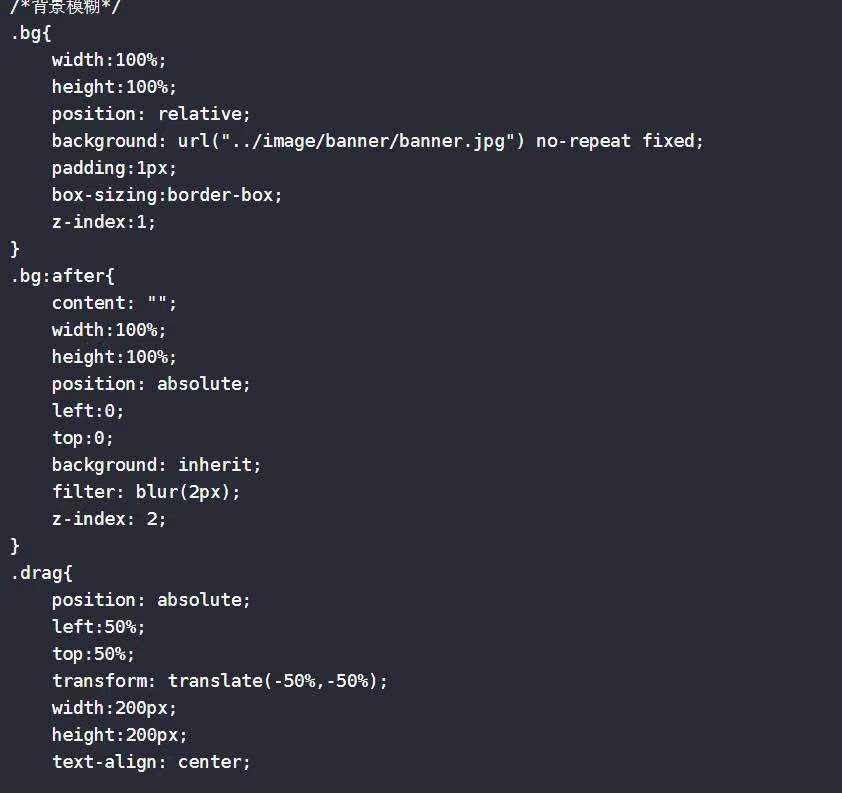
# 10.CSS3线性渐变功能（炫）：



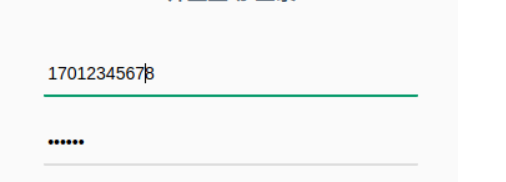
# 11.blur模糊背景的应用：

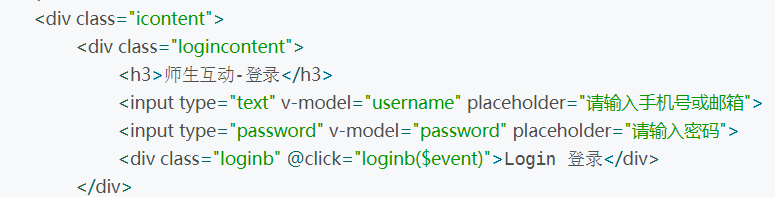
一般用的伪类来做的模糊块儿





# input标签的改装：

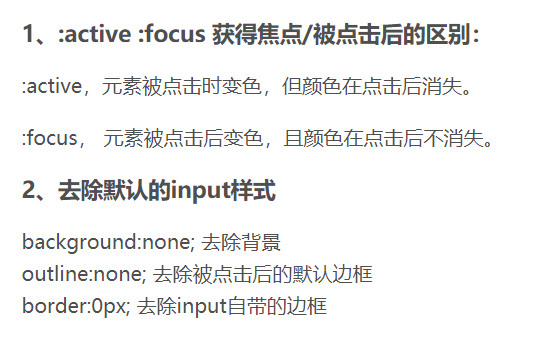




改装部分：

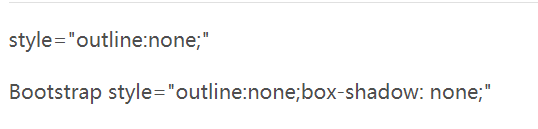


一般的操作就这些就可以自定义input了



主要的操作就有以上这些，

然后bootstrap那些恶心的边框模糊其实是box-shadow!!!!

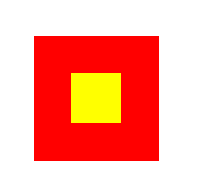


# button,input点击之后的边框去除使用：

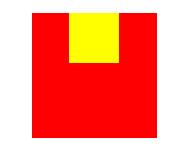
outline:*none*;直接加在style里

# 关于absolute和input(file)框的美化心得

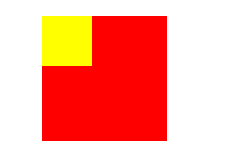
在元素被设置为absolute之后，其元素的top和left都是以父元素为边界开始的，即absolute之后，top:0就是顶部挨着父级元素的顶部，left:0就是挨着父级元素的最左边

这个本来是居中的。

设置为absolute，top:0之后

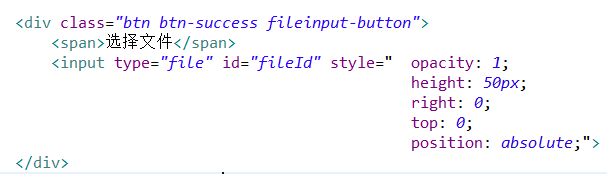
从父元素顶部边界开始。越大就越离上边界越远。

设置为absolute, top:0, left:0之后

从父元素左边边界开始。越大越离左边界越远

所以懂了这个之后 关于input的美化就是

将input用div包裹起来，将其input元素锁定在自己的div内，然后将其  
opacity设置成0（透明），然后在自己的div就可以随意设置样式了。



覆盖的思路