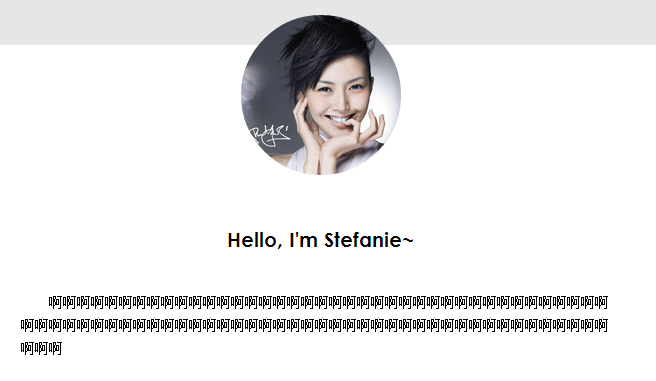
Baidu-ife

参考链接：https://www.anotherhome.net/1969

Task0001-7总结：

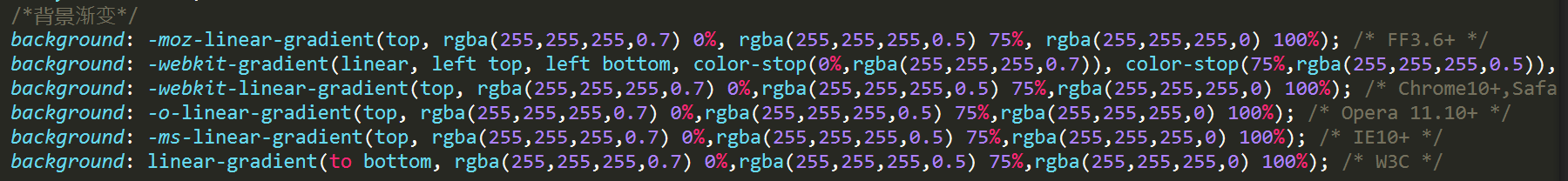
Index.html:

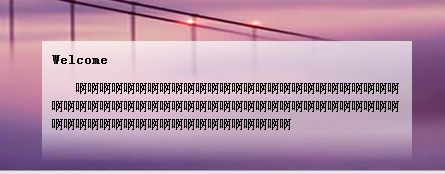
1. 对于每个页面，一般其header和footer都是通用的，所以我们在定义header和footer时，把它的css样式都放在style.css文件中，这样在创建别的页面时，只要引入css文件即可调用，实现了复用性，不用再写一遍；
2. 设置超链接，当鼠标移动上去时，可能样式有一定改变，可以用\*\*\*:hover{}来设置；
3. 头像的设置



设置上边距margin-top：-30px

1. 设置背景渐变



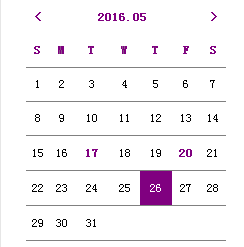


Blog.html:

1. 对于某些看上去较为复杂的界面，其实使用margin负边距来设置的；



1. 日历的设置方法：设置是一个table，每个小格有自己的style来设置



Gallery.html:

1. 图片流设置，分为两块



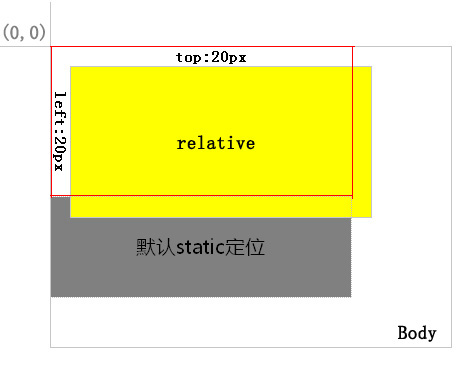
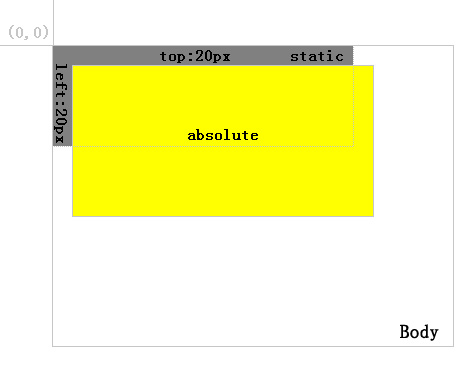
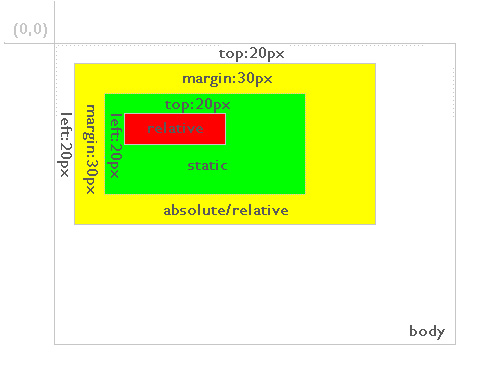
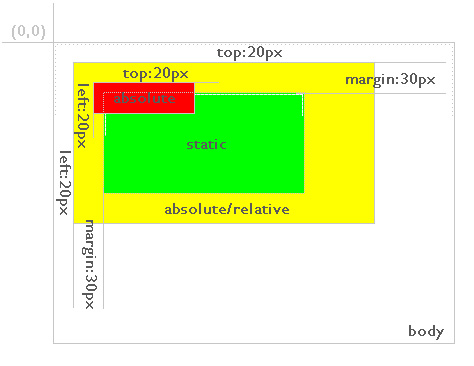
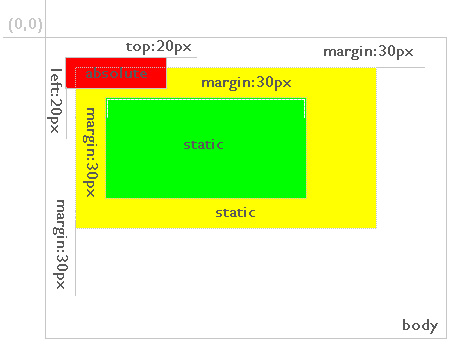
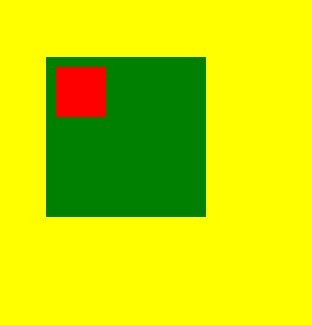
先分为right和left两块，在分别在两部分内在设置其小的img框架；

About.html:

1. 

这个content里面，先在中间插入一条宽度10px的灰色竖线，然后再分别设置每个简介的模块：

[**CSS+DIV布局中absolute和relative**](http://selfcontroller.iteye.com/blog/1826119)

这里向大家简单介绍一下CSS+DIV布局中absolute和relative属性的用法和区别，定位为relative的元素脱离正常的文本流中，但其在文本流中的位置依然存在，而定位为absolute的层脱离正常文本流，但与relative的区别是其在正常流中的位置不在存在。   
  
在用CSS+DIV进行布局定位的时候，一直对position的四个属性值relative,absolute,static,fixed分的不是很清楚，以致经常会出现让人很郁闷的结果。今天研究了一下，总算有所了解。在此总结一下：   
  
先看下position各个属性值的定义：   
  
1、static：默认值。没有定位，元素出现在正常的流中（忽略top,bottom,left,right或者z-index声明）。   
  
2、relative：生成相对定位的元素，通过top,bottom,left,right的设置相对于其正常位置进行定位。可通过z-index进行层次分级。   
  
3、absolute：生成绝对定位的元素，相对于static定位以外的第一个父元素(如果直接父元素没有position属性，则继续向上寻找有position的父元素，直到找到含有position的元素，并根据这个元素进行定位，如果所有父元素都没有这个属性，则根据body进行定位)进行定位。元素的位置通过"left","top","right"以及"bottom"属性进行规定。可通过z-index进行层次分级。   
  
4. fixed：生成绝对定位的元素，相对于浏览器窗口进行定位。元素的位置通过"left","top","right"以及"bottom"属性进行规定。可通过z-index进行层次分级。   
  
static与fixed的定位方式较好理解，在此不做分析。下面对应用的较多的relative和absolute进行分析：   
  
1、relative。   
  
定位为relative的元素脱离正常的文本流中，但其在文本流中的位置依然存在。如图1：   
  
   
黄色背景的层定位为relative，红色边框区域为其在正常流中的位置。在通过top、left对其定位后，从灰色背景层的位置可以看出其正常位置依然存在。   
  
2、absolute。   
  
定位为absolute的层脱离正常文本流，但与relative的区别是其在正常流中的位置不再存在。如图2：   
  
   
可以看到，在将黄色背景层定位为absolute后，灰色背景层自动补上。   
  
3、relative与absolute的主要区别：   
  
首先，是上面已经提到过的在正常流中的位置存在与否。   
  
其次，relative定位的层，总是相对于其最近的父元素，无论其父元素是何种定位方式。如图3：   
  
   
图中，红色背景层为relative定位，其直接父元素绿色背景层为默认的static定位。红色背景层的位置为相对绿色背景层top、left个20元素。而如果红色背景层定位为absolute，则情形如图4：   
  
   
可以看到，红色背景层依然定义top:20px；left:20px；但其相对的元素变为定位方式为absolute或relative的黄色背景层。因此，对于absolute定位的层，总是相对于其最近的定义为absolute或relative的父层，而这个父层并不一定是其直接父层。如果其父层中都未定义absolute或relative，则其将相对body进行定位，如图5：   
  
   
除top、left、right、bottom定位外，margin属性值的定义也符合上述规则。   
  
absolute：绝对定位，它的定位分两种情况，如下：   
  
1、 有设定 Top、Right、Bottom、Left 的情况，这里又分了两种情况如下（如上所述）：   
  
      (1). 父级（并不单单指直接父级，如果直接父级没有position属性，则继续往上查找有position的的父级元素）没 position 属性，浏览器左上角(即 Body)为“坐标原始点”进行定位，位置由 Top、Right、Bottom、Left 属性决定。   
  
      (2). 父级有 position 属性，父级的“坐标原始点”为原始点   
  
2、 没有设定 Top、Right、Bottom、Left 的情况，默认依据父级（直接父级）的“内容区域原始点”为原始点(如果父级 Padding 属性，“坐标原始点”和“内容区域原始点”不一样)，如图：   
  
   
代码如下：

**Html代码  [收藏代码](javascript:void())**

1. **<div** style="position:relative; top:20px; right:20px; width:300px; height:300px; background:yellow;padding-top:30px;padding-left:30px;"**>**
2. **<div** style="padding:10px 0px 0px 10px;margin:30px 0px 0px 30px;background:green;width:150px; height:150px"**>**
3. **<div** style="position:absolute; width:50px;height:50px;background:red"**>**
4. **</div>**
5. **</div>**
6. **</div>**