1.制作雪碧图

切图，将小图标上传到<https://www.toptal.com/developers/css/sprite-generator>制作雪碧图。可在右侧选择图标排列方式。

2.html页面框架

根据效果图将页面分为三部分

3.设置header及footer部分

小图标定位，主要用到background-position属性。

4.设置内容部分

利用flex使价格及时间行垂直居中，为价格块设置margin-left: auto;使其右对齐

border-image: linear-gradient(white,#999999,white)4;

border-image:linear-gradient (to right, rgba(255,0,0,0), rgba(153,153,153,0.8), rgba(153,153,153,1), rgba(153,153,153,0.8), rgba(255,0,0,0))

background: url(./images/css\_sprites.png) no-repeat 0 0/5.7rem 16.3rem ;

问题

设置html{font-size:62.5%；}换算rem不生效，测试后发现是在chrome浏览器中有一个问题是字体小于12px统一都按12px

因此设置html{font-size:625%；}，其他单位换算时都统一除以100

由于效果图是二倍图，设置背景图片大小时缩放了，导致定位时出现问题

使用雪碧图时，背景图片大小设置为原图大小，容器大小也无需缩放，在最后添加transform:scale(0.5);属性。

内容部分使用transform:scale(0.5);进行缩放出现错位，添加属性transform-origin: center bottom;设置缩放基准点。改变基准点可以实现小图标和文字对齐，但是由于占据的原始尺寸不变而出现难以垂直居中的情况。

查找资料后发现zoom属性虽然可以解决占位问题，但只有ie支持，因此只能设置backg-size等比例缩小，同时改变容器大小和位置。

下拉选择框用select实现时，option的宽度总是比select宽，因此需要用ul li实现

zoom属性（ie专用）

**zoom**：normal | [<number>](http://www.w3chtml.com/css3/values/numeric/number.html) | [<percentage>](http://www.w3chtml.com/css3/values/other/percentage.html)

**normal：**使用对象的实际尺寸。

[**<number>**](http://www.w3chtml.com/css3/values/numeric/number.html)**：**用浮点数来定义缩放比例。不允许负值

[**<percentage>**](http://www.w3chtml.com/css3/values/other/percentage.html)**：**用百分比来定义缩放比例。不允许负值

zoom与transfrom：scale还存在区别，：

zoom的缩放是相对于左上角的；而scale默认是居中缩放；

zoom的缩放改变了元素占据的空间大小；而scale的缩放占据的原始尺寸不变，页面布局不会发生变化；

zoom和scale对元素的渲染计算方法可能有差异（需要自己动手，用高清图，仔细去看其中的区别）。

对文字的缩放规则不一致。zoom缩放依然受限于最小12像素中文大小限制；而scale就是纯粹的对图形进行比例控制，文字50%原来尺寸。

flex设置最后一块右对齐，为该块添加margin-left: auto;

为plat添加阴影利用box-shadow属性

box-shadow: h-shadow v-shadow blur spread color inset;

h-shadow 必需。水平阴影的位置。允许负值。

v-shadow 必需。垂直阴影的位置。允许负值。

blur 可选。模糊距离。

spread 可选。阴影的尺寸。

color 可选。阴影的颜色。

inset 可选。将外部阴影 (outset) 改为内部阴影。

昨天日报提到的问题 z-index无效，是由于Z-index 仅能在定位元素上奏效

main的position:absolute属性改为margin后，增加内容至出现滚动条后，go图标会位于最上层进而遮挡header栏的图标。nav标签的内容也处于最上层。因此仍采用position:absolute。

为了将nav栏固定于页面头部，将nav作为header的一部分，为main层设置z-index=-1使其位于下层。

css实现倒三角

1. .triangle{
2. border-left: 0.05rem solid transparent;
3. border-right: 0.05rem solid transparent;
4. border-top: .05rem solid  #999999;
5. }

task-6优化

1.减少div标签的使用

头部header和nav放入p标签中

部分div改为button

小三角形位置微调

将部分div替换为p标签时了解了一下各标签的嵌套问题

1.块级元素（block）与内联元素（inline）的区别：

块级元素用来搭建网站架构、布局、承载内容，负责整体的大的方面。块级元素总是会另起一行开始，其高度、行高和边距都是可控的。内联元素负责局部和细节，在同一行显示，其高度、行高和边距不可控。

通过display：block;或者display:inline两者可以相互转化。

可变元素是可以根据上下文语境决定是块级元素还是内联元素。

2.分类

**块元素**(block element) HTML标签分类明细

address - 地址

blockquote - 块引用

center - 居中对齐块

dir - 目录列表

div - 常用块级容易，也是css layout的主要标签

dl - 定义列表

fieldset - form控制组

form - 交互表单 （只能用来容纳其它块元素）

h1 - 大标题

h2 - 副标题

h3 - 3级标题

h4 - 4级标题

h5 - 5级标题

h6 - 6级标题

hr - 水平分隔线

isindex - input prompt

menu - 菜单列表

noframes - frames可选内容，（对于不支持frame的浏览器显示此区块内容

noscript - 可选脚本内容（对于不支持script的浏览器显示此内容）

ol - 排序表单

p - 段落

pre - 格式化文本

table - 表格

ul - 非排序列表

**内联元素**(inline element) 一般都是基于语义级(semantic)的基本元素，只能容纳文本或者其它内联元素。

a - 锚点

abbr - 缩写

acronym - 首字

b - 粗体(不推荐)

bdo - bidi override

big - 大字体

br - 换行

cite - 引用

code - 计算机代码(在引用源码的时候需要)

dfn - 定义字段

em - 强调

font - 字体设定(不推荐)

i - 斜体

img - 图片

input - 输入框

kbd - 定义键盘文本

label - 表格标签

q - 短引用

s - 中划线(不推荐)

samp - 定义范例计算机代码

select - 项目选择

small - 小字体文本

span - 常用内联容器，定义文本内区块

strike - 中划线

strong - 粗体强调

sub - 下标

sup - 上标

textarea - 多行文本输入框

tt - 电传文本

u - 下划线

var - 定义变量

**可变元素**是基于以上两者随环境而变化的，它的基本概念就是根据上下文关系确定该元素是块元素还是内联元素。一旦确定了它的类别，它就遵循块元素或者内联元素的规则限制。

applet - java applet

button - 按钮

del - 删除文本

iframe - inline frame

ins - 插入的文本

map - 图片区块(map)

object - object对象

script - 客户端脚本

3.块级元素和内联元素的嵌套规则：

最基本：内联不能嵌套块级，块级可以嵌套内联元素

<div><h1></h1><p></p></div> 正确（块级并列）

<a href="#"><span></span></a> 正确（内联嵌套内联）

<span><div></div></span> 错误（内联嵌套块级）

有几个特殊的块级元素只能包含内嵌元素，不能再包含块级元素，这几个特殊的标签是：h1、h2、h3、h4、h5、h6、p、dt。所以说p里面不能嵌套div。

特殊的<li>里面可以嵌套div（<li>可以但<p>不行）

块级元素与块级元素并列、内联元素与内联元素并列

<div><h2></h2><p></p></div> 正确

<div><a href="#"></a><span></span></div> 正确

<div><h2></h2><span></span></div> 错误（块级和内联并列了）

今天完成的事情

这几天公司的事要写app界面，就用纯css写了几个页面拼凑了一下，也算复习了一下之前学的，一直没来得及做任务