**软件工程专业综合实训报告**

**京东首页制作**

班级：软件1402班

学号：2014234060518

姓名：何冰倩

2017年 10 月 8 日

[**一、实训项目概述** 2](#_Toc495588789)

[**1.** **实训任务** 2](#_Toc495588790)

[**2.** **实训目的** 2](#_Toc495588791)

[3. **应用技术** 2](#_Toc495588792)

[**二、实训内容** 2](#_Toc495588793)

[**1．** **开发环境配置** 2](#_Toc495588794)

[**2．** **技术方案** 2](#_Toc495588795)

[**3．** **设计** 3](#_Toc495588796)

[**4．** **实现** 3](#_Toc495588797)

[**5．** **测试** 8](#_Toc495588798)

[**三、实训总结** 8](#_Toc495588799)

**一、实训项目概述**

1. **实训任务**

本次实训主要完成了京东首页的静态实现

1. **实训目的**

本次实训意在练习复杂页面的布局，多种轮播图的原生JavaScript实现，J avaScript的BOM和DOM对象操作，ES6新特性的练习，JQuery插件的使用以及封装属于自己的JS库。

1. **应用技术**

该实训项目主要使用了HTML，CSS，JavaScript，JQuery等语言，其中使用了HTML5，CSS3以及ES6的部分内容。

**二、实训内容**

1. **开发环境配置**

因为项目主要使用HTML，CSS，JavaScript以及JQuery并不需要复杂的开发环境，所以这里主要写一下Babel的配置（[Babel](https://babeljs.io/)是一个广泛使用的转码器，可以将ES6代码转为ES5代码，从而在现有环境执行），或者使用Babel在线转换 <https://babeljs.io/repl/>

1. 首先安装babel-cli（用于在终端使用babel）

npm install -g babel-cli

1. 然后安装babel-preset-es2015插件

npm install --save babel-preset-es2015

注：Babel5版本默认包含各种转换插件，然而Babel6.x相关转换插件需要自己下载，如transform-es2015-arrow-functions、transform-es2015-classes等，而ES2015 preset包含了所有插件。如果不安装任何插件，那么在命令行进行转换是没有任何效果的！其中--save参数自动更新package.json文件，写进依赖项。

1. 在命令行输入：

babel es6.js --presets es2015

输出：

　　 "use strict";

　　 [1, 2, 3].map(function (x) {

　　 return x \* x;

　　 });

注：后面的参数--presets es2015表示使用该插件进行编译，如果不写上转换是没有效果的。

1. 插件配置

每一次都写上该参数那是很麻烦的，可以在当前目录下新建配置文件 .babelrc。

但是在windows系统中，不允许直接右键建立没有文件名的文件，可以通过cmd命令行创建：

在当前文件夹打开cmd并键入命令

type nul>.babelrc

即可在当前目录下建立文件.babelrc，接着在文件中写入：

{

　　"presets": ['es2015']

}

那么就可以直接在命令行中使用babel es6.js进行转换而无需添加表明所用插件的参数

除了建立.babelrc文件之外，也可在package.json中进行配置，添加以下属性即可：

"babel": {

　　"presets": ["es2015"]

}

1. **技术方案**

该实训项目主要使用了HTML，CSS，JavaScript，JQuery等语言，其中使用了HTML5，CSS3以及ES6的部分内容。

1. **设计**

无论页面布局是简单还是复杂，我们需要做的第一步必定是分析布局。该项目页面内容较多，细节较多，我们可以大而化小，逐步细化。

首先我们把整个页面划分为头部，主体和底部，然后主体部分根据内容可以又分为若干个块。

1. **实现**

页面实现部分，这里主要描述布局以及难点，其余不再赘述。

1. **顶部**

**顶部效果图如下**：



**改变浏览器宽度效果如下**：



这里顶部由于浏览器宽度发生变化，布局也产生了变化。这里可以使用JavaScript，通过监听document对象的宽度，以实现分别布局。或者是使用CSS3中的媒体查询方法，这也是我选择的方法（下面各处媒体查询原理相同，不再赘述）。

这里将顶部分为两个div，最顶部多个选项卡为一个div（div1），其余为一个div（div2）。

1. **div1**

顶部多个选项卡实际也是多个超链接，其功能以及实现为：

1. 鼠标划入显示子菜单，划出子菜单消失。这里可以选用JavaScript通过设置鼠标划入划出事件，来设置元素隐藏和显示，或者采用JQuery的hide与show方法，更简单的有JQuery的hover方法。我在这里采用的是CSS3的hover方法，这个方法较以上两种加载速度更快。
2. 第一个选项卡有地址选择功能，点击后更改选项卡的内容。这里使用原生JavaScript实现，目前没有箭头移动改变选中项的功能，后续会添加。
3. **div2**

我们根据内容又将div2分解成若干小块。值得注意的是搜索框中的关键字条，这里是一个自动切换，同时有鼠标点击后清空搜索栏的小细节，不容忽视。这里的功能使用原生JavaScript的定时器就能轻松实现。

1. **主体**
   1. **模块1**

**效果如图**



这里将模块一分为经典的左中右三栏布局（浏览器宽度变化，导致布局变化，同上采用媒体查询不再赘述）：

* + 1. **左**

功能选项卡，鼠标划入每行，即会显示相应的子菜单，鼠标移出，子菜单消失。同顶部div1实现类似，这里也是选用了CSS3的hover方法。

* + 1. **中**

中间又分为上下两部分，上面是一个最基本的轮播图，下面又分为两小块。这里主要叙述轮播图的实现。

**轮播图如下**：



**轮播图的功能有：**

a.每隔若干秒自动切换图片，同时图片对应的下方小红点，会发生相应移动。b.鼠标移入轮播图，图片停止自动播放，同时两侧按钮从隐藏变为显示。c.点击按钮，图片会向上一张或者下一张切换；点击下方白点，图片可以定位到指定图片。d.鼠标移出轮播图若干秒后，轮播图就当前图片继续向下一张循环轮播。

**实现：**

这里轮播图的功能看似复杂，实际抓住关键并不难。

这里的关键即图片和白点是绑定的，如果它们存放于两个数组，我们可以利用数组的index属性。

1.设置定时器，循环播放图片以及移动白点。

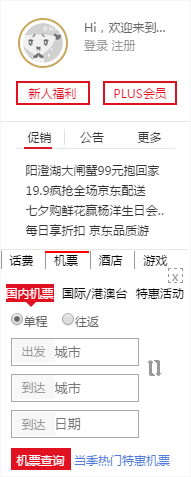
2.获取到图片对象以及下面白点对象数组，当鼠标移入时，关掉JS的定时器，鼠标移出是重现开启定时器。

3.对向前和向后按钮，以及白点对象数组添加相应事件。

* + 1. **右**

右侧又分为上中下三部分，这里着重叙述底部功能与实现。

**效果图如下：**



**功能：**

* + - 1. 当鼠标分别移入话费，机票，酒店和游戏选项时，子菜单从底部向上划入，选项页面向上消失，点击关闭按钮以相反方向还原。
      2. 在子菜单中，切换选项卡，左右无缝切换。
      3. 点击输入框，可以自动选择出发城市和日期等。

**实现：**

a. 为了实现这种无缝划入效果，需要把选项部分和菜单拼接，这里我们将这两部分放在一个盒子中，通过绝对定位控制其移动。由于是渐进效果，这里使用了个人JS库中的运动函数。由于划入选项不同，拼接的菜单也不同，我们可以在鼠标划入时，通过JS控制菜单显示的内容。以上也可以通过jQuery实现。

b. 无缝切换与一般切换，最大的区别在于，无缝切换可以看到当前目标和下一个目标同时运动。因此我们可以采用a中的定位移动办法。

c. input输入框的选项问题，这里采用了jQuery的 datePicke时间选择插件以及自己写的的地点选择插件。效果如图：



* 1. **模块2（秒杀部分）**

**效果图如下：**

****

**功能:**

* + - 1. 倒计时时钟显示
      2. 左边为左右滑动无缝轮播图，右边为普通轮播图
      3. 鼠标移入轮播图的图片时，图片向上移动，鼠标移出时还原。

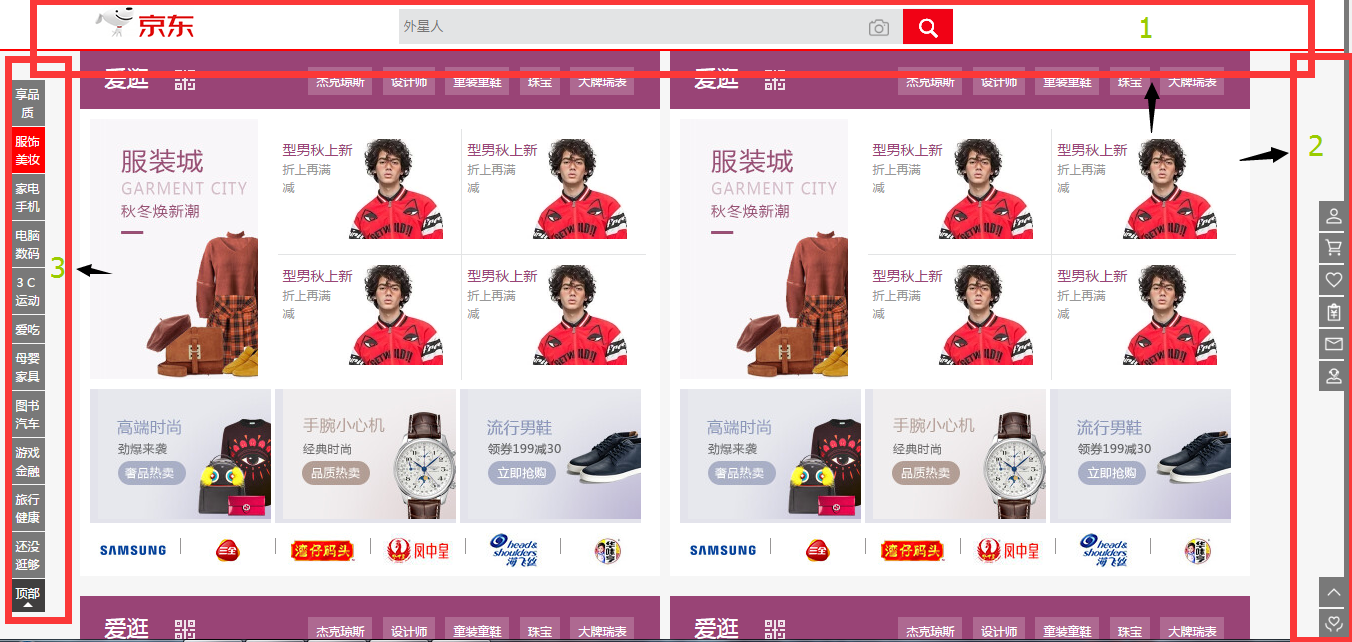
**实现：**

1. 这里倒计时时钟，由JS的定时器设置，每秒重新获取一次时间，并重现设置。
2. 左右滑动无缝轮播图，原理即定位控制法。最大的难点在于首尾的无缝切换。在这里我们将第一个对象复制一份（记为副本1），添加到末尾。每次当轮播到副本1时，用CSS将轮播图定位到第一个对象的位置，因为CSS定位速度特别快，几乎肉眼不可见，这里便实现了无缝且顺序正确。
3. 这里使用个人JS库的运动函数。将运动量设置为图片的margin值或者定位top值等（以下图片运动类似，不再赘述）。
   1. **模块三**

**功能：**

当滑动轮距离页面顶部若干距离后，会显示顶部搜索框1，以及侧边索引栏3。2部分固定，不随页面变化。同时3部分添加了索引的功能，点击相应的模块会快速索引到目的地。

**效果如图：**

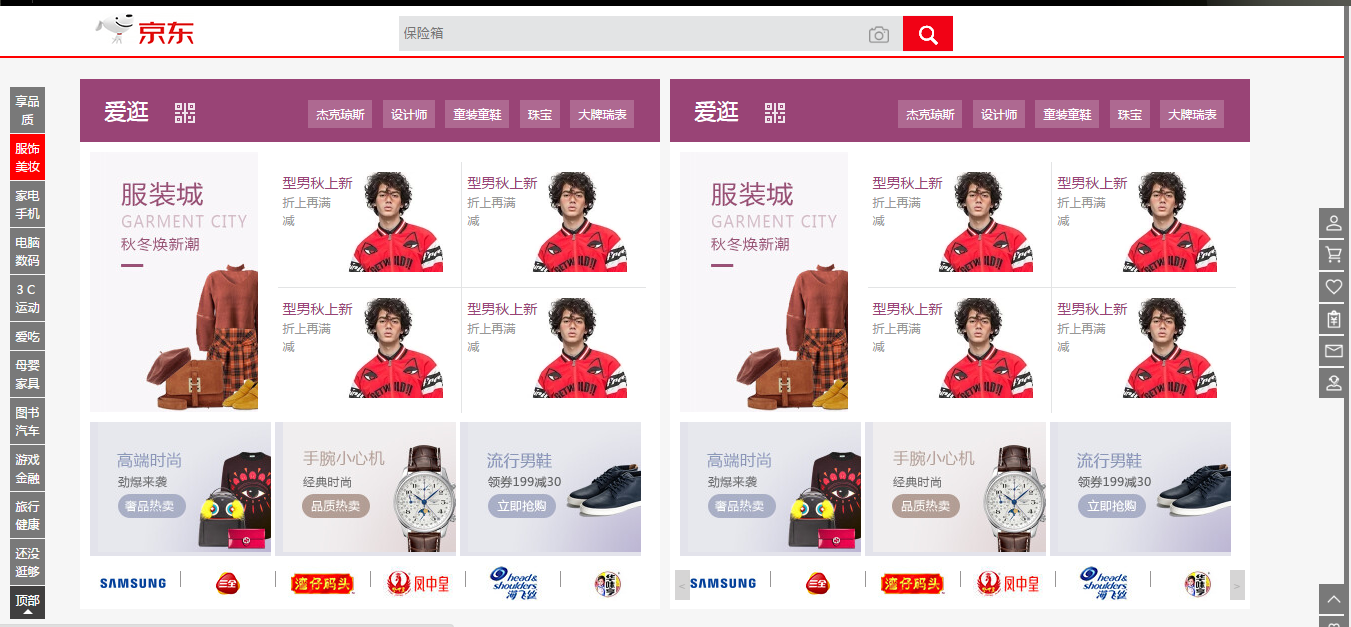
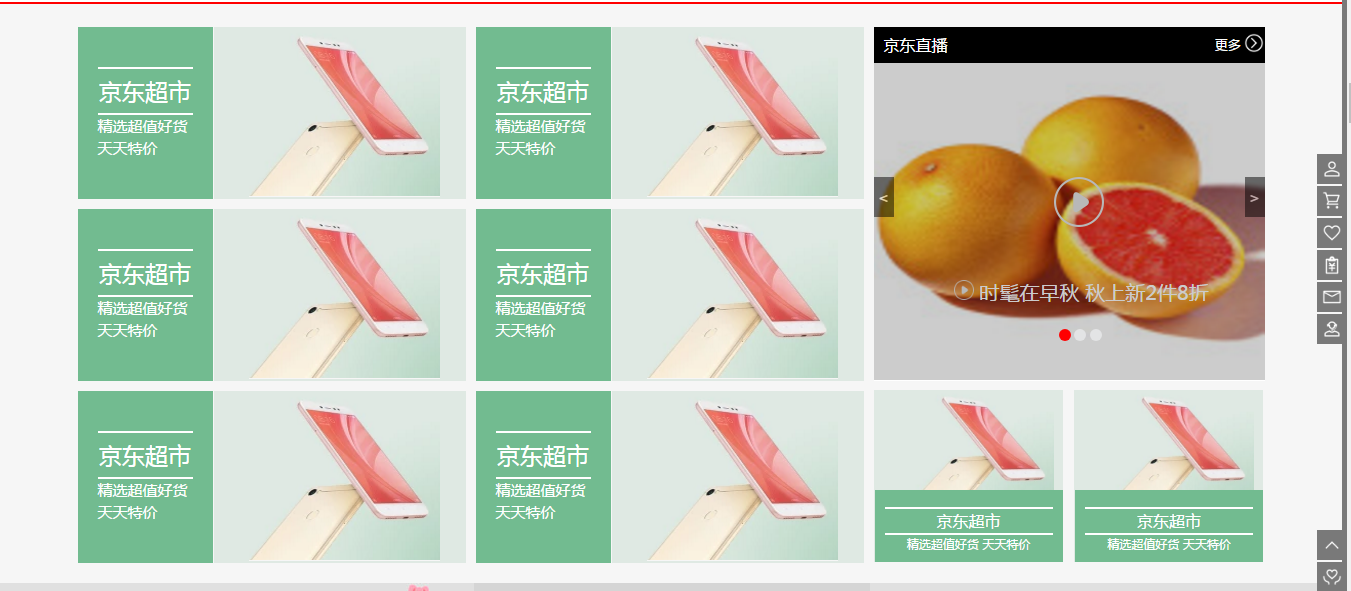


**实现：**

1. 1部分通过jQuery的scrollTop方法，获取document当前距离顶部的距离，从而实现对1部分显示或隐藏的控制。
2. 2部分由于不随页面变化而变化，故而使用fix定位。
3. 3部分的显示和隐藏，同1部分一样。3部分的索引功能，通过jQuery改变document的scrollTop值来实现。这时我们需要估算每个模块大概的高度，从而进行定位。
   1. **模块四**

要点在上文已说，不再赘述。

**效果图如下：**

****

1. **底部**

底部只是复杂布局，细节较多。

**效果图如下：**

****

1. **测试**

多个轮播图，各处媒体查询在非IE6显示正常。兼容IE6正在学习中。

**三、实训总结**

这次实训教给我很多东西，不仅是知识上的丰富，更是内心的充盈。说实话，这个项目还不完善，很多地方小细节并没有做到最好，函数封装尚未彻底。即便如此，只要开始了便有收获。开始这个项目的时候，我的内心是拒绝的。复杂的页面写过，但是没有写过这么复杂详细的页面。万事开头难，当我沉下心开始写的时候，才发现写这种页面，不仅能熟练学过的知识，还能带来内心的充实感。同时我发现在练习、实践、总结这个过程中，我会逐渐明确需要补充什么知识，下一步需要学习什么。感谢这次实训，希望自己越来越好。