



第三十四天到第三十六天： 我是精明的小卖家（二）

进阶

👤 Varsha (/mentor/detail/id/2) | 🏠 零基础学院 (/college/detail/id/5)

开始学习

人数 有 160 人在学习该课程，有 114 人已经完成该课程

时间 平均用时 3.5 天

关键词 JavaScript

课程概述

作业提交截止时间：09-01

第三十四天到第三十六天 我是精明的小卖家（二）

课程目标

今天我们将学习SVG和Canvas，来继续丰富我们的销售报表

课程描述

需求描述



当上一课任务完成时，面对一个复杂数据的表格，估计没有人会仔细看这份数据的内容是什么，看也看不出什么。所以我们希望找到某种方式，让数据变得更加易读和易懂，而数据可视化正式解决这一问题的最佳答案。

在表格上方增加两个图表，一个折线图和一个柱状图，用于展现不同数据在12个月的销售情况。

先了解一下数据可视化

简要阅读下面两篇文章了解数据可视化

- 数据可视化百科词条
(<https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8F%AF%E8%A7%86%E5%8C%96/1252367?fr=aladdin>)
- ECharts数据可视化规范 (<http://vis.baidu.com/chartusage/overview/>)

小练习

阅读

我们将通过SVG及Canvas来分别实现折线图及柱状图，我们先来看看 SVG，使用SVG来实现我们的柱状图

- W3School SVG课程 (<http://www.w3school.com.cn/svg/index.asp>)
- MDN SVG (<https://developer.mozilla.org/zh-CN/docs/Web/SVG>)

小练习编码

这个练习的代码和报表页面无关，单独新开一个页面来实现，通过SVG实现如下需求

- 画一条线
- 画一个矩形
- 画一个圆形
- 显示一些文字
- 画一朵花
- 画一颗小树

先来个柱状图

需求详细描述

没有过多复杂的要求，我们先把焦点聚焦在图形的绘制上。使用SVG，来实现这个柱状图，包括以下元素：

- 横轴
- 纵轴
- 数据项
- 固定只显示华东地区手机12个月的数据
- 图表样式不做限制，可以参考ECharts
- 暂时不在图表上做任何文字显示

实现思路

当你看完SVG，你会发现，大部分工作量都是在计算每个图形元素的位置及高宽

所以，找一个计算器很重要



新建一个js文件叫做：bar.js，把柱状图的代码放在这个文件里。

画每个柱子其实很简单，就是绘制一个矩形，但在绘制的过程中，我们需要考虑数据对应的柱子高度是多少，而这个比例我们在这个练习中，需要以所有数据的最大值来进行参考和计算

```
function 绘制一个柱状图(柱状图数据) {  
    定义好柱状图绘制区域的高度，宽度，轴的高度，宽度  
    定义好每一个柱子的宽度及柱子的间隔宽度  
    定义好柱子颜色，轴的颜色  
  
    拿到柱状图中的最大值Max  
    根据Max和你用来绘制柱状图图像区域的高度，进行一个数据和像素的折算比例  
  
    绘制横轴及纵轴  
    遍历数据 {  
        计算将要绘制柱子的高度和位置  
        绘制每一个柱子  
    }  
}
```

Canvas小练习

小练习编码

这个练习的代码和报表页面无关，单独新开一个页面来实现，通过Canvas实现如下需求

- 画一条线
- 画一个矩形
- 画一个圆形
- 显示一些文字
- 画一个时钟
- 画一朵云

再来个折线图

我们再学习使用HTML5中第二个重要的绘图方式，Canvas，来绘制折线图

- MDN Canvas (https://developer.mozilla.org/zh-CN/docs/Web/API/Canvas_API/Tutorial)
- W3school Canvas (http://www.w3school.com.cn/tags/html_ref_canvas.asp)

需求详细描述

和柱状图一样，包括以下元素：

- 横轴
- 纵轴
- 每个数据对应坐标中的数据点，可以用一个直径为5的实心圆
- 每个数据点之间连接的直线
- 固定只显示华东地区手机12个月的数据



实现思路

请准备好计算器。

新建一个js文件叫做：line.js，把折现图的代码放在这个文件里。

```
function 绘制一个折线图(折线图数据) {
    定义好折线图绘制区域的高度，宽度，轴的高度，宽度
    定义好每一个数据点的直径，颜色，线的颜色，宽度
    定义好没两个数据点之间的横向间隔距离

    拿到折线图中的最大值Max
    根据Max和你用来绘制折线图图像区域的高度，进行一个数据和像素的折算比例

    绘制横轴及纵轴
    遍历数据 {
        计算将要绘制数据点的坐标
        绘制数据点
        if 不是第一个点 {
            绘制这个数据点和上一个数据点的连线
        }
        记录下当前数据点的数据用于下一个点时绘制连线
    }
}
```

让图表数据可变

上面我们的数据都是用的固定的，接下来我们需要让图表的数据可变。

我们给表格增加一个鼠标滑过的事件响应，当鼠标滑过任何一行时，把这一行的数据在两个图表中进行呈现

把两个图表做成左右布局，各占一半宽度，表格在图表下方，表单在图表上方，几个内容保证在一个屏幕中能够完整呈现

设计思路

我们不妨将你的两个图表进行一下调整，让它们和数据解耦的组件，提供一些方法（如果已经有面向对象基础的同学可以进行类的封装）

```
折线图 = {

    图表数据: Array

    相关各种定义

    .....

    绘制图表: function() {

    }

    .....

    设置数据: function(数据) {
        根据新的数据从新 绘制图表()
    }

    .....

};
```

这样，在鼠标滑动过某一行表格时，获取到对应行的数据，然后调用图表的设置数据方法。顺便提供一下鼠标滑过的设计思路，当然我相信到这里大部分同学已经知道怎么做了

实现思路一：

```
// 在绘制表格的时候，给对应的td或者tr添加一个自定义属性，这一格数据属于哪个商品哪个区域
```

```
表格添加mouseover事件 = function () {  
    获取对应tr或者td的商品及区域的自定义属性  
    根据上面两个属性在数据中获取对应的12个月的数据  
    调用图表的设置数据方式  
}
```

实现思路二：

```
表格添加mouseover事件 = function () {  
    拿到响应事件对应的tr，然后依次遍历其中的td，获取其中的数据  
    调用图表的设置数据方式  
}
```

不妨两种都试试，看看自己觉得哪种好

绘制多条折线图

需求描述

单独看一行数据，只能看一个指标自己的趋势，我们往往还需要在不同数据之间进行比较，所以我们需要在折线图中绘制多条数据。

- 根据表单的选择，在折线图中显示相应的折现，和对应表格中的数据对应
- 每一条线选择不同的颜色
- 另外保留上面的鼠标hover某一行时显示某一行数据的图表，但鼠标移开表格后，再恢复到显示表单对应的所有数据

设计思路

对于折线图而言，其实画一条线和画多条线没有本质区别，需要我们额外考虑的点是：

- 纵坐标的最大值取决于所有数据的最大值，而不仅仅是一个数据
- 颜色的管理，管理一个颜色序列，分配给每一条线

我们把整个流程再做一下拆解和梳理，对于画每一条线而言，需要的包括：

- 数据
- 颜色
- 两个点之间的间隔
- 数据点直径
-

上面这些有的数据对于每一条线是不一样的，有些是一样的，不一样的我们通过参数传递，一样的可以直接读取对象的属性
整体流程变为：

- 计算整体的数据与像素的比例
- 遍历数据，渲染每一条折线

绘制多个柱状图（可选）

需求描述



这个是个附加需求，有余力的同学可以尝试：

柱状图稍微复杂一些，我们先看个例子 (<http://echarts.baidu.com/examples/editor.html?c=bar-label-rotation>)

- 以每个月为第一个维度进行柱状图的聚合
- 然后再按照表格同样的分类方式对数据进行如上方例子类似的聚合
- 每行数据有一个颜色，每一类数据用同系颜色

提交

把你的代码放在Github后进行提交

总结

依然把今天的学习用时，收获，问题进行记录

下一个任务预告

明天我们继续该任务的下一个环节

课程优秀学习笔记

暂无优秀学习笔记~

IFE微信公众号：baidu_ife IFE微博：Baidu前端技术学院 (<http://weibo.com/u/5568860641?topnav=1&wvr=6&topsug=1>)

友情链接： 百度EFE (<http://efe.baidu.com>) 百度校园 (<http://campus.baidu.com/>) 百度校园招聘
(<http://talent.baidu.com/external/baidu/campus.html>) 百度技术学院 (<http://bit.baidu.com/?fr=ife/>)

资源： IFE 2015 (<https://github.com/baidu-ife>) IFE 2016 (<http://ife.baidu.com/2016/static/index.html>) IFE 2017
(<http://ife.baidu.com/2017>)

©2017 Baidu 使用百度前必读 (<http://www.baidu.com/duty/>) 意见反馈 (<http://jianyi.baidu.com>) 京ICP证030173号

