# d3选择元素

var body = d3.select("body"); //选择文档中的body元素  
var p1 = d3.select("p"); //选择body中的第一个p元素  
var p = d3.selectAll("p"); //选择body中的所有p元素  
var svg = body.select("svg"); //选择body中的svg元素  
var rects = svg.selectAll("rect"); //选择svg中所有的svg元素  
d3.select('.brand') // 选择类为brand的元素  
d3.select('#brand') //选择id为brand的元素  
d3.select('.brand sum') // 选择类为brand的子元素sum

# 建立映射

var y = d3.scale.linear().domain([15,90]).range([250,0]);  
// 注意这里返回的是一个函数来的,链式的语法  
y(90); //y 就是一个函数了.  
  
var x = d3.scale.log().domain([100,10000]).range([0,1000])  
x(100) // 表示一种Log的函数.  
svg.attr('width',600).attr('height',600); // 一定要先设置这个svg的宽度和高度才能显示出来的。  
svg.append('circle').attr('r', r(1380000000)).  
attr('fill','red').attr('cx', x(13330)).attr('cy', y(77));  
//就是链式语法

# 数据绑定

<!DOCTYPE html>  
<html>  
 <head>  
 <script type="text/javascript" src="d3.v3.min.js"></script>  
 </head>  
 <body>  
 </body>  
</html>

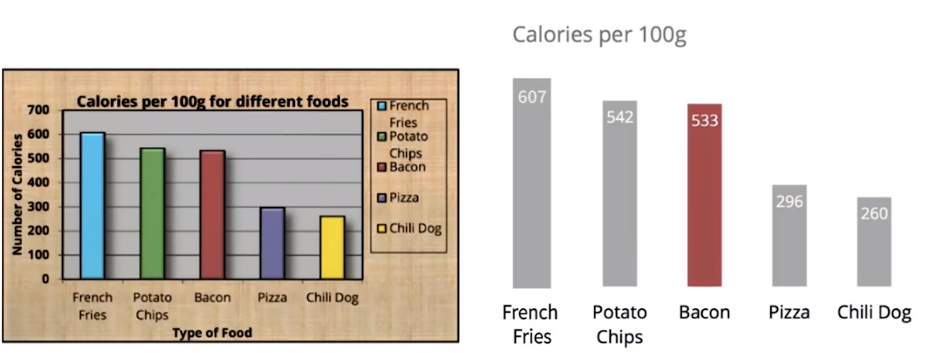
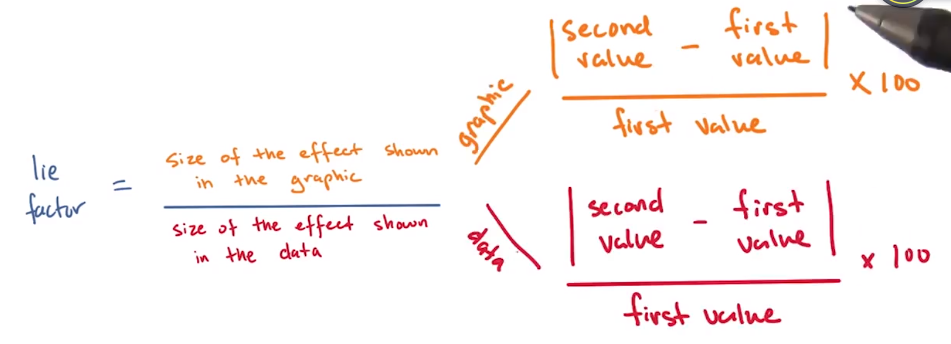
比如这个原始的网页中

var theData = [1,2,3] //一个数组  
var p = d3.select("body").selectAll("p")  
 .data(theData)  
 .enter()  
 .append("p")  
 .text("hello");

在上面那段Javascript代码中，有一个方法是：.selectAll("p"). D3.js中的selectAll方法使用的是CSS3中的选择器来获取DOM元素。不同于select方法（该方法获取的是第一个目标元素），而selectAll方法选择的是所有匹配的元素。 但是！！初识的HTML网页中并不包含，也就是说，该方法返回的是一个空的选择结果（selection）。 这个data操作符是把一个数据数组（可能是数值数组、对象数组或者别的什么数组）与当前的选择结果联系（join）起来。 在本例中，并没有提供key值，因此theData数组中的每一个元素都被分配到当前选择结果中的一个元素上。数组中第一个元素，也就是数字1，被分配到第一个元素，第二个元素2被分配到相应的第二个，以此类推。 但是！！正如上面说过的，初识网页中根本没有一个，那么到底怎么回事？ 与D3.js中的其他方法不同的是，data操作符返回的是三个Virtual selection。 这三个virtual selection是enter,update以及exit。 enter选集：对所有缺失的元素以占位符placeholder替代。 update选集：包含现有的元素，并绑定到数据 剩下的元素最终都会出现在exit选集中，并被移除。 因此，虚拟enter选集中包含的是元素的占位符。 我们将会的后面章节继续讨论虚拟选集enter,update,exit，而现在，我们就把焦点放在enter虚拟选集上。

var x = d3.scale.linear()  
 .domain([0, d3.max(data)])  
 .range([0, 420]);//创建刻度  
  
d3.select(".chart")  
 .selectAll("div")  
 .data(data)  
 .enter().append("div")  
 .style("width", function(d) { return x(d) + "px"; }) //返回了一个函数  
 .text(function(d) { return d; });

# 可视化设计原则

* 适当使用颜色
* 图表垃圾 
* 数据墨水比率:描述数据本身的内容与描述其他内容的比值(需要提高数据墨水比率)  由左边的图形向右边图形的转变. 删除:背景,各自,颜色,图表,标题.
* 失真系数(lie factor)  这个计算得公式有点难理解. 计算步骤: 