1.d3就是Data-Driven Documents 是一个数据驱动的文档，也就是用js驱动dom实现可视化

2.为什么用d3:

因为比起来现在的可视化工具（echarts highcharts）d3可以实现更复杂的效果

3.基本api：

1.选择元素 select() selectAll()

可以选择标签 class id等

使用：d3.select('span')等之类的 //获取一个元素

d3.selectAll('span')等之类的 //获取一组元素

2.绑定数据 data() datum()

将数据绑定到元素上，就是元素可以获取到数据

使用：d3.selectAll('span').data(arr) //data()里面放置一组元素

d3.selectAll('span').datum(a) //datum()里面放置一个元素

注意：当使用data的时候，对应的获取元素用的是selectAll，那么每一个数据对应一个元素，有顺序的。

当使用datum的时候，对应的也是使用selectAll，那么每一个元素对应的就是一个数据

3.插入元素 append() insert() remove()

这个和jquery里面是一样的

使用：append()后面

insert()前面

remove()删除

4.添加图形

画布 append('svg')

矩形 append('rect') 注意：矩形之后是需要设置宽高的

5.添加样式

attr() 在获取添加了元素之后，我们就可以用这个添加样式，css的样式都可以用key:value形式添加

6.比列尺（比较重要的概念）

将某一区域的值映射到另一区域，其大小关系不变

根据d3.js中api的说法就是domain是设置定义域，range是输出范围

(1)比列尺有几种：线性比列尺

能将一个连续的区间，映射到另一个区间，比如arr=[1,5,2,6,4,10] 我们定义最小的值等于0 最大的值等于100：

var min=d3.min(arr);

var max=d3.max(arr);

var linear=d3.scale.linear().domain([min,max]).range([0,100]);

linear(1)//0 现在arr中最小的1对应的就是0 同样的最大的10对应的就是100

在这里面min max找到数组中的最小和最大值，d3.scale.linear返回一个线性比列尺，domain定义定义域，range定义值域

注意：d3.scale.linear定义返回的是一个函数

(2)序数比列尺：

比如arr=[1,2,3,4] brr=['香蕉'，‘苹果’，‘橘子’，‘柠檬’] 想让arr和brr一一对应，这就不能用线性了

var ordinal=d3.scale.ordinal().domain(arr).range(brr) 这样就一一对应了

7.坐标轴

理解一下svg里面有几种图形：圆形 矩形 折线 椭圆 线段 多边形 路径

var data=[2,3,4,5,67]

var linear=d3.scale.linear().domain([0,d3.max(data)]).range([0,300])

var axis=d3.svg.axis().scale(linear).orient('bottom').ticks(7)

//d3.svg.axis 获取d3中的坐标轴组件 scale执行比列尺 orient制定比例尺的朝向 ticks执行刻度的数量

上面设置好坐标轴之后，现在需要将坐标轴添加到我们的元素中

svg.append('g').call(axis)//call将当前坐标集作为参数传递给此函数

当然也可以设置坐标的样式，直接用css样式 比如.axis path{}

注意：

1.如果想要将图形的背景色改变就需要用attr('fill','red')