d3.goup VS. d3.groups

<https://github.com/d3/d3-array/blob/v2.11.0/README.md#group>

d3 6.0 migration guide

<https://observablehq.com/@d3/d3v6-migration-guide#group>

replace d3.nest

|  |
| --- |
| var groupTweets = d3.group(incomingData, d=>d.user)  console.log("check groupTweets",groupTweets)  // pack  var packChart = d3.pack().padding(10)  packChart.size([500, 500])  // console.log("check packChart",packChart)  var packableTweets = {id: "All Tweets", values: groupTweets}  console.log("check packableTweets",packableTweets)  var root = d3.hierarchy(packableTweets, d=>{  // Trick of migrating v4 to v6  if (!d.hasOwnProperty("values")) {  d['values'] = d[1];  }  return d.values;  })  .sum(()=>1) |

d3 6.0 treemap example

<https://observablehq.com/@d3/cascaded-treemap>

pack circle inspirations

<http://nbremer.github.io/occupationscanvas/>

<https://graphics.reuters.com/HEALTH-CORONAVIRUS/BATS/qzjpqglbxpx/index.html>

Sankey diagram

Add links 时定义stroke颜色才能正确显示link的等宽度；

方法一：.attr(“stroke”, gray)才正常显示

方法二：CSS .className{stroke: gray}

数据内的link-value的数值比例会同样对应到stroke-width

Append(“text”), The dy attribute indicates a shift along the y-axis on the position of an element or its content.

根据位置判断条件设置位置

|  |
| --- |
| // add nodes title  node.append("text")  .attr("x", d=>d.x0 -6)  .attr("y", d=>(d.y1+d.y0)/2)  .attr("dy", "0.35em")  .attr("text-anchor", "end")  .text(d=>d.name)  .filter(d=>d.x0 < width/2)  .attr("x", d=>d.x0 -6)  .attr("text-anchor", "start") |

d3.format(",.0f") 逗号=每三个0用逗号分隔一下；.0=小数点后保留0位

20210308:

var parseDate = d3.timeParse("%b-%y")

输出的形式是标准时间= Fri Jul 01 2011 00:00:00 GMT+0800 (中国标准时间)

输出不是设定的时间格式，这个格式是迎合原数据日期格式，为了统一读取成标准时间

var reparseDate = d3.timeFormat("%y/%B")

输出按照自定义格式的数据，在axis中调用

格式自定义缩写含义

|  |
| --- |
| * %a - abbreviated weekday name. * %A - full weekday name. * %b - abbreviated month name. * %B - full month name. * %c - date and time, as “%a %b %e %H:%M:%S %Y”. * %d - zero-padded day of the month as a decimal number [01,31]. * %e - space-padded day of the month as a decimal number [ 1,31]. * %H - hour (24-hour clock) as a decimal number [00,23]. * %I - hour (12-hour clock) as a decimal number [01,12]. * %j - day of the year as a decimal number [001,366]. * %m - month as a decimal number [01,12]. * %M - minute as a decimal number [00,59]. * %p - either AM or PM. * %S - second as a decimal number [00,61]. * %U - week number of the year (Sunday as the first day of the week) as a decimal number [00,53]. * %w - weekday as a decimal number [0(Sunday),6]. * %W - week number of the year (Monday as the first day of the week) as a decimal number [00,53]. * %x - date, as “%m/%d/%y”. * %X - time, as “%H:%M:%S”. * %y - year without century as a decimal number [00,99]. * %Y - year with century as a decimal number. * %Z - time zone offset, such as “-0700”. * There is also a a literal “%” character that can be presented by using double % signs. |

常用自定义格式：

%Y=四位数字年份，eg：2021

%b=月份缩写， eg：Jan

%m=两位数字月份，eg：01

20210309

X/Y axis字体字号可以通过.style("font", "14px times")设置，更简洁是通过class在<style>中设置.axis{font: 14px sans-serif; }

坐标刻度间隔自定义（以往虽然指定数字但内部一定程度自动分配不完全实现自定义），.ticks(d3.timeDay.every(4))，这里是每隔四天，其他设置还有

|  |
| --- |
| • d3.timeMillisecond : Milliseconds • d3.timeSecond : Seconds • d3.timeMinute : Minutes • d3.timeHour : Hours  • d3.timeDay : Days • d3.timeWeek : This is an alias for d3.timeSunday for a week • d3.timeSunday : A week starting on Sunday • d3.timeMonday : A week starting on Monday • d3.timeTuesday : A week starting on Tuesday • d3.timeWednesday : A week starting on Wednesday • d3.timeThursday : A week starting on Thursday • d3.timeFriday : A week starting on Friday • d3.timeSaturday : A week starting on Saturday • d3.timeMonth : Months starting on the 1st of the month • d3.timeYear : Years Starting on the 1st day of the year |

坐标刻度文字旋转角度

|  |
| --- |
| .selectAll("text")  .style("text-anchor", "end")  .attr("transform", "rotate(-25)")  .attr("dx", ".5em")  .attr("dy", ".7em") |

Y坐标轴label旋转-90（先放入g，移动g调整旋转的原点，再旋转比较好理解）

更多translate与rotate执行顺序见index\_how they work\_translateOrratate.html

|  |
| --- |
| // y label  svg.append("g")  .attr("transform", "translate(" + 50 + "," + 250 +")")  .append("text")  // .attr("x", 450)  // .attr("y", 250)  .attr("dy", ".3em")//相对于左边定义的一个框非右侧y轴  .attr("transform", "rotate(-90)")  .style("text-anchor","middle")  .text("Value") |

连接线curve的方式

|  |
| --- |
| <https://github.com/d3/d3-shape/blob/master/README.md#curves>  常用  .curve(d3.curveBasis)  .curve(d3.curveStep) |

0310:

自定义坐标开始以及显示什么刻度使用.tickValues([50,150,250])；

于此相似但并是不一个功能的

console.log("check ticks", d3.ticks(50,250,5))=> [50, 100, 150, 200, 250]

console.log("check tickIncrement", d3.ticks(50,250,5)) => 50

console.log("check tickStep", d3.tickStep(50,250,5)) => 50

<https://ghenshaw-work.medium.com/customizing-axes-in-d3-js-99d58863738b>

0312

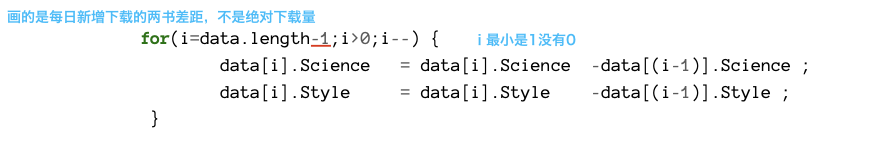
Transition的运动方式动画演示，函数原理

<https://bl.ocks.org/d3noob/fa4cfde5450852cfef0603857f080585>

<https://observablehq.com/@d3/easing>

|  |
| --- |
| • linear  • quad  • cubic  • poly  • sin  • exp • circle • bounce • back • elastic |

0313: for循环区间



，5-1，没有0；执行完foe-输出结果-foo减一和下一个foe结合在一块；0通不过



Axis对应path，line

|  |
| --- |
| .x.Axis line{ /\*刻度线\*/  display: none;  }  .x.Axis path{ /\*轴线\*/  display: none;  } |