二维图表玩转组件化

姚向阳

小米高级软件工程师

自我介绍

姚向阳

小米： IDE/工具链方向

奇虎 360： Qmap， QCharts

明源云：明源云客、工作流

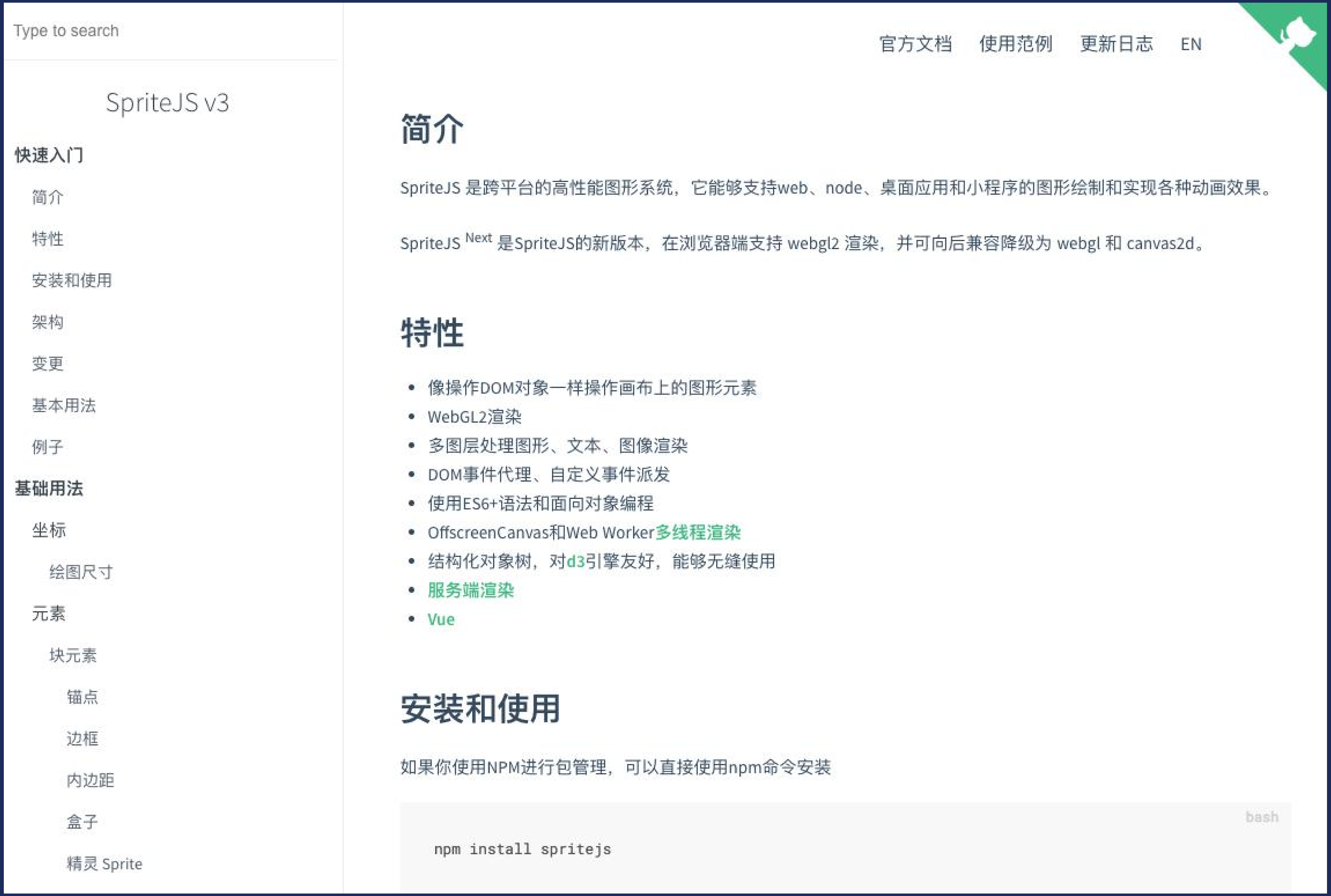
百度：百度移动首页、阿拉丁

大纲

- 1. 背景
- 2. 解决样式复杂
- 3. 实现组件化
- 4. 适配Vue、React问题
- 5. 不足、后续未来

背景

“ 可视化积累，满足部分个性化需求



简介

SpriteJS 是跨平台的高性能图形系统，它能够支持web、node、桌面应用和小程序的图形绘制和实现各种动画效果。

SpriteJS Next 是SpriteJS的新版本，在浏览器端支持 webgl2 渲染，并可向后兼容降级为 webgl 和 canvas2d。

特性

- 像操作DOM对象一样操作画布上的图形元素
- WebGL2渲染
- 多图层处理图形、文本、图像渲染
- DOM事件代理、自定义事件派发
- 使用ES6+语法和面向对象编程
- OffscreenCanvas和Web Worker多线程渲染
- 结构化对象树，对d3引擎友好，能够无缝使用
- 服务端渲染
- Vue

安装和使用

如果你使用NPM进行包管理，可以直接使用npm命令安装

```bash
npm install spritejs
```

Echarts

> “优秀，~~文档复杂，二次开发困难~~

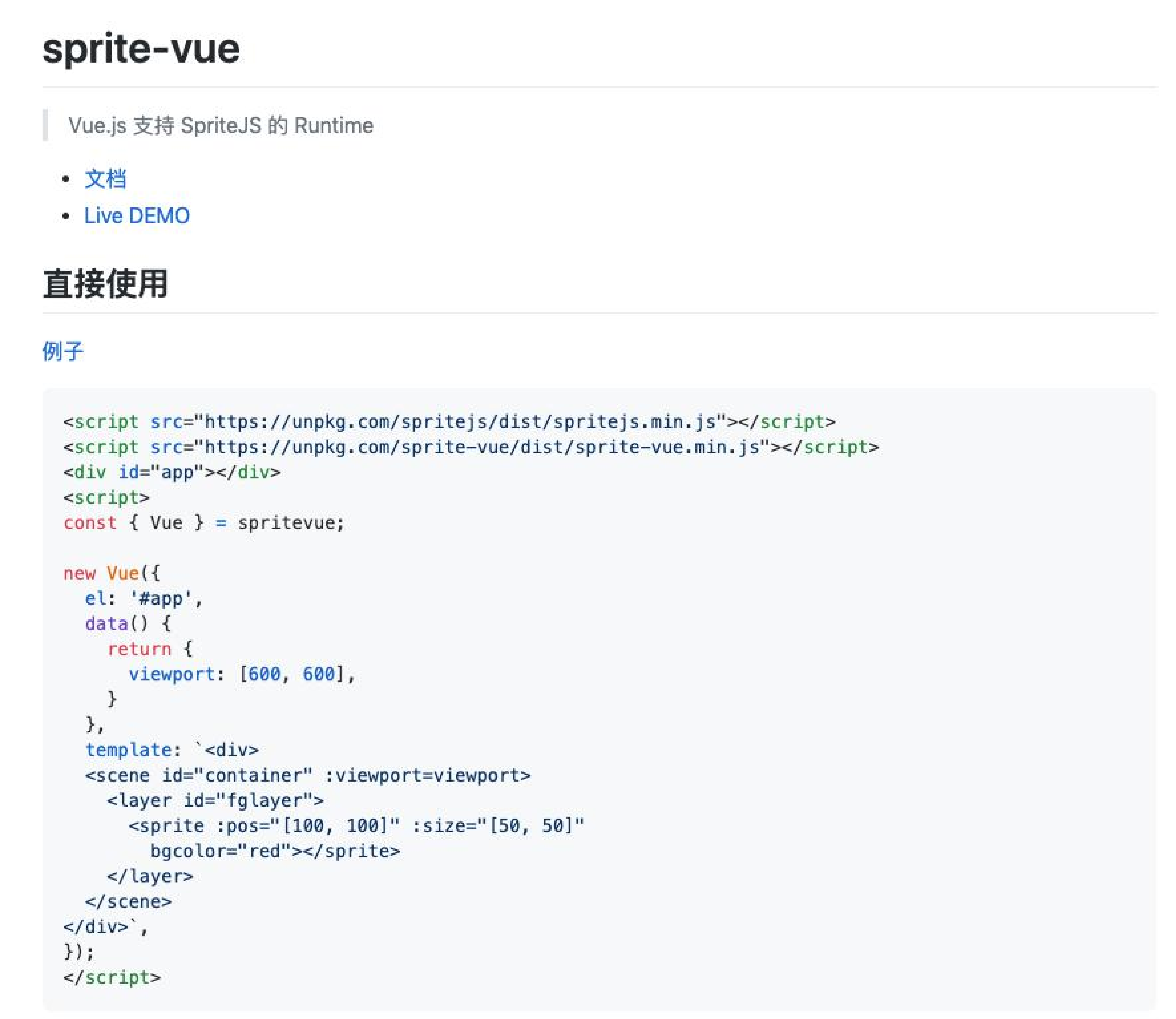
耗时需求

“ 修改图表样式

如何解决

Api配置多，文档多、样式设置复杂

图形业务关系、render，mvvm，spritejs



sprite-vue

> Vue.js 支持 SpriteJS 的 Runtime

- 文档
- Live DEMO

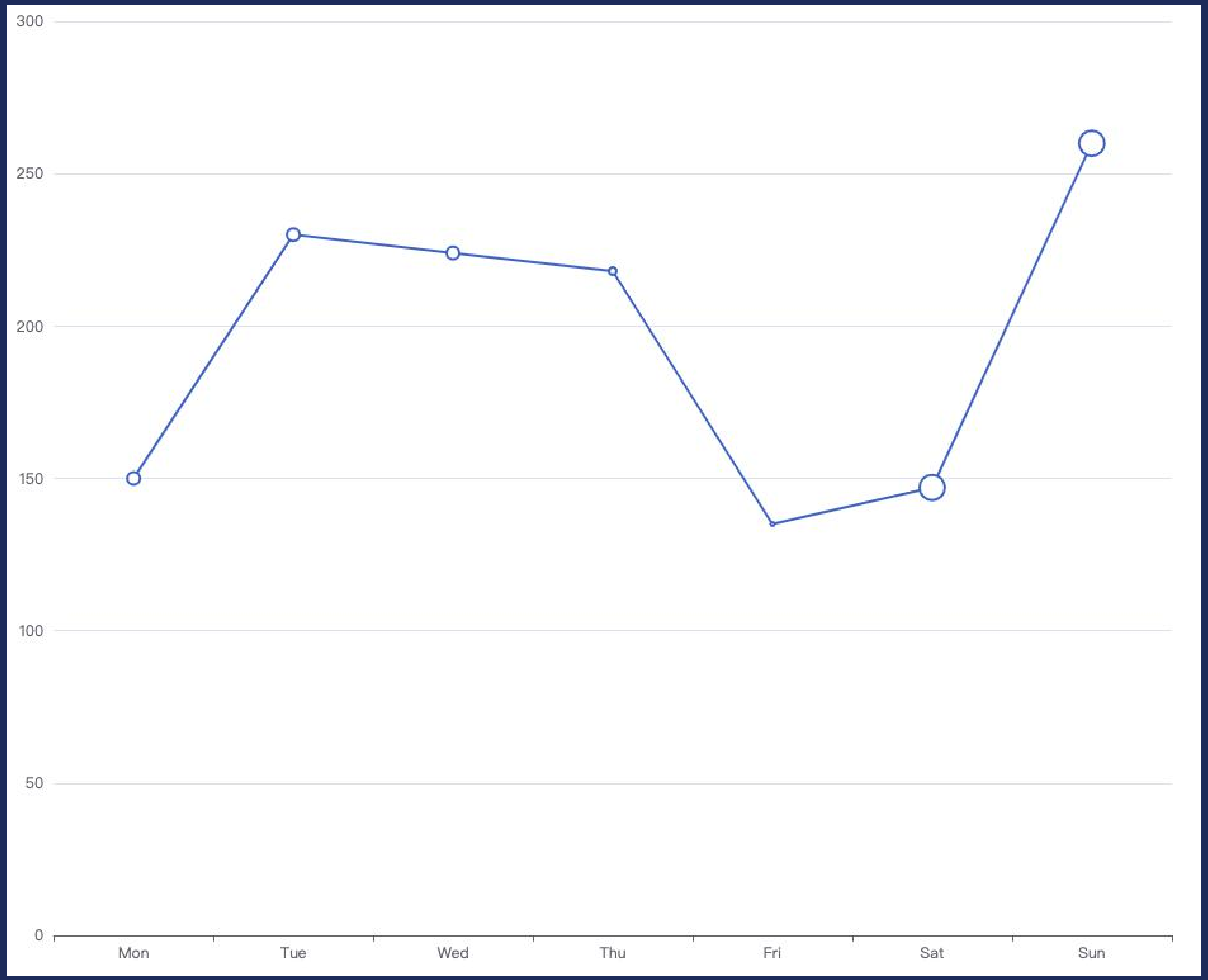
直接使用

例子

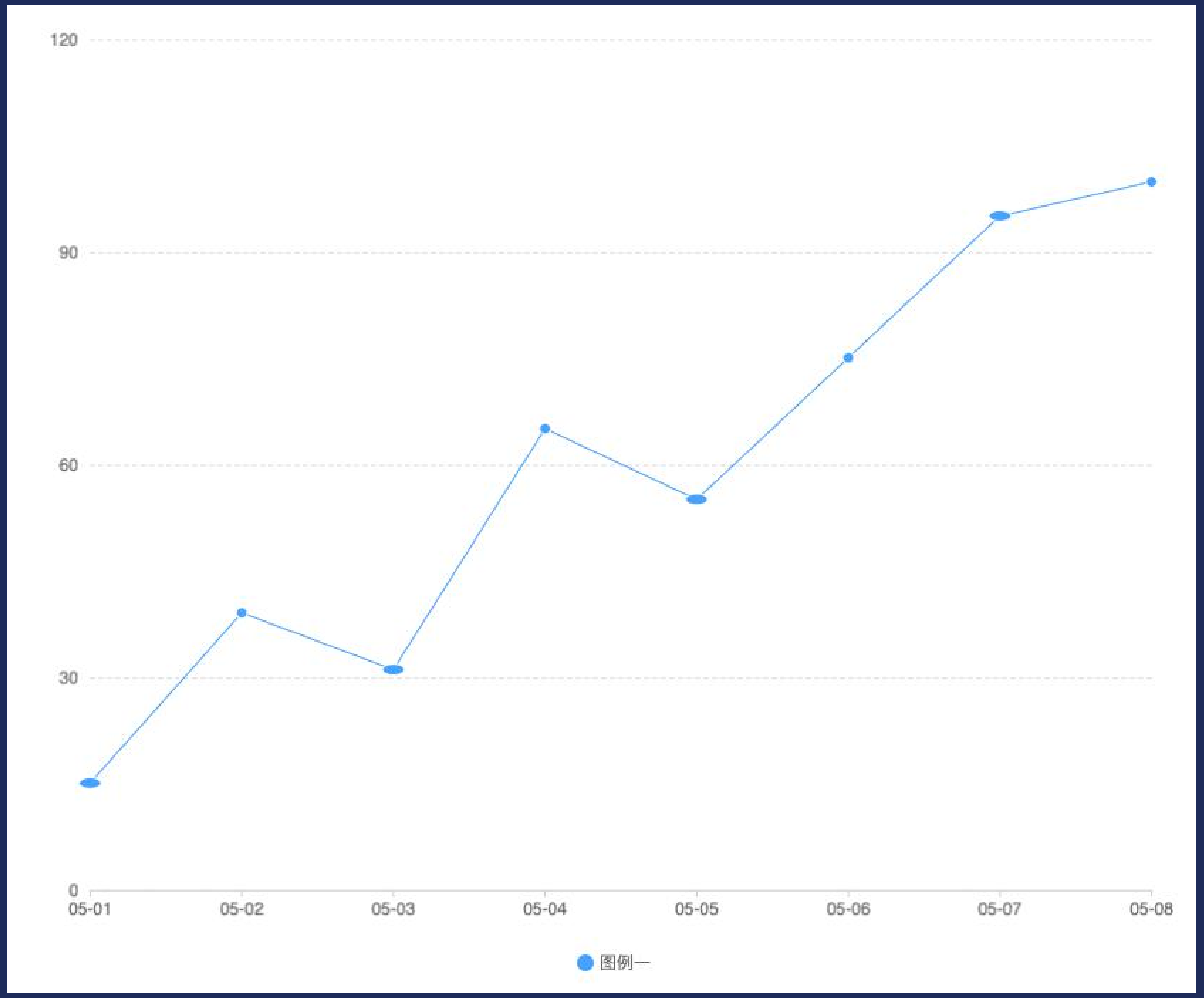
```
<script src="https://unpkg.com/spritejs/dist/spritejs.min.js"></script>
<script src="https://unpkg.com/sprite-vue/dist/sprite-vue.min.js"></script>
<div id="app"></div>
<script>
const { Vue } = spritevue;

new Vue({
  el: '#app',
  data() {
    return {
      viewport: [600, 600],
    }
  },
  template: `<div>
  <scene id="container" :viewport=viewport>
    <layer id="fglayer">
      <sprite :pos="[100, 100]" :size="[50, 50]"
        bgcolor="red"></sprite>
    </layer>
  </scene>
</div>`,
});
</script>
```

解决样式思路

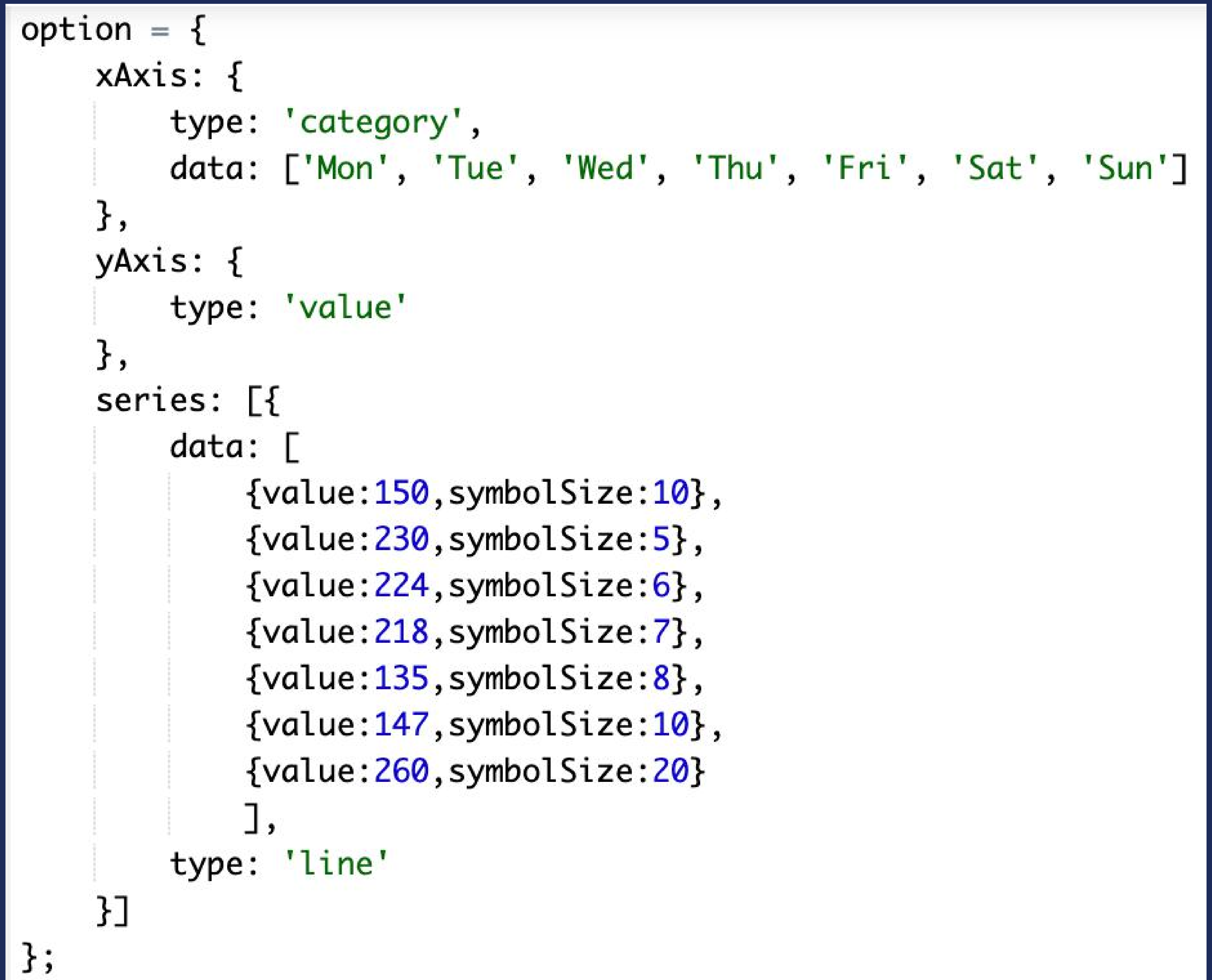


300
250
200
150
100
50
0
Mon
Tue
Wed
Thu
Fri
Sat
Sun

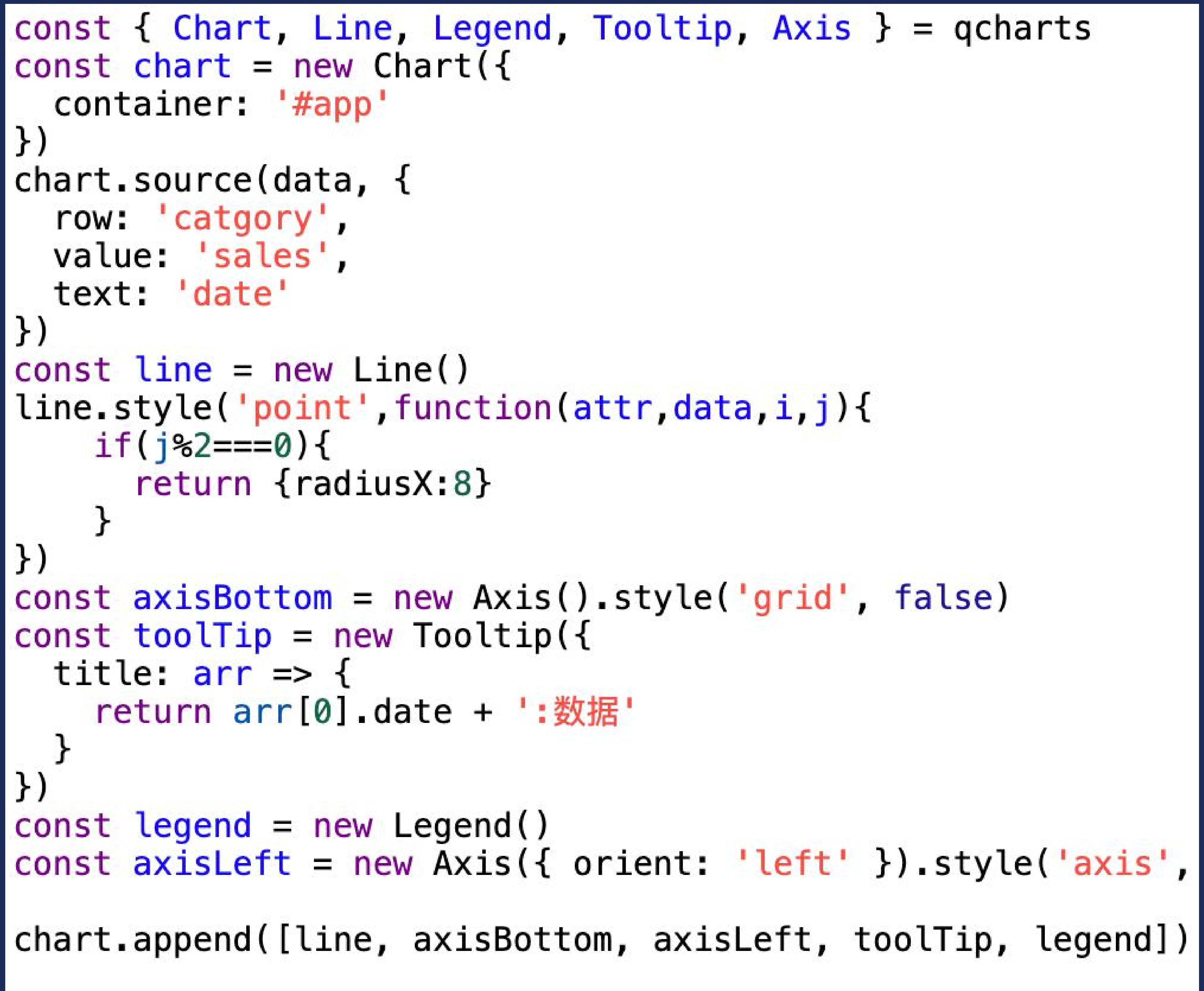


120
90
60
30
0
05-01
05-02
05-03
05-04
05-05
05-06
05-07
05-08
● 图例一

基本用法



```
option = {
    xAxis: {
        type: 'category',
        data: ['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun']
    },
    yAxis: {
        type: 'value'
    },
    series: [{
        data: [
            {value:150,symbolSize:10},
            {value:230,symbolSize:5},
            {value:224,symbolSize:6},
            {value:218,symbolSize:7},
            {value:135,symbolSize:8},
            {value:147,symbolSize:10},
            {value:260,symbolSize:20}
            ],
        type: 'line'
    }]
};
```



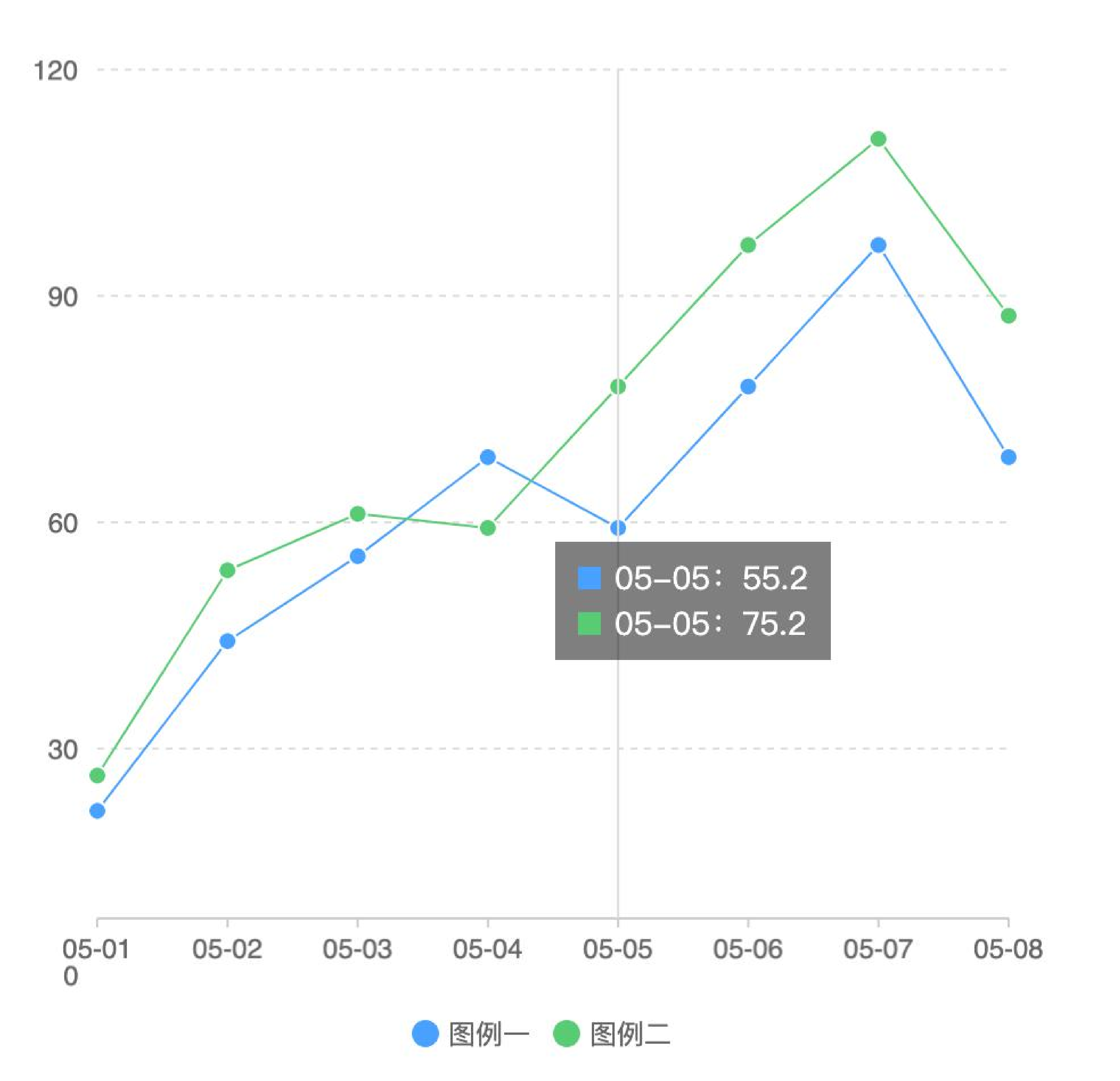
```
const { Chart, Line, Legend, Tooltip, Axis } = qcharts
const chart = new Chart({
  container: '#app'
})
chart.source(data, {
  row: 'catgory',
  value: 'sales',
  text: 'date'
})
const line = new Line()
line.style('point',function(attr,data,i,j){
    if(j%2===0){
      return {radiusX:8}
    }
})
const axisBottom = new Axis().style('grid', false)
const toolTip = new Tooltip({
  title: arr => {
    return arr[0].date + ':数据'
  }
})
const legend = new Legend()
const axisLeft = new Axis({ orient: 'left' }).style('axis',

chart.append([line, axisBottom, axisLeft, toolTip, legend])
```

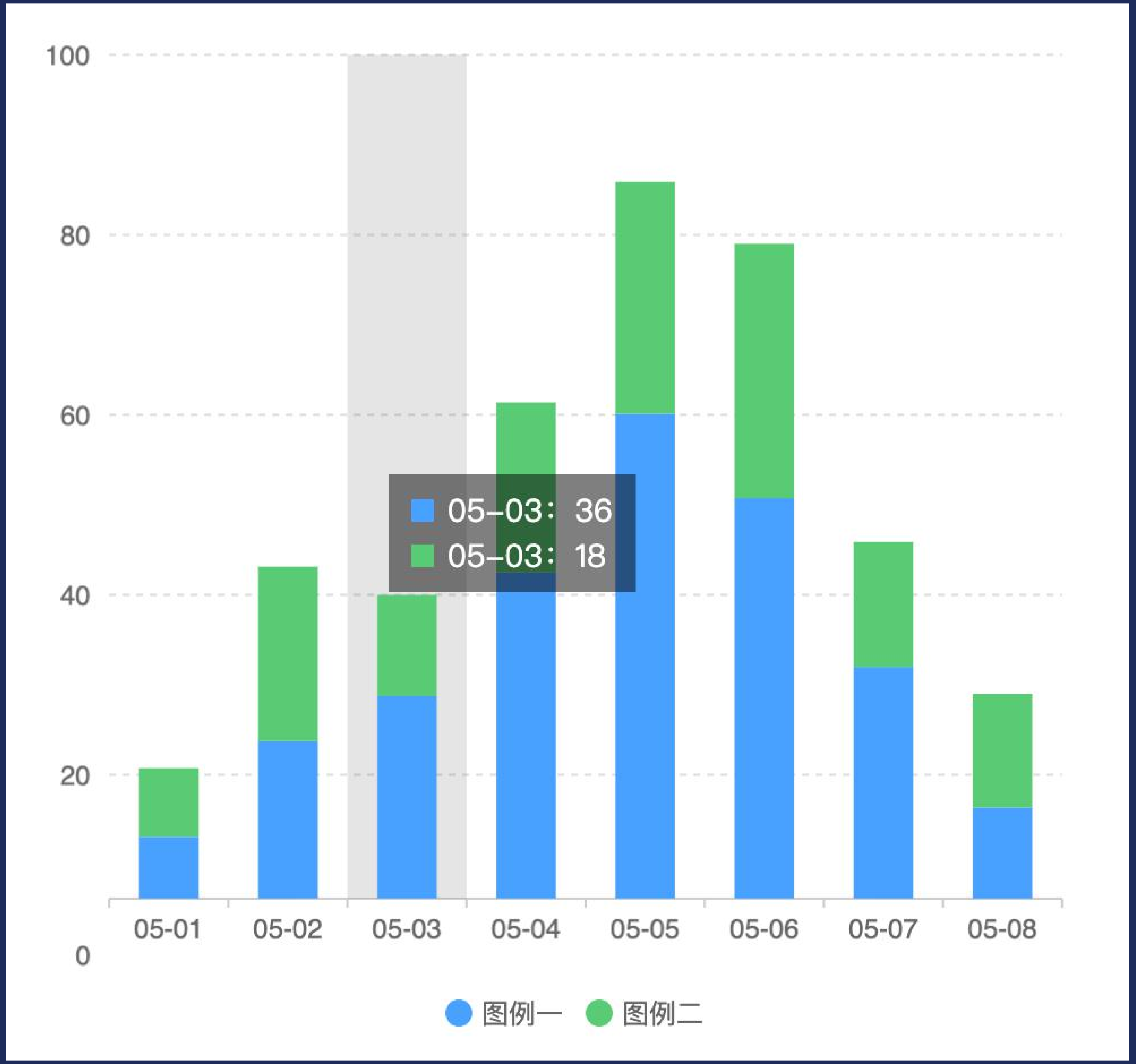
属性归纳

“ ~~点、~~面、线、文本

属性归纳



120
90
60
30
0
05-01
05-02
05-03
05-04
05-05
05-06
05-07
05-08
05-05：55.2
05-05：75.2
● 图例一
● 图例二



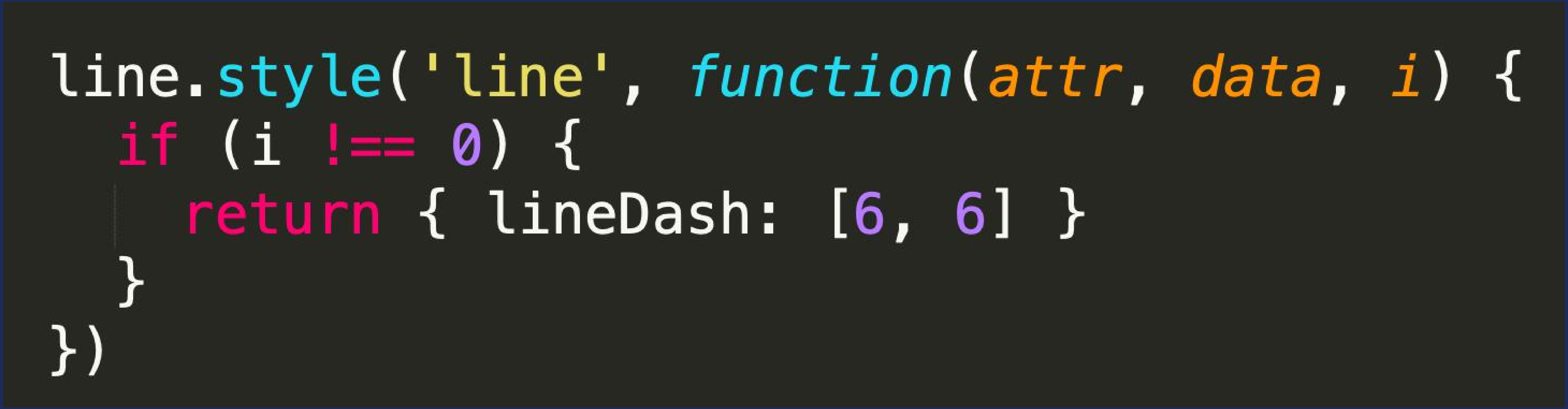
100
80
60
40
20
0
05-01
05-02
05-03
05-04
05-05
05-06
05-07
05-08
05-03：36
05-03：18
● 图例一
● 图例二

面、线、文本

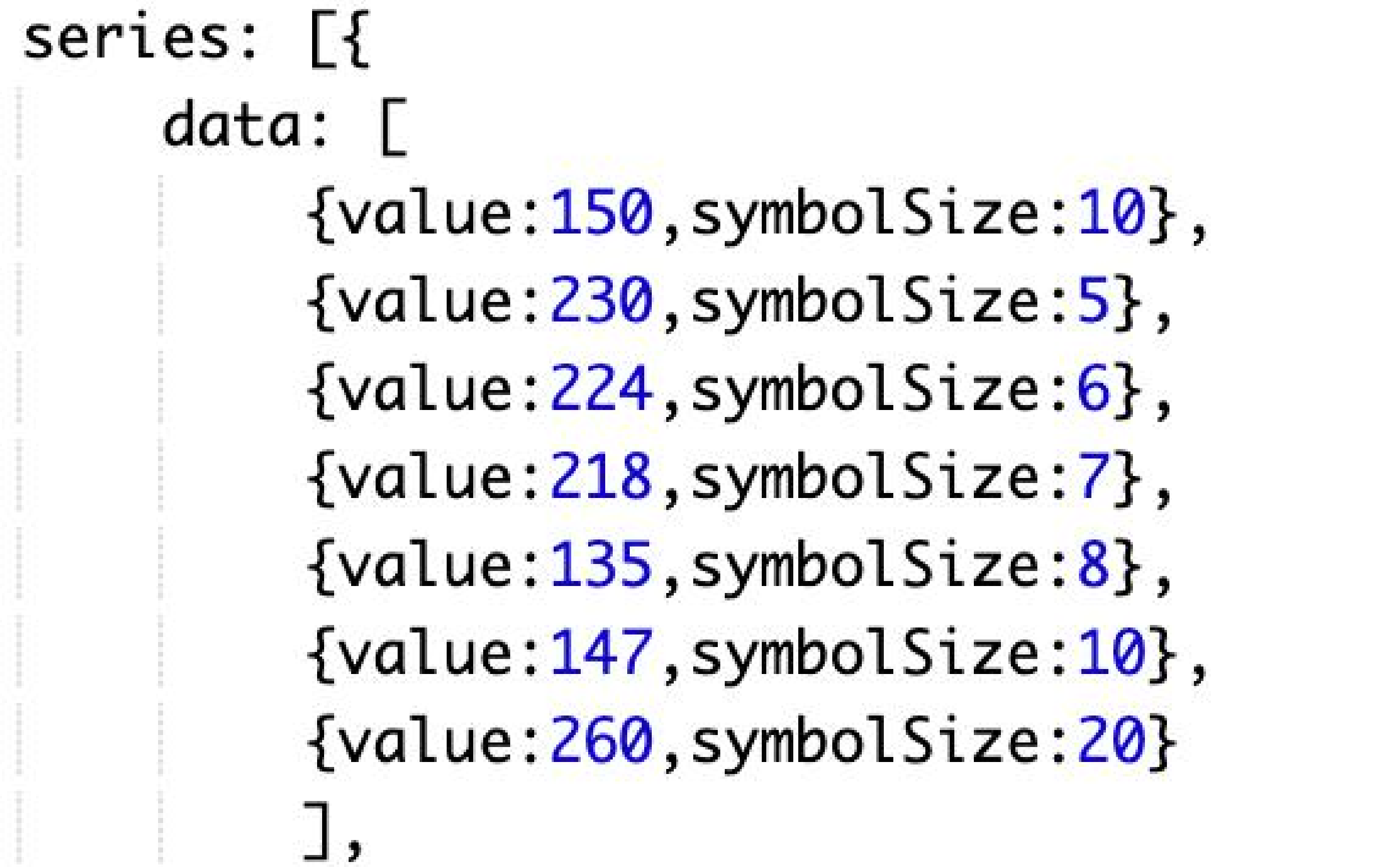
线	面	文本
qrpmi cAmjmp	qrpmi cAmjmp	` eamjmp
jgl cB_qf	dgjjAmjmp	dgjjAmjmp
jgl cA_n	jgl cU gbrf	dml rQgxc
jgl cHmgl	_l af mp	dml rD_k gjw
qrpmi cAmjmp	mn_agw	dml rU cgef r
jgl cU gbrf	rp_l qj_rc	_l af mp
mn_agw	bgqnj_w	n_bbgl e
rp_l qj_rc		mn_agw
bgqnj_w		rp_l qj_rc
		bgqnj_w

https://spritejs.org/#/

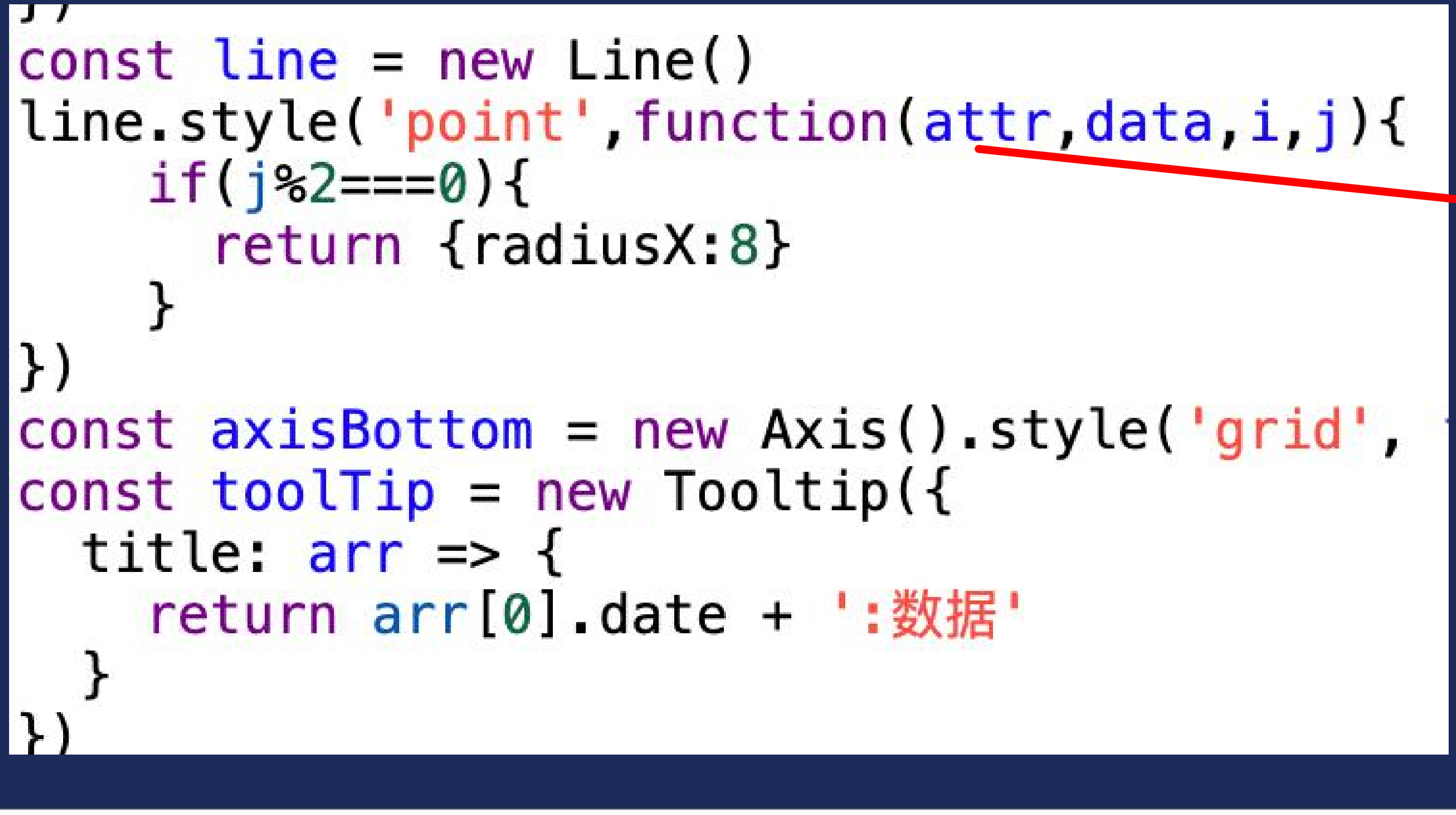
回调函数



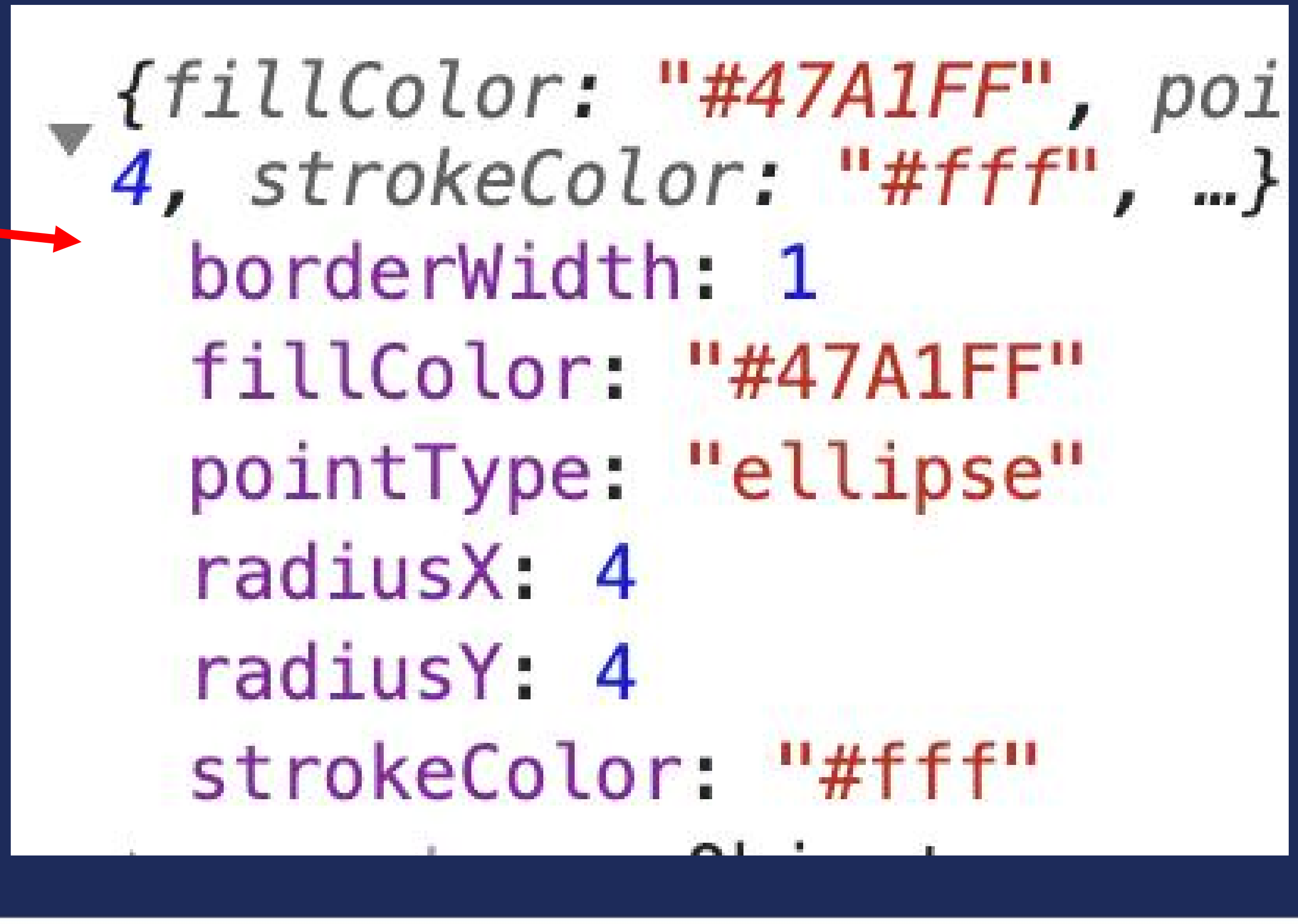
```
line.style('line', function(attr, data, i) {
  if (i !== 0) {
    return { lineDash: [6, 6] }
  }
})
```



```
series: [{
    data: [
        {value:150,symbolSize:10},
        {value:230,symbolSize:5},
        {value:224,symbolSize:6},
        {value:218,symbolSize:7},
        {value:135,symbolSize:8},
        {value:147,symbolSize:10},
        {value:260,symbolSize:20}
        ],
```

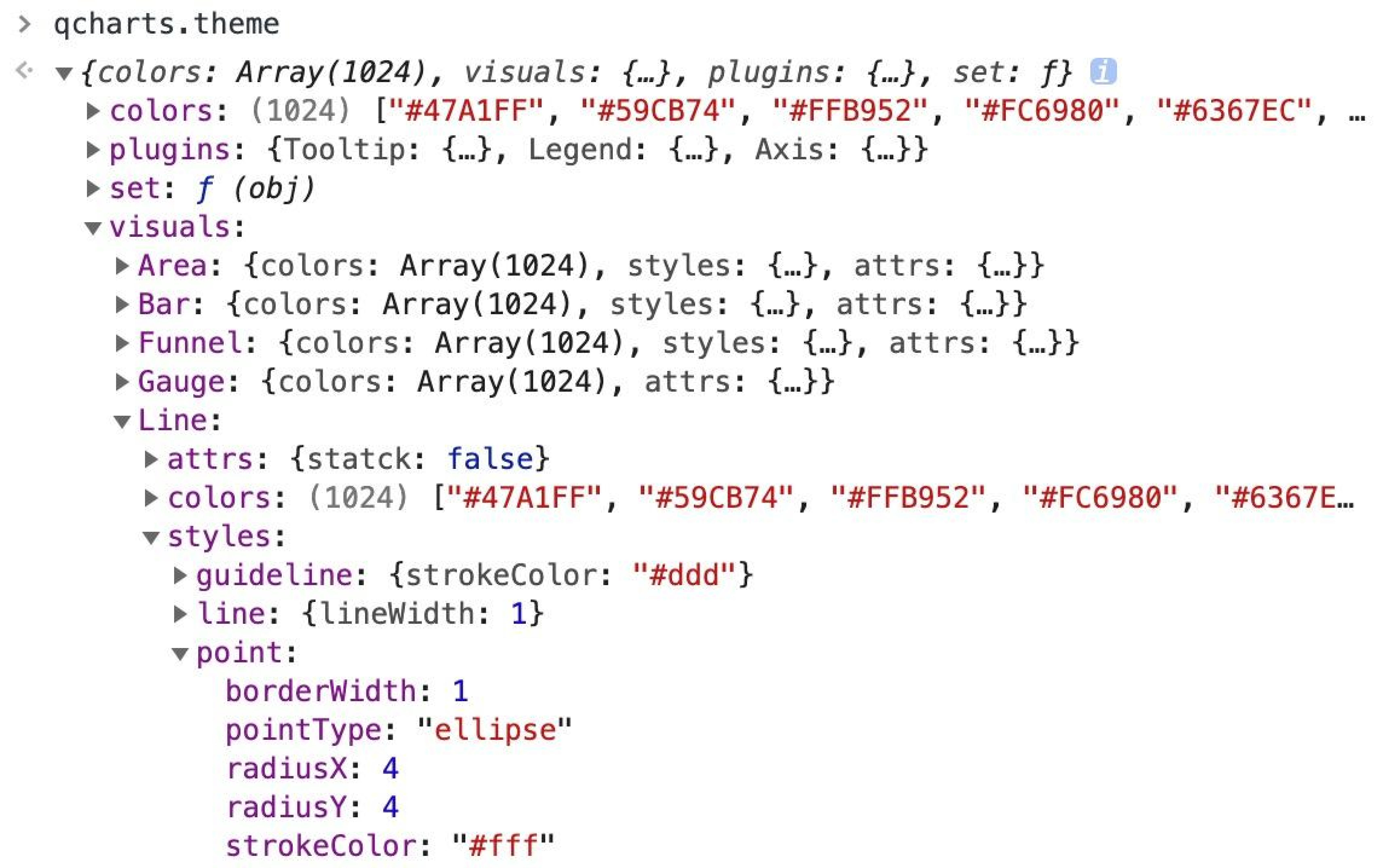


```
const line = new Line()
line.style('point',function(attr,data,i,j){
    if(j%2===0){
      return {radiusX:8}
    }
})
const axisBottom = new Axis().style('grid',
const toolTip = new Tooltip({
  title: arr => {
    return arr[0].date + ':数据'
  }
})
```



```
▼ {fillColor: "#47A1FF", poi
  4, strokeColor: "#fff", …}
    borderWidth: 1
    fillColor: "#47A1FF"
    pointType: "ellipse"
    radiusX: 4
    radiusY: 4
    strokeColor: "#fff"
```

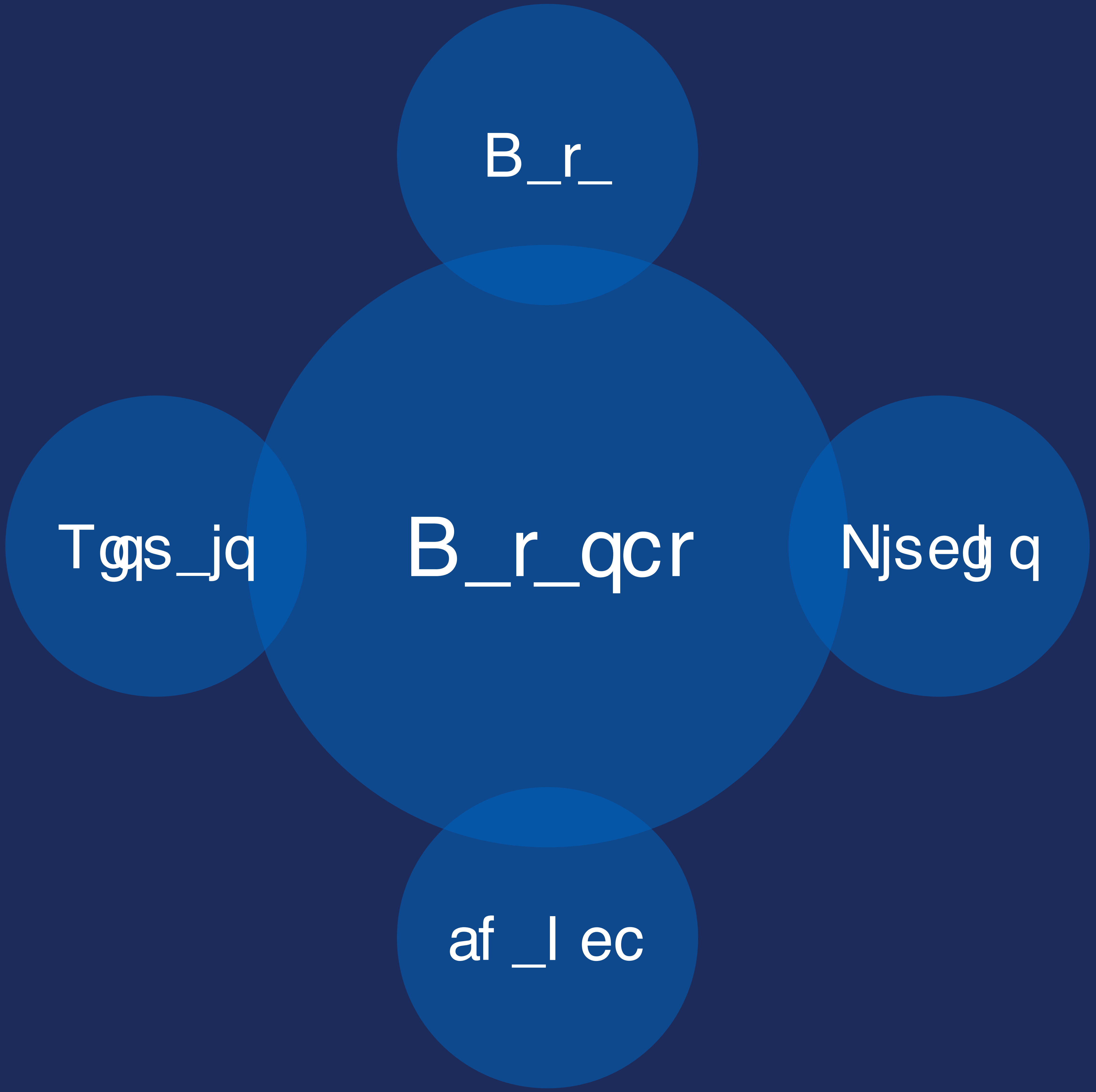
全局样式



```
> qcharts.theme
<· ▼{colors: Array(1024), visuals: {…}, plugins: {…}, set: ƒ} i
   ▶colors: (1024) ["#47A1FF", "#59CB74", "#FFB952", "#FC6980", "#6367EC", …
   ▶plugins: {Tooltip: {…}, Legend: {…}, Axis: {…}}
   ▶set: ƒ (obj)
   ▼visuals:
     ▶Area: {colors: Array(1024), styles: {…}, attrs: {…}}
     ▶Bar: {colors: Array(1024), styles: {…}, attrs: {…}}
     ▶Funnel: {colors: Array(1024), styles: {…}, attrs: {…}}
     ▶Gauge: {colors: Array(1024), attrs: {…}}
     ▼Line:
       ▶attrs: {statck: false}
       ▶colors: (1024) ["#47A1FF", "#59CB74", "#FFB952", "#FC6980", "#6367E…
       ▼styles:
         ▶guideline: {strokeColor: "#ddd"}
         ▶line: {lineWidth: 1}
         ▼point:
            borderWidth: 1
            pointType: "ellipse"
            radiusX: 4
            radiusY: 4
            strokeColor: "#fff"
```

如何实现组件化

数据驱动



B_r_

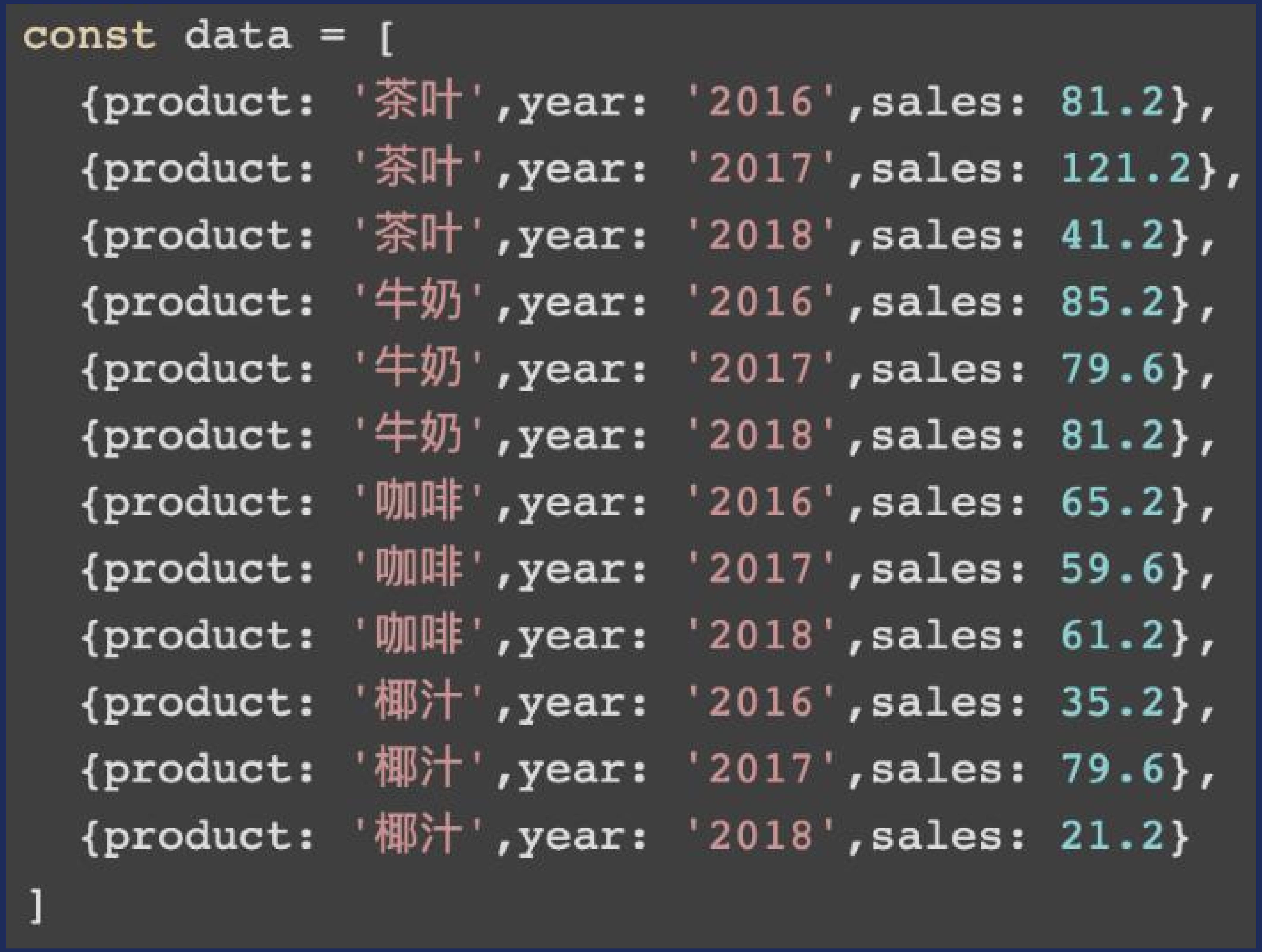
Tgqs_jq

B_r_qcr

Njsegl q

af _l ec

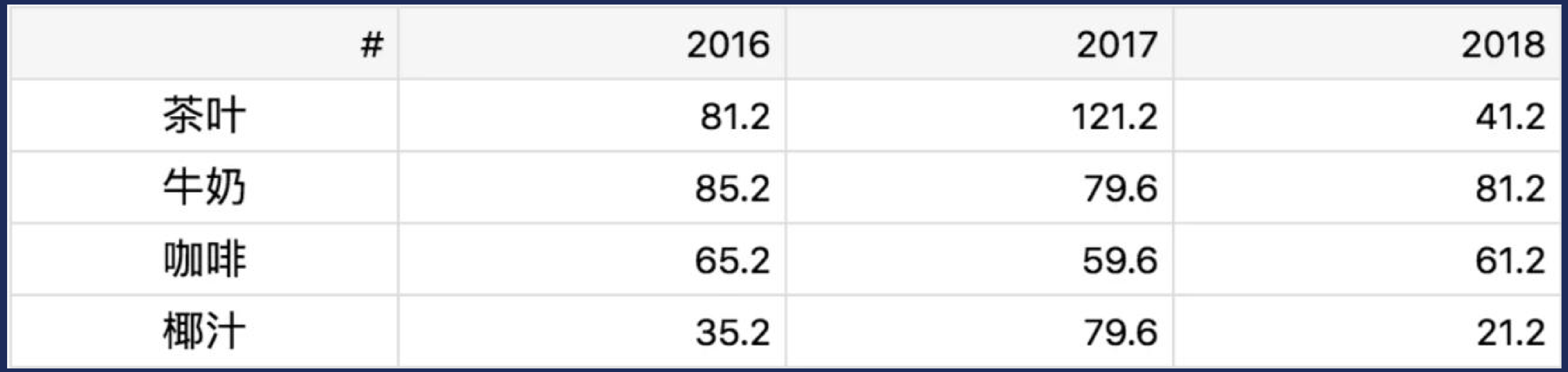
数据二维表



```
const data = [
  {product: '茶叶',year: '2016',sales: 81.2},
  {product: '茶叶',year: '2017',sales: 121.2},
  {product: '茶叶',year: '2018',sales: 41.2},
  {product: '牛奶',year: '2016',sales: 85.2},
  {product: '牛奶',year: '2017',sales: 79.6},
  {product: '牛奶',year: '2018',sales: 81.2},
  {product: '咖啡',year: '2016',sales: 65.2},
  {product: '咖啡',year: '2017',sales: 59.6},
  {product: '咖啡',year: '2018',sales: 61.2},
  {product: '椰汁',year: '2016',sales: 35.2},
  {product: '椰汁',year: '2017',sales: 79.6},
  {product: '椰汁',year: '2018',sales: 21.2}
]
```

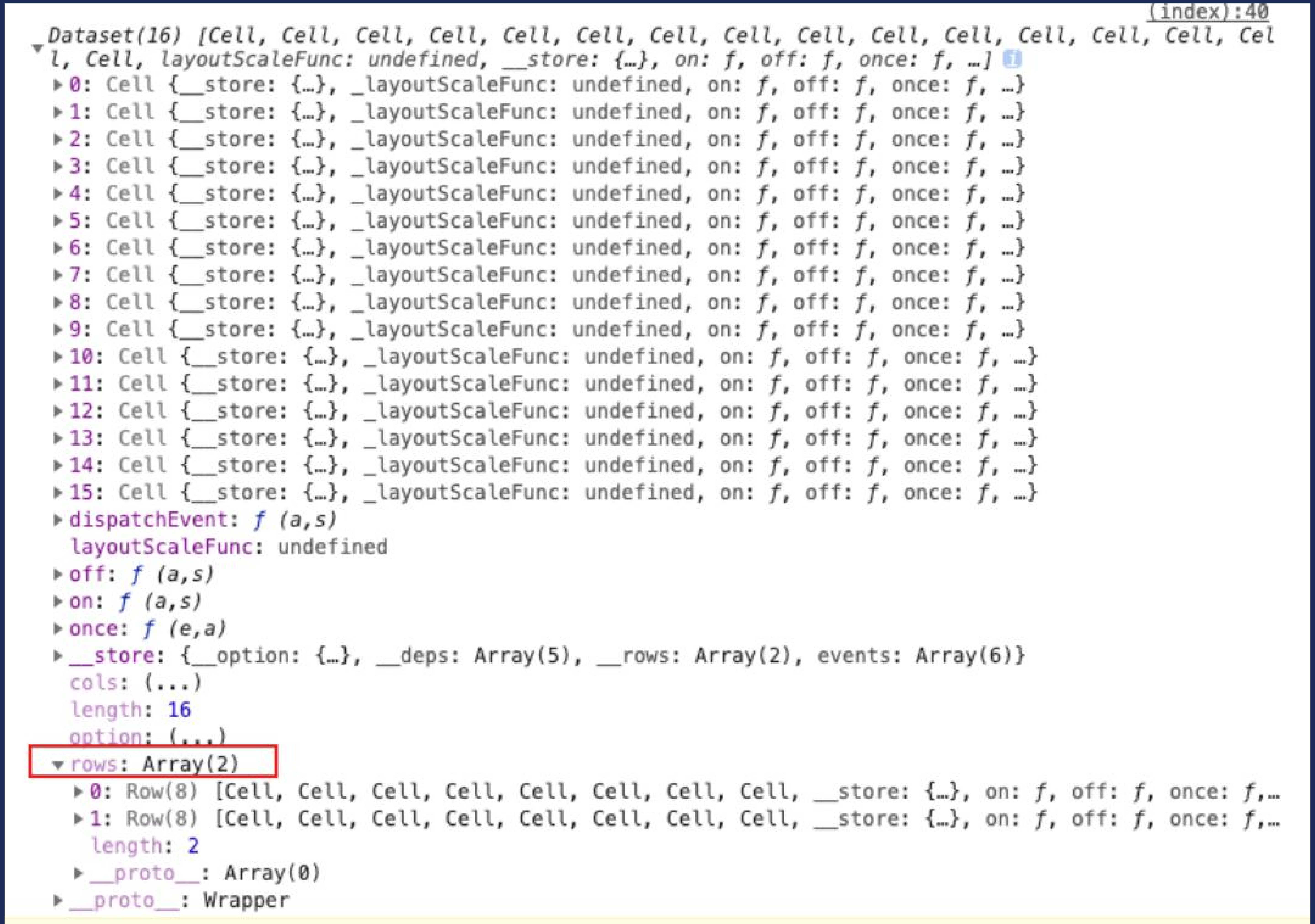
Rows: product

Cols: year



#	2016	2017	2018
茶叶	81.2	121.2	41.2
牛奶	85.2	79.6	81.2
咖啡	65.2	59.6	61.2
椰汁	35.2	79.6	21.2

Dataset



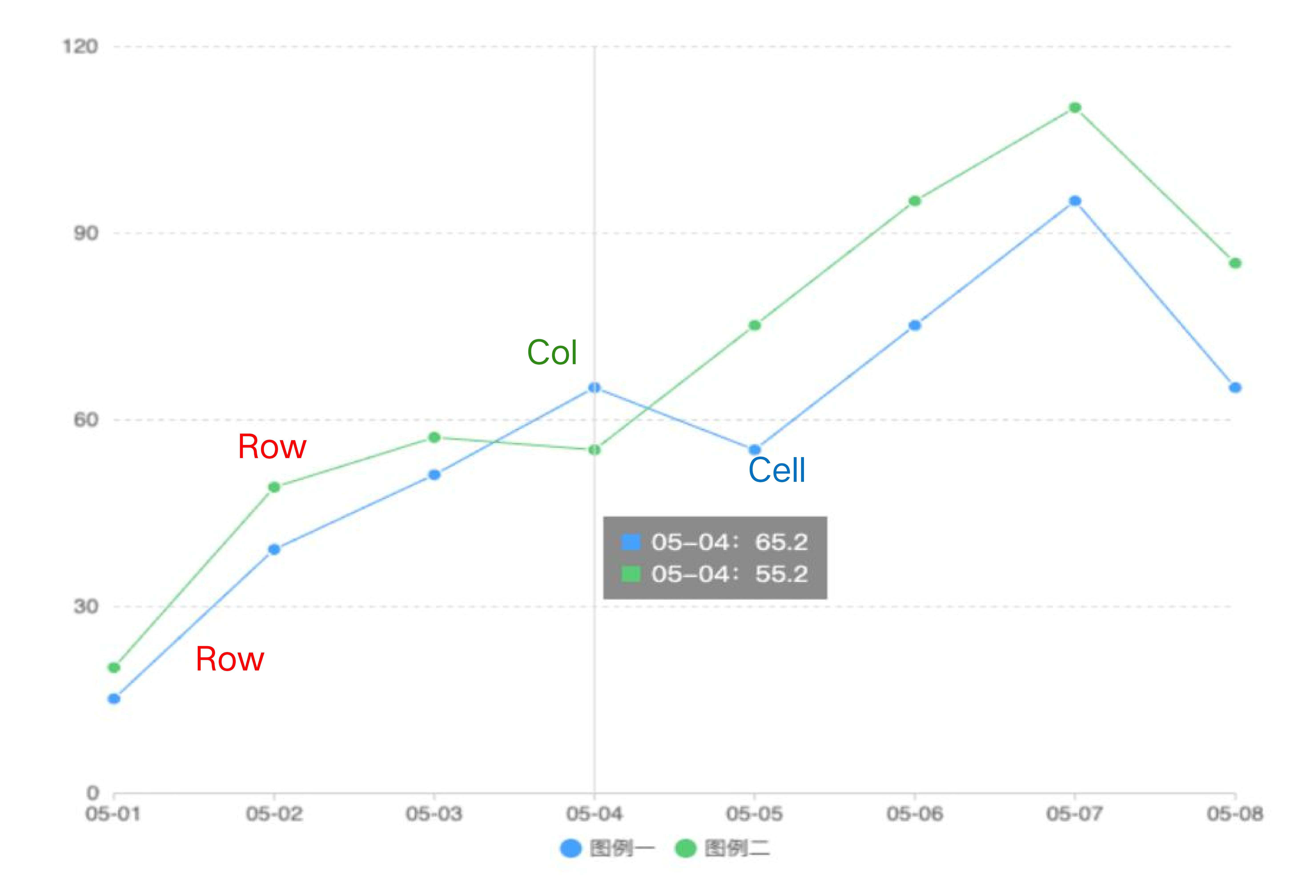
```
                                                                                    (index):40
▼Dataset(16) [Cell, Cell, Cell, Cell, Cell, Cell, Cell, Cell, Cell, Cell, Cell, Cell, Cell, Cell, Cel
 l, Cell, layoutScaleFunc: undefined, __store: {…}, on: f, off: f, once: f, …] ℹ
  ▶0: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶1: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶2: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶3: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶4: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶5: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶6: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶7: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶8: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶9: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶10: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶11: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶12: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶13: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶14: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶15: Cell {__store: {…}, _layoutScaleFunc: undefined, on: f, off: f, once: f, …}
  ▶dispatchEvent: f (a,s)
   layoutScaleFunc: undefined
  ▶off: f (a,s)
  ▶on: f (a,s)
  ▶once: f (e,a)
  ▶__store: {__option: {…}, __deps: Array(5), __rows: Array(2), events: Array(6)}
   cols: (...)
   length: 16
   option: (...)
  ▼rows: Array(2)
    ▶0: Row(8) [Cell, Cell, Cell, Cell, Cell, Cell, Cell, Cell, __store: {…}, on: f, off: f, once: f,…
    ▶1: Row(8) [Cell, Cell, Cell, Cell, Cell, Cell, Cell, Cell, __store: {…}, on: f, off: f, once: f,…
     length: 2
    ▶__proto__: Array(0)
  ▶__proto__: Wrapper
```

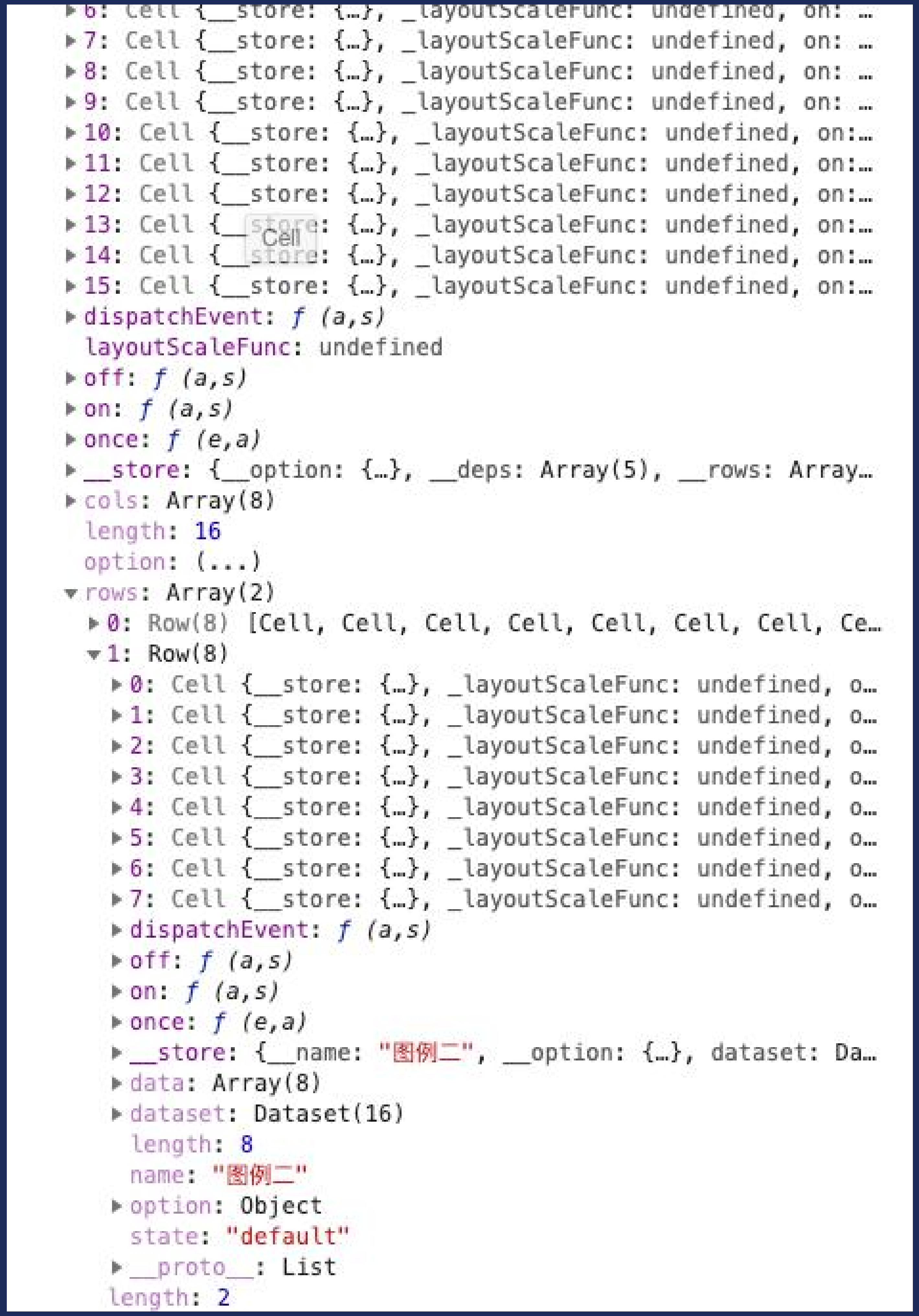
“为什么可以数据驱动

“

因为Dataset能够描述图表所有状态

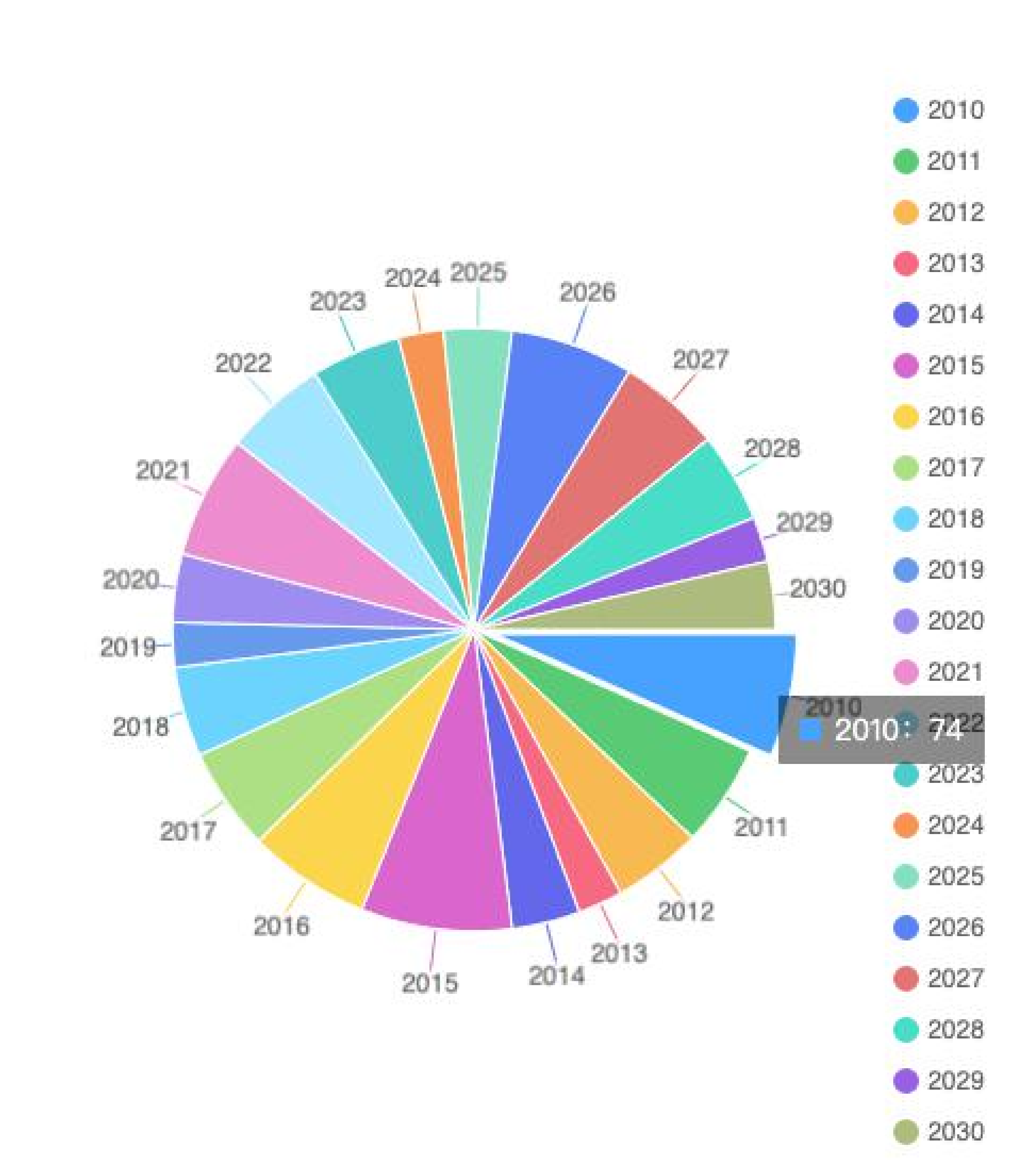
渲染显示

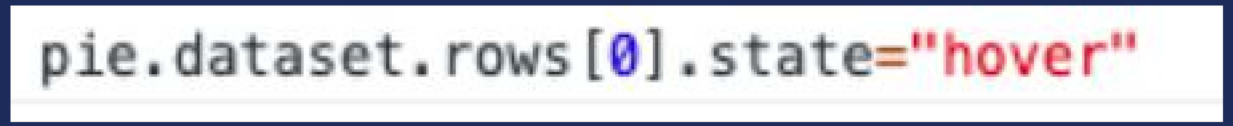




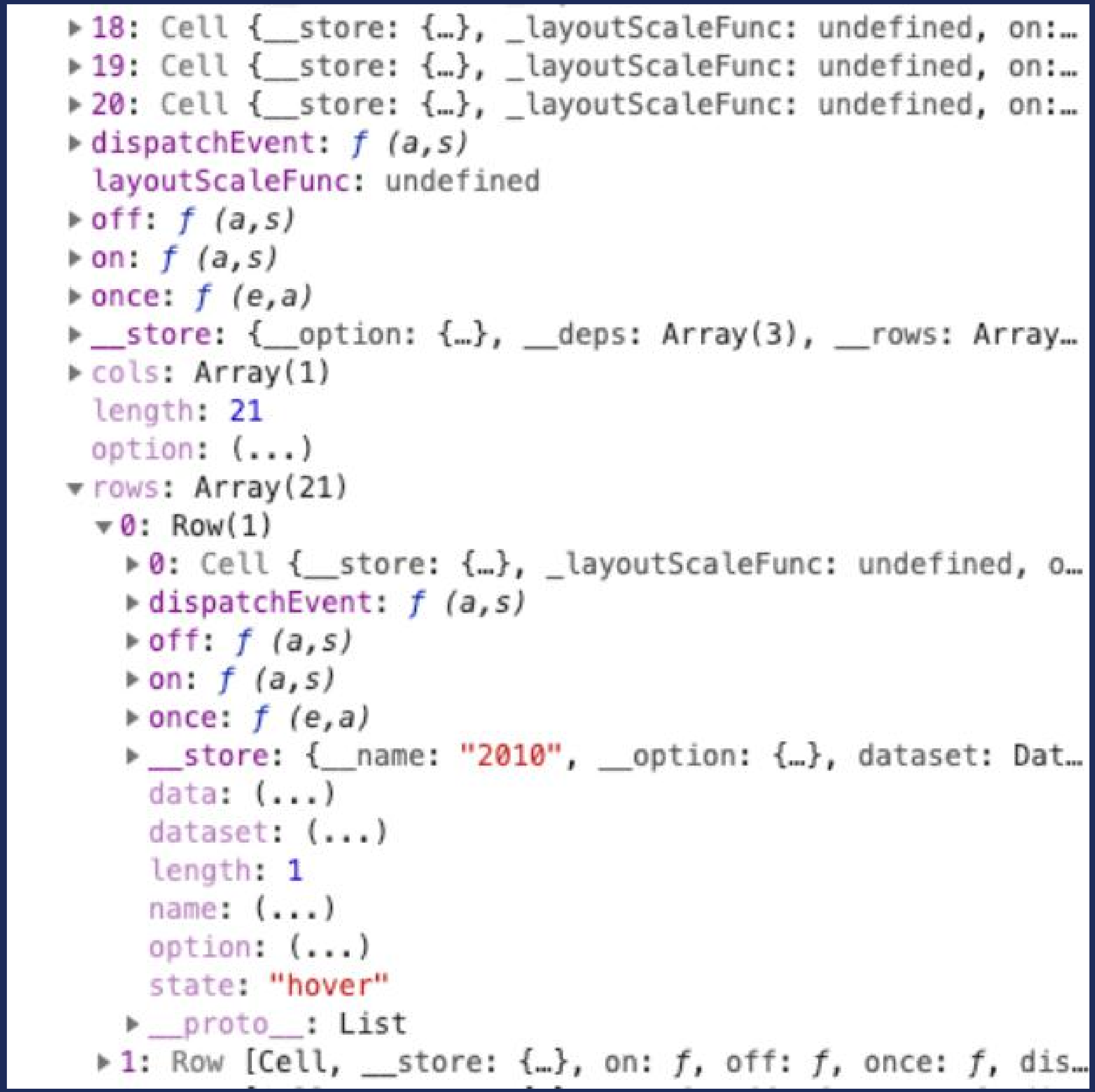
```
▸6: Cell {__store: {…}, _layoutScaleFunc: undefined, on: …
▸7: Cell {__store: {…}, _layoutScaleFunc: undefined, on: …
▸8: Cell {__store: {…}, _layoutScaleFunc: undefined, on: …
▸9: Cell {__store: {…}, _layoutScaleFunc: undefined, on: …
▸10: Cell {__store: {…}, _layoutScaleFunc: undefined, on:…
▸11: Cell {__store: {…}, _layoutScaleFunc: undefined, on:…
▸12: Cell {__store: {…}, _layoutScaleFunc: undefined, on:…
▸13: Cell {__store: {…}, _layoutScaleFunc: undefined, on:…
▸14: Cell {__store: {…}, _layoutScaleFunc: undefined, on:…
▸15: Cell {__store: {…}, _layoutScaleFunc: undefined, on:…
▸dispatchEvent: f (a,s)
 layoutScaleFunc: undefined
▸off: f (a,s)
▸on: f (a,s)
▸once: f (e,a)
▸__store: {__option: {…}, __deps: Array(5), __rows: Array…
▸cols: Array(8)
 length: 16
 option: (...)
▾rows: Array(2)
  ▸0: Row(8) [Cell, Cell, Cell, Cell, Cell, Cell, Cell, Ce…
  ▾1: Row(8)
    ▸0: Cell {__store: {…}, _layoutScaleFunc: undefined, o…
    ▸1: Cell {__store: {…}, _layoutScaleFunc: undefined, o…
    ▸2: Cell {__store: {…}, _layoutScaleFunc: undefined, o…
    ▸3: Cell {__store: {…}, _layoutScaleFunc: undefined, o…
    ▸4: Cell {__store: {…}, _layoutScaleFunc: undefined, o…
    ▸5: Cell {__store: {…}, _layoutScaleFunc: undefined, o…
    ▸6: Cell {__store: {…}, _layoutScaleFunc: undefined, o…
    ▸7: Cell {__store: {…}, _layoutScaleFunc: undefined, o…
    ▸dispatchEvent: f (a,s)
    ▸off: f (a,s)
    ▸on: f (a,s)
    ▸once: f (e,a)
    ▸__store: {__name: "图例二", __option: {…}, dataset: Da…
    ▸data: Array(8)
    ▸dataset: Dataset(16)
     length: 8
     name: "图例二"
    ▸option: Object
     state: "default"
    ▸__proto__: List
   length: 2
```

状态响应



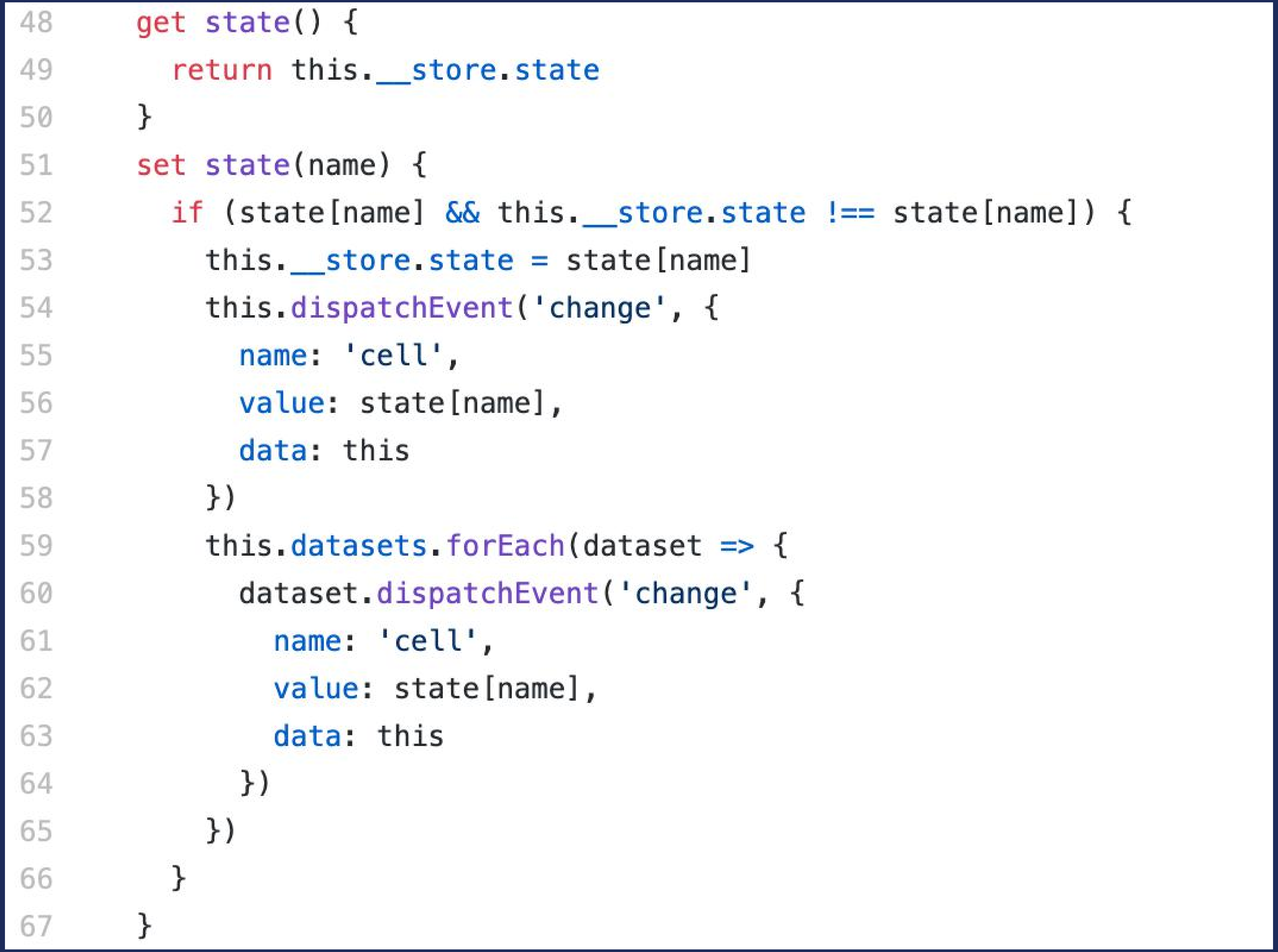


```
pie.dataset.rows[0].state="hover"
```



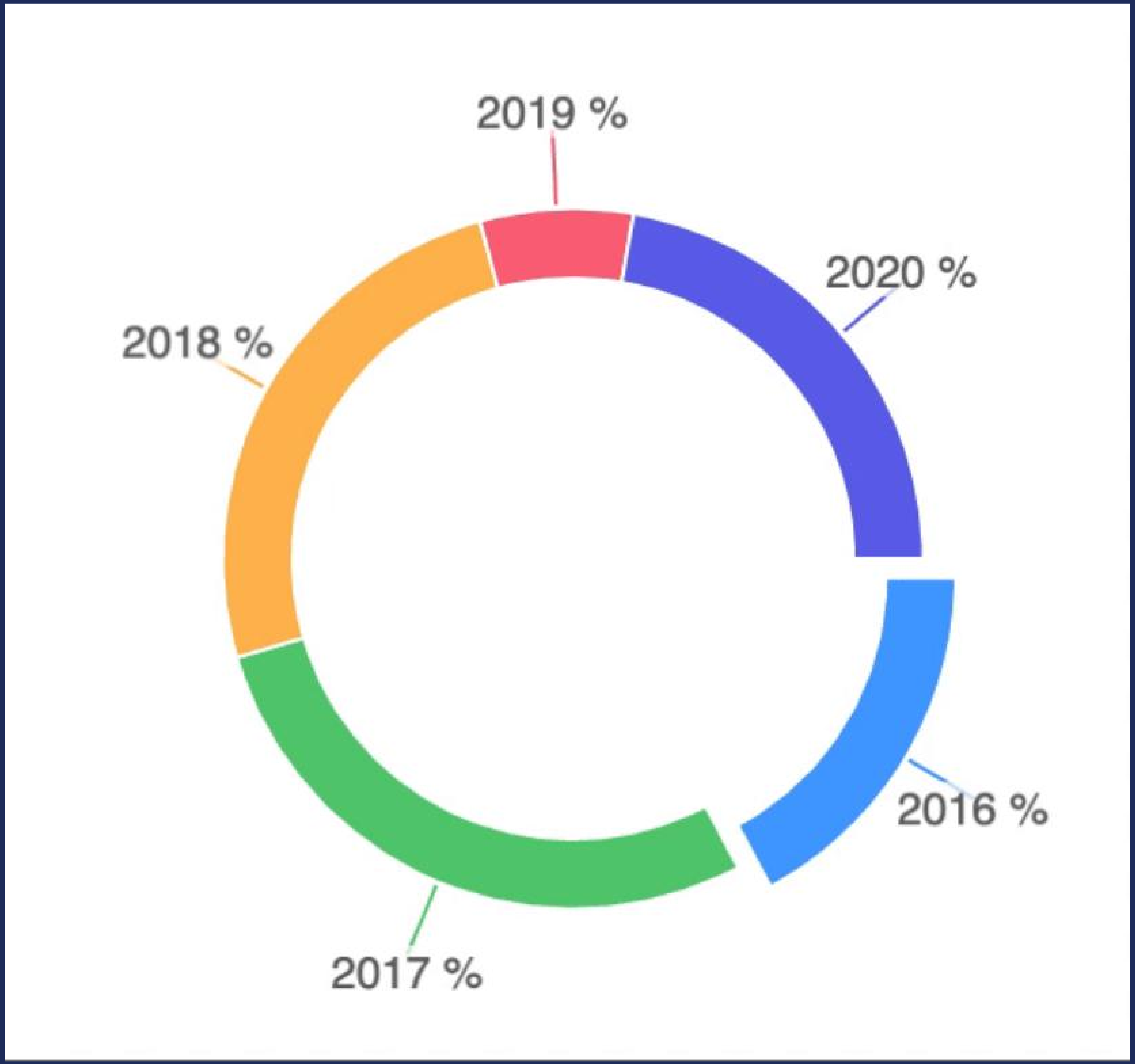
```
▶18: Cell {__store: {…}, _layoutScaleFunc: undefined, on:…
▶19: Cell {__store: {…}, _layoutScaleFunc: undefined, on:…
▶20: Cell {__store: {…}, _layoutScaleFunc: undefined, on:…
▶dispatchEvent: ƒ (a,s)
 layoutScaleFunc: undefined
▶off: ƒ (a,s)
▶on: ƒ (a,s)
▶once: ƒ (e,a)
▶__store: {__option: {…}, __deps: Array(3), __rows: Array…
▶cols: Array(1)
 length: 21
 option: (...)
▼rows: Array(21)
 ▼0: Row(1)
   ▶0: Cell {__store: {…}, _layoutScaleFunc: undefined, o…
   ▶dispatchEvent: ƒ (a,s)
   ▶off: ƒ (a,s)
   ▶on: ƒ (a,s)
   ▶once: ƒ (e,a)
   ▶__store: {__name: "2010", __option: {…}, dataset: Dat…
    data: (...)
    dataset: (...)
    length: 1
    name: (...)
    option: (...)
    state: "hover"
   ▶__proto__: List
 ▶1: Row [Cell, __store: {…}, on: ƒ, off: ƒ, once: ƒ, dis…
```

Cell.js

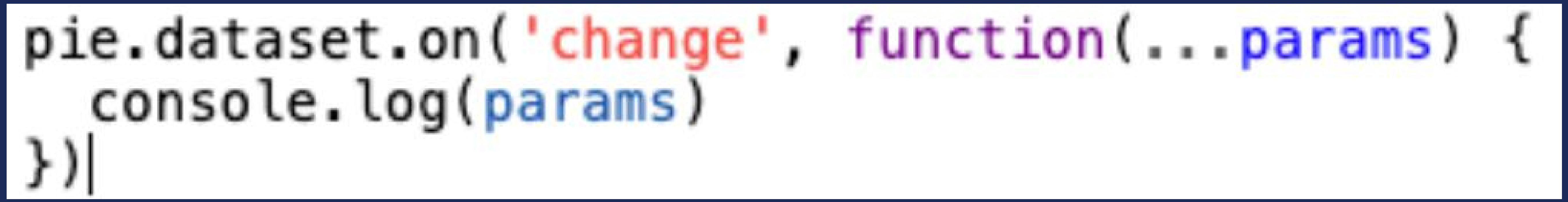


```
  get state() {
    return this.__store.state
  }
  set state(name) {
    if (state[name] && this.__store.state !== state[name]) {
      this.__store.state = state[name]
      this.dispatchEvent('change', {
        name: 'cell',
        value: state[name],
        data: this
      })
      this.datasets.forEach(dataset => {
        dataset.dispatchEvent('change', {
          name: 'cell',
          value: state[name],
          data: this
        })
      })
    }
  }
```

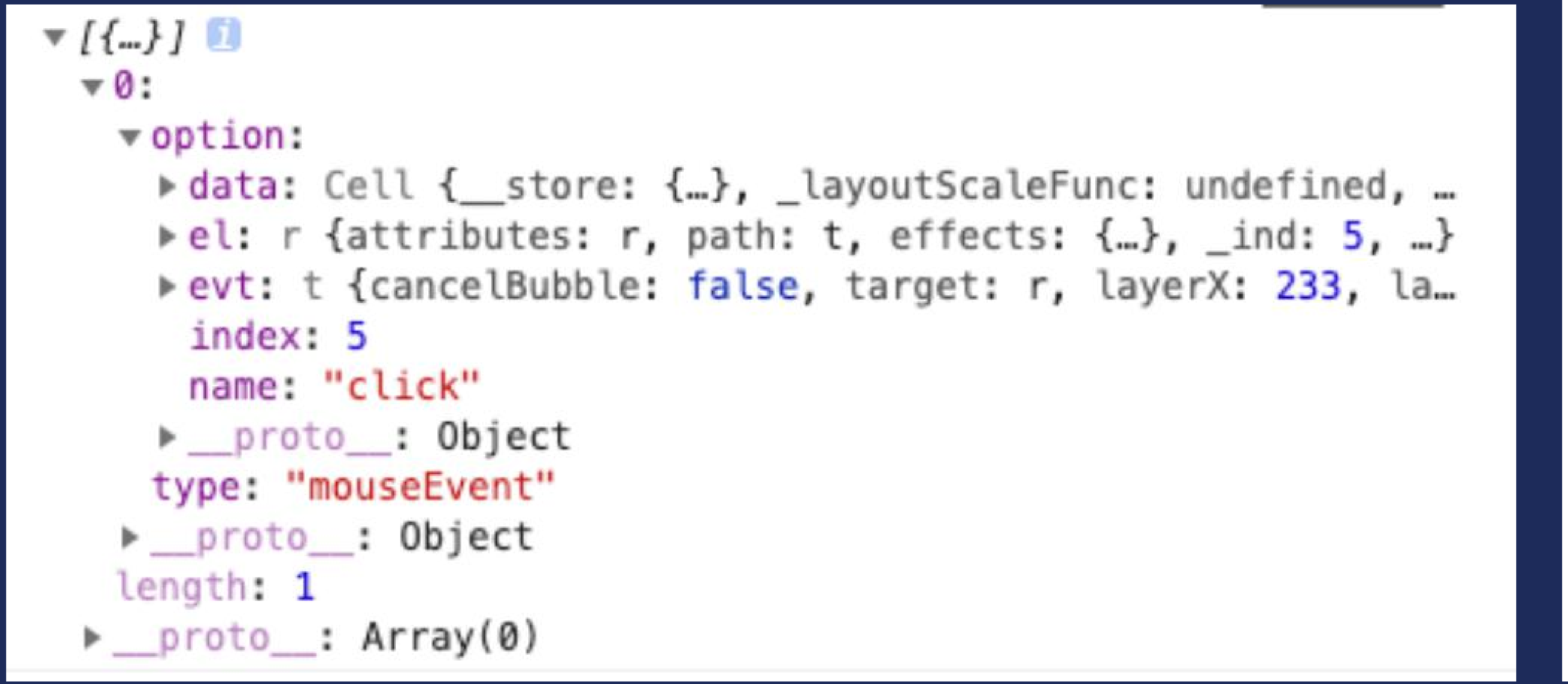
数据驱动



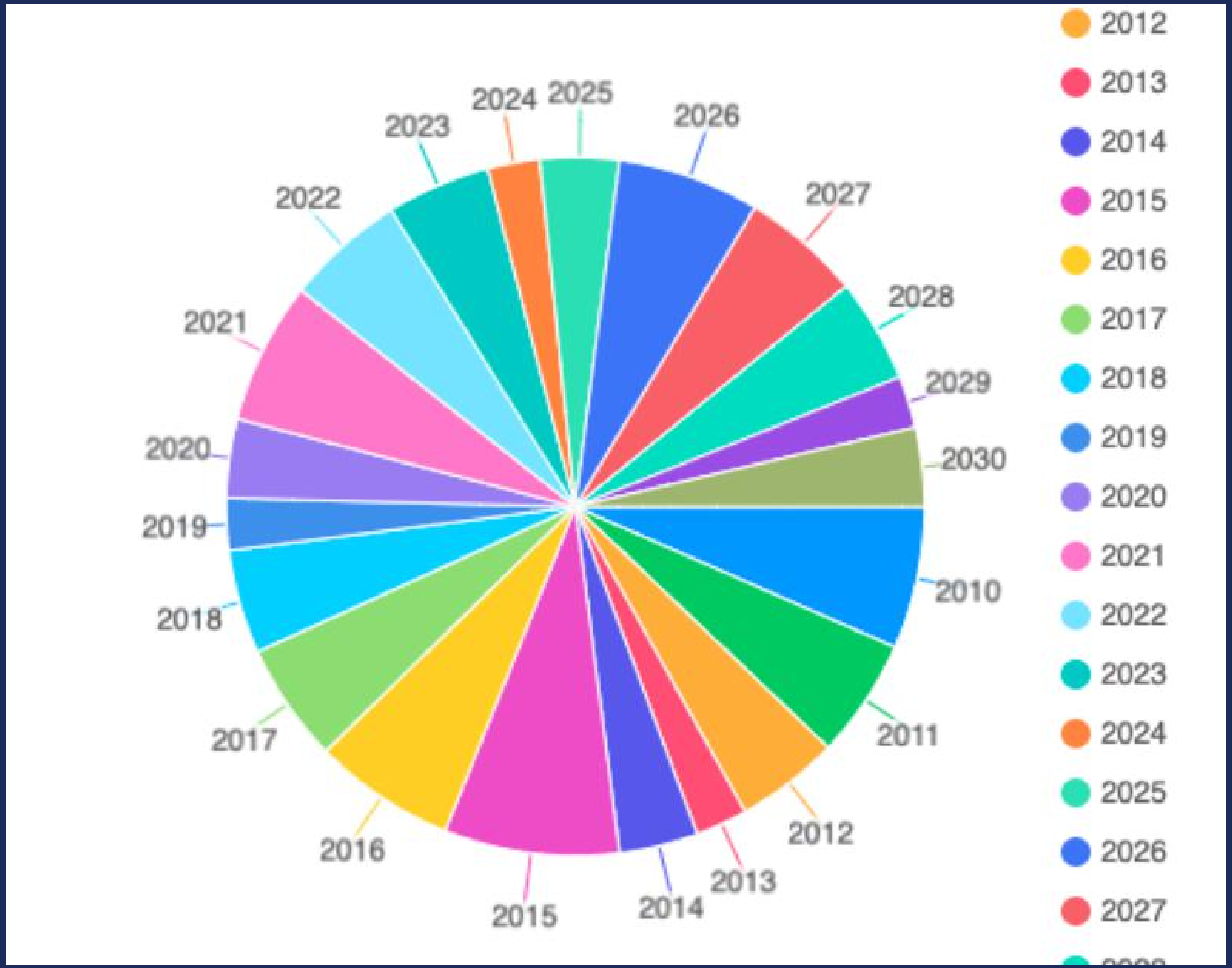
数据与事件



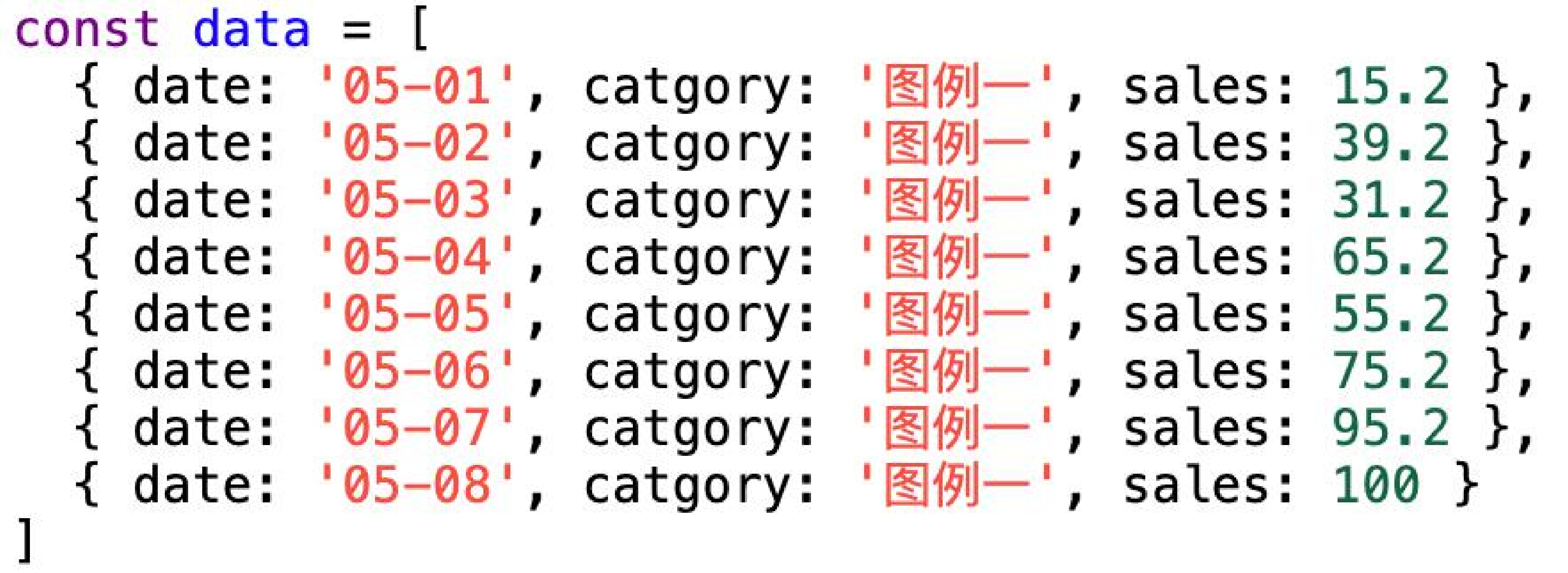
```
pie.dataset.on('change', function(...params) {
  console.log(params)
})
```



```
▼ [{…}] i
  ▼ 0:
    ▼ option:
      ▶ data: Cell {__store: {…}, _layoutScaleFunc: undefined, …
      ▶ el: r {attributes: r, path: t, effects: {…}, _ind: 5, …}
      ▶ evt: t {cancelBubble: false, target: r, layerX: 233, la…
        index: 5
        name: "click"
      ▶ __proto__: Object
      type: "mouseEvent"
    ▶ __proto__: Object
    length: 1
  ▶ __proto__: Array(0)
```



数据驱动



```
const data = [
  { date: '05-01', catgory: '图例一', sales: 15.2 },
  { date: '05-02', catgory: '图例一', sales: 39.2 },
  { date: '05-03', catgory: '图例一', sales: 31.2 },
  { date: '05-04', catgory: '图例一', sales: 65.2 },
  { date: '05-05', catgory: '图例一', sales: 55.2 },
  { date: '05-06', catgory: '图例一', sales: 75.2 },
  { date: '05-07', catgory: '图例一', sales: 95.2 },
  { date: '05-08', catgory: '图例一', sales: 100 }
]
```

B_r_

Line,Pie,Area···

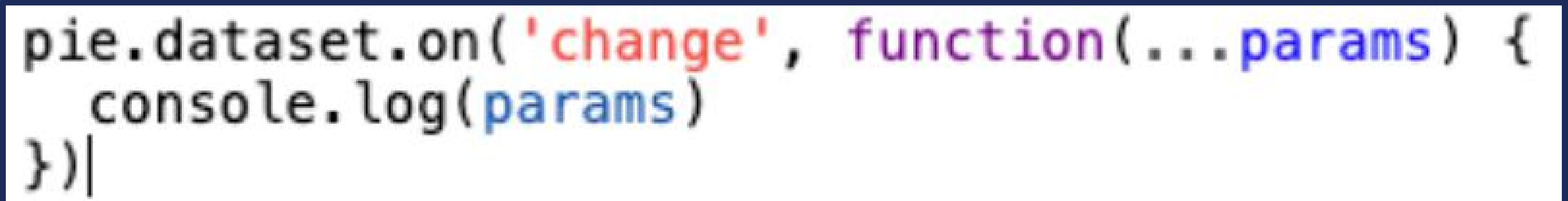
Tgqs_jq

B_r_qcr

Njsedg q

Axis,Tooltip,Legend···

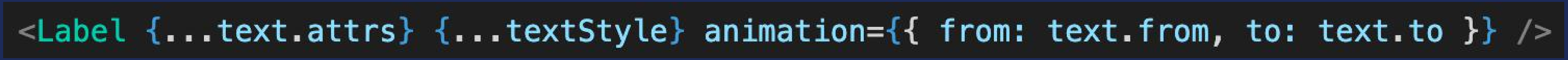
af _l ec



```
pie.dataset.on('change', function(...params) {
  console.log(params)
})
```

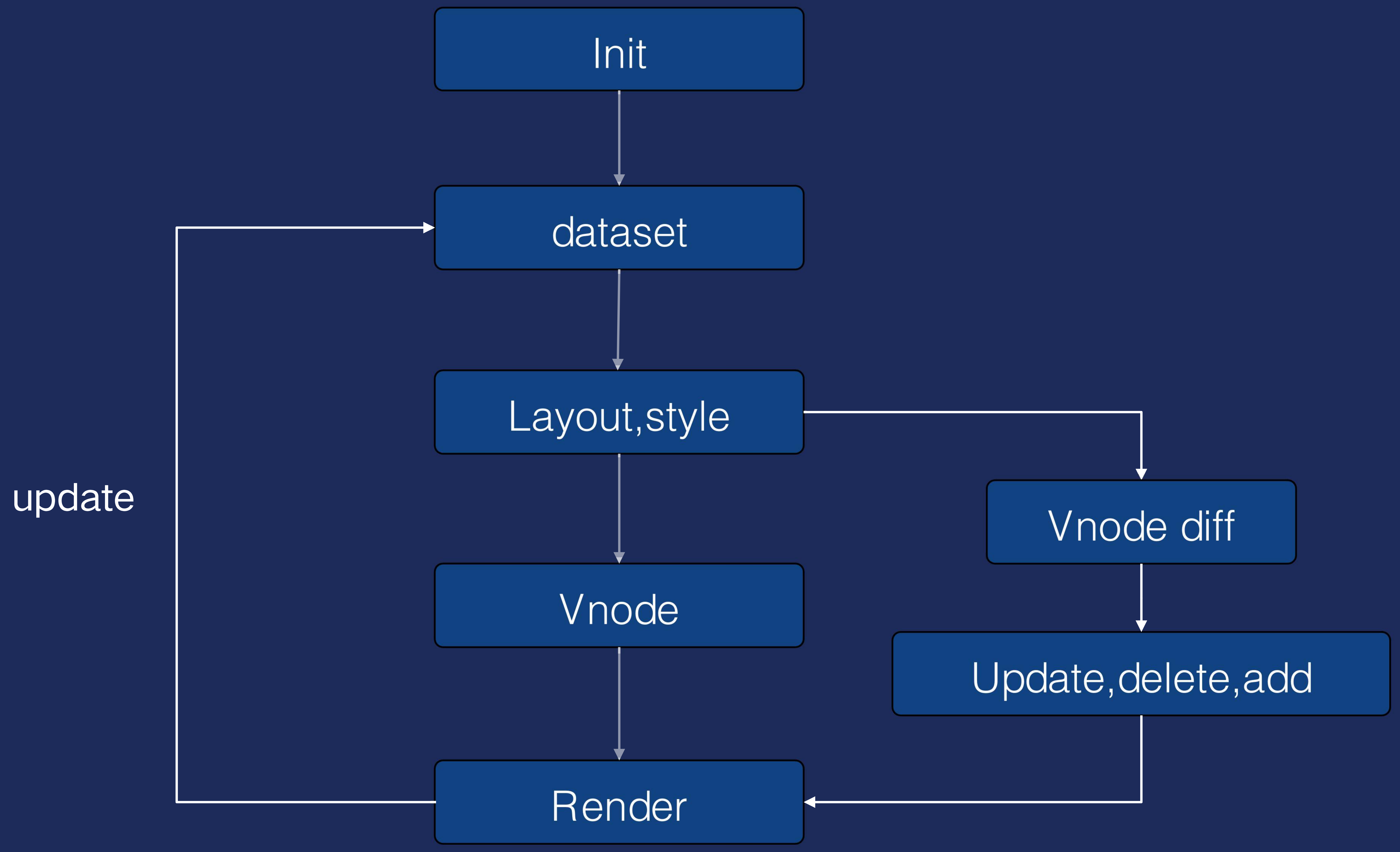
“ 可以开心的写码了

组件-JSX



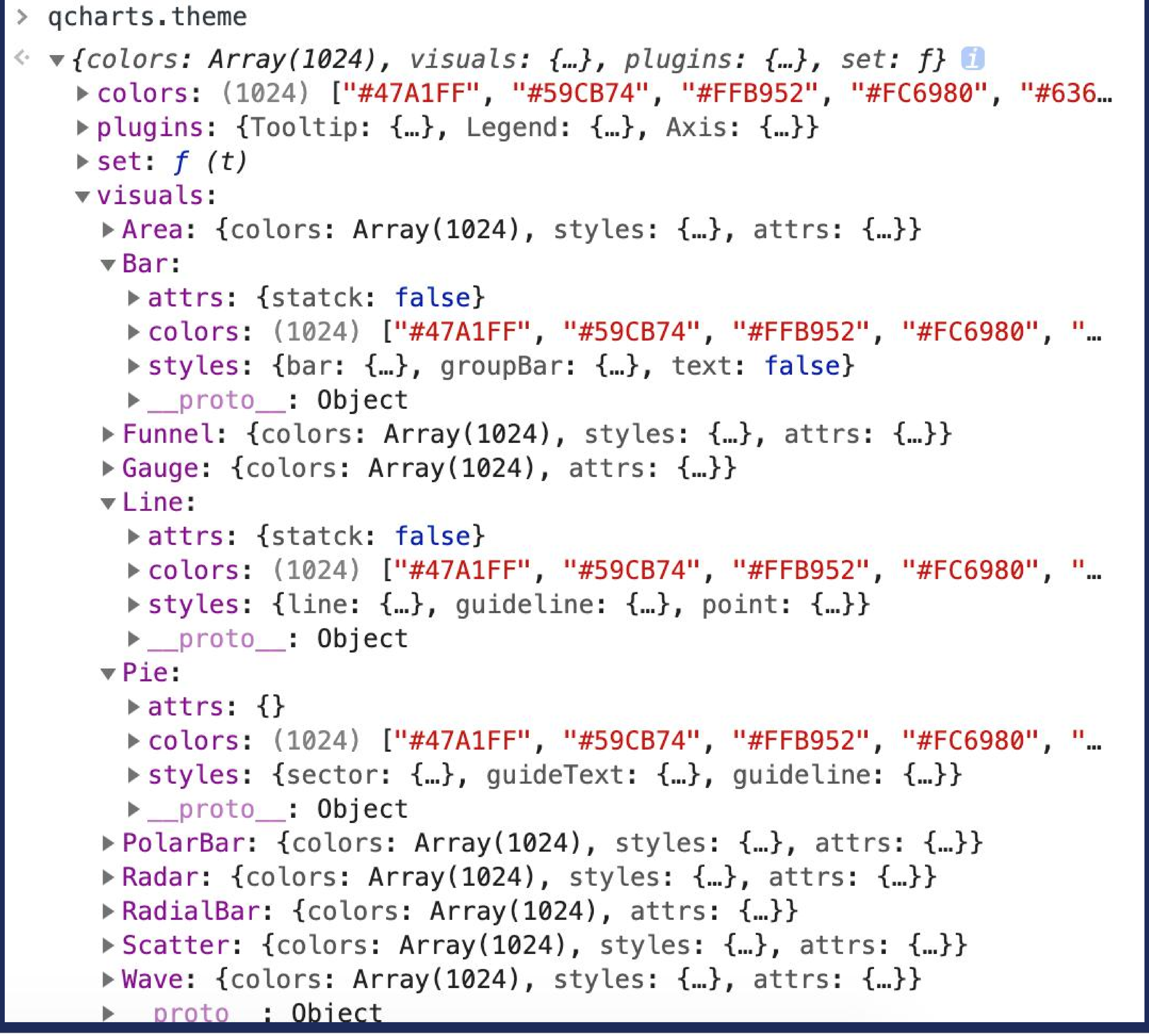
```
<Label {...text.attrs} {...textStyle} animation={{ from: text.from, to: text.to }} />
```

组件-流程


Init
dataset
Layout,style
Vnode
Render
update
Vnode diff
Update,delete,add

适配

属性，样式

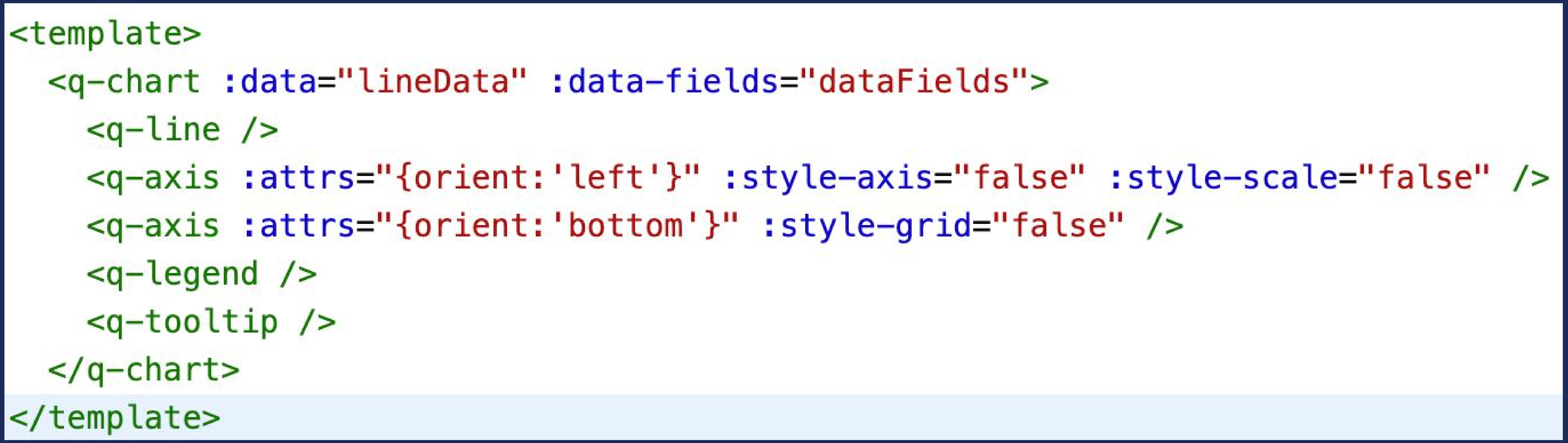


```
> qcharts.theme
<· ▼{colors: Array(1024), visuals: {…}, plugins: {…}, set: f} i
   ▶colors: (1024) ["#47A1FF", "#59CB74", "#FFB952", "#FC6980", "#636…
   ▶plugins: {Tooltip: {…}, Legend: {…}, Axis: {…}}
   ▶set: f (t)
   ▼visuals:
     ▶Area: {colors: Array(1024), styles: {…}, attrs: {…}}
     ▼Bar:
       ▶attrs: {statck: false}
       ▶colors: (1024) ["#47A1FF", "#59CB74", "#FFB952", "#FC6980", "…
       ▶styles: {bar: {…}, groupBar: {…}, text: false}
       ▶__proto__: Object
     ▶Funnel: {colors: Array(1024), styles: {…}, attrs: {…}}
     ▶Gauge: {colors: Array(1024), attrs: {…}}
     ▼Line:
       ▶attrs: {statck: false}
       ▶colors: (1024) ["#47A1FF", "#59CB74", "#FFB952", "#FC6980", "…
       ▶styles: {line: {…}, guideline: {…}, point: {…}}
       ▶__proto__: Object
     ▼Pie:
       ▶attrs: {}
       ▶colors: (1024) ["#47A1FF", "#59CB74", "#FFB952", "#FC6980", "…
       ▶styles: {sector: {…}, guideText: {…}, guideline: {…}}
       ▶__proto__: Object
     ▶PolarBar: {colors: Array(1024), styles: {…}, attrs: {…}}
     ▶Radar: {colors: Array(1024), styles: {…}, attrs: {…}}
     ▶RadialBar: {colors: Array(1024), attrs: {…}}
     ▶Scatter: {colors: Array(1024), styles: {…}, attrs: {…}}
     ▶Wave: {colors: Array(1024), styles: {…}, attrs: {…}}
     ▶__proto__: Object
```

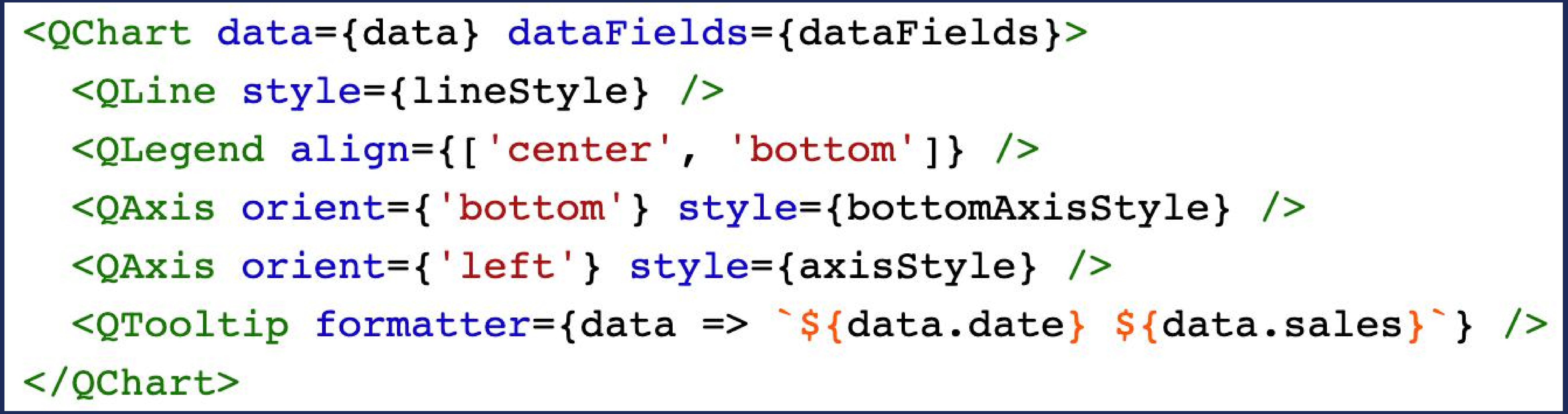
适配

> “
> 我只负责透传，其它啥也没干

适配



```
<template>
  <q-chart :data="lineData" :data-fields="dataFields">
    <q-line />
    <q-axis :attrs="{orient:'left'}" :style-axis="false" :style-scale="false" />
    <q-axis :attrs="{orient:'bottom'}" :style-grid="false" />
    <q-legend />
    <q-tooltip />
  </q-chart>
</template>
```



```
<QChart data={data} dataFields={dataFields}>
  <QLine style={lineStyle} />
  <QLegend align={['center', 'bottom']} />
  <QAxis orient={'bottom'} style={bottomAxisStyle} />
  <QAxis orient={'left'} style={axisStyle} />
  <QTooltip formatter={data => `${data.date} ${data.sales}`} />
</QChart>
```

17
50
182
619
869
存储机构
1577
89
307
495
100
80
60
40
20
0
22
76
87
120
90
60
30
0
1月
2月
3月
4月
5月
6月
7月
8月
9月
10月
11月
12月
0
20
40
60
80
100
0%
0%
0%
0%
100%

未来、展望

> “
>
> 完善、优化

InfoQ | 写作平台

☑ InfoQ 写作平台是 InfoQ 开放给开发者的高端技术社区，创作者可以在这里自由创作和发布内容。

☑ 写作平台将为创作者提供签约、培训、资金扶持等一系列权益，助力作者成长为高精尖技术人才；同时也为企业提供品牌、活动打造、内容传播等服务，与伙伴一同成长。

扫码申请创作者
企业/个人均可申请
→

扫码进入写作平台
企业/个人均可申请
→

GMTC

全球大前端技术大会

THANKS

InfoQ