Graphic类 API文档

1. showRectGraph (element, config) 绘制柱状图

使用方法：

var rectGraph = new Graphic();

var rectData=[   
 "09:00": "12",  
 "10:00": "1",  
 "11:00": "19",

];

//绘制柱形图  
rectGraph.showRectGraph("#rect", {

//画布高度，必须

height: 200,

//柱形数据，必须

data: rectData,

//边距，非必须

padding:{

top:10,

left:50

}

//是否显示x轴，非必须

xAxis:false,

//柱形样式，非必须

className:”rect-class”,  
 //点击柱形，非必须

onClick: function (param) {  
 },

//鼠标悬浮，非必须

onMouseover: function (param) {  
 },

//鼠标离开，非必须

onMouseout: function (param) {  
 },

//x轴刻度间距，非必须

xDistance: 2  
});

|  |  |  |
| --- | --- | --- |
| 入参 | 参数描述 | 参数详情 |
| element | 表示绘制在哪个容器中 | 例如：  ”.className”、  “#id” |
| config | 柱状图配置：  {  height: 400,  data: {  key1: value1,  key2: value2  …  },  padding:{  top:20,  right:30,  bottom:20,  left  },  xAxis:true,  className:”rect-class”,  xDistance: 1,  onClick:function(param){},  onMouseover:function(param){},  onMouseout:function(param){}  } | height: 画布高度，默认为0  data: key为X轴刻度文字  padding:柱状图与画布的边距，默认为{top:20,right:30,bottom:20,left}，可设置部分值，见示例  xAxis:是否显示x轴  className: 柱体样式，默认为” default-rect”  xDistance: x轴相邻刻度距离，默认为1  onClick:  //点击柱形触发事件  //param.index:第几个柱形  //param.value:柱形对应Y轴值  onMouseover:  //鼠标悬浮触发事件  //param.index:第几个柱形  //param.value:柱形对应Y轴值  onMouseout:  //鼠标离开触发事件  //param.index:第几个柱形  //param.value:柱形对应Y轴值 |

1. updateRectGraph(data)

data同showRectGraph中的config.data

1. showLineGraph (element, config)绘制曲线图

使用方法：

var lineGraph = new Graphic();

var lineData=[   
 "09:00": "12",  
 "10:00": "1",  
 "11:00": "19",

];

//绘制曲线图  
lineGraph.showLineGraph("#line", {

//画布高度，必须

height: 200,

//曲线图数据，必须

data: lineData,

//边距，非必须

padding:{

top:10,

left:50

}

//鼠标悬浮圆点上方是否放大，非必须

pinch:false,

//是否显示x轴，非必须

xAxis:false,

//点击曲线上的圆点，非必须

onClick: function (param) {  
 },

//鼠标悬浮，非必须

onMouseover: function (param) {  
 },

//鼠标离开，非必须

onMouseout: function (param) {  
 },

//x轴刻度间距，非必须

xDistance: 2  
});

|  |  |  |
| --- | --- | --- |
| 入参 | 参数描述 | 参数详情 |
| element | 表示绘制在哪个容器中 | 例如：  ”.className”、  “#id” |
| config | 曲线图配置：  {  height: 400,  data: {  key1: value1,  key2: value2  …  },  pinch:true,  xAxis:true,  xDistance: 1,  onClick:function(param){},  onMouseover:function(param){},  onMouseout:function(param){}  } | height: 画布高度，默认为0  data: key为X轴刻度文字  pinch:鼠标悬浮在散点上时是否显示动画放大效果，默认为true  xAxis:是否显示x轴  xDistance: x轴相邻刻度距离，默认为1  onClick:  //点击曲线上的圆点触发事件  //param.index:第几个点  //param.value:对应Y轴值  onMouseover:  //鼠标悬浮触发事件  //param.index:第几个点  //param.value:对应Y轴值  onMouseout:  //鼠标离开触发事件  //param.index:第几个点  //param.value:对应Y轴值 |

1. updateLineGraph(data)

data同showLineGraph中的config.data

1. showPointGraph(element, config)

使用方法：

var pointGraph = new Graphic();

//模拟数据  
function getTestPointData() {  
 //模拟散点数据  
 var pointData = {};  
 var user = ["user01", "user02", "user03", "user04", "user05", "user06", "user07", "user08"];  
 var date = ["06-01", "06-02", "06-03", "06-04", "06-05", "06-06", "06-07", "06-08"];  
 for(var i=0;i<user.length;i++){  
 pointData[user[i]] = [];  
 for(var j=0;j<date.length;j++){  
 pointData[user[i]].push({  
 mark : date[j],  
 value : Math.ceil(Math.random()\*10)  
 });  
 }  
 }  
 return pointData;  
}

var pointData = getTestPointData();

//绘制曲线图  
pointGraph. showPointGraph("#point", {

//画布高度，必须

height: 200,

//散点数据，必须

data: pointData,

//边距，非必须

padding:{

top:10,

left:50

}

//鼠标悬浮散点上方是否放大，非必须

pinch:false,

//散点最大半径，非必须

pointSize:10,

//是否显示x轴，非必须

xAxis:false,

//点击散点，非必须

onClick: function (param) {  
 },

//鼠标悬浮，非必须

onMouseover: function (param) {  
 },

//鼠标离开，非必须

onMouseout: function (param) {  
 },

//x轴刻度间距，非必须

xDistance: 2  
});

|  |  |  |
| --- | --- | --- |
| 入参 | 参数描述 | 参数详情 |
| element | 表示绘制在哪个容器中 | 例如：  ”.className”、  “#id” |
| config | 散点图配置：  {  height: 400,  data: {  key1:[{  mark:””,  value:””  },{  mark:””,  value:””  }],  key2: [{  mark:””,  value:””  },{  mark:””,  value:””  }]  …  },  pinch:true,  pointSize:10,  xAxis:true,  xDistance: 1,  onClick:function(param){},  onMouseover:function(param){},  onMouseout:function(param){}  } | height: 画布高度，默认为0  data: key为y轴刻度文字，mark值为x轴刻度文字，value值为散点值  pinch:鼠标悬浮在散点上时是否显示动画放大效果，默认为true  pointSize:散点最大半径，默认为15  xAxis:是否显示x轴  xDistance: x轴相邻刻度距离，默认为1  onClick:  //点击曲线上的圆点触发事件  //param.index:第几个点  //param.value:对应Y轴值  onMouseover:  //鼠标悬浮触发事件  //param.index:第几个点  //param.value:对应Y轴值  onMouseout:  //鼠标离开触发事件  //param.index:第几个点  //param.value:对应Y轴值 |

1. updatePointGraph(data)

data同showPointGraph中的config.data

1. showMapGraph(element, config)

使用方法：

var mapGraph = new Graphic();

var mapData = [

[118, 36, 9],

[100, 33, 9],

[108, 30, 7]

];

//绘制曲线图  
mapGraph. showPointGraph("#map", {

//画布高度，必须

height: 200,

//地图geojson文件路径，必须

mapPath:” map-json/china.geo.json”,

//散点数据，必须

data: mapData,

//安全值，非必须

safeLine:6,

//边距，非必须

padding:{

top:10,

left:50

}

//鼠标悬浮散点上方是否放大，非必须

pinch:false,

//散点最大半径，非必须

pointSize:10,

//点击散点，非必须

onClick: function (param) {  
 },

//鼠标悬浮，非必须

onMouseover: function (param) {  
 },

//鼠标离开，非必须

onMouseout: function (param) {  
 },

//x轴刻度间距，非必须

xDistance: 2  
});

|  |  |  |
| --- | --- | --- |
| 入参 | 参数描述 | 参数详情 |
| element | 表示绘制在哪个容器中 | 例如：  ”.className”、  “#id” |
| config | 曲线图配置：  {  height: 400,  data: [  [lng, lat, value],  [lng, lat, value]  …  ],  mapPath:” map-json/china.geo.json  ”,  safeLine:6,  pinch:true,  pointSize:10,  xDistance: 1,  onClick:function(param){},  onMouseover:function(param){},  onMouseout:function(param){}  } | height: 画布高度，默认为0  data: lng为经度值，lat为纬度值，value值为数据大小  mapPath:地图geojson文件相对于resources文件夹的路径  safeLine:安全值，超过安全值的散点会用扩散动画显示，如果没有则不显示动画  pinch:鼠标悬浮在散点上时是否显示动画放大效果，默认为true  pointSize:散点最大半径，默认为15  xDistance: x轴相邻刻度距离，默认为1  onClick:  //点击曲线上的圆点触发事件  //param.index:第几个点  //param.value:对应Y轴值  onMouseover:  //鼠标悬浮触发事件  //param.index:第几个点  //param.value:对应Y轴值  onMouseout:  //鼠标离开触发事件  //param.index:第几个点  //param.value:对应Y轴值 |

1. updateMapGraph(data)

data同showMapGraph中的config.data