

# Amcharts JS 版设置属性 / 方法详细

## 1、坐标轴（ Y 轴 ）

序号	属性名 / 方法名	作用	对象获取方法 / 常用属性值	示例
1	valueAxis 对象	图表的 Y 轴，一个图表中可以有多 Y 轴	Var valueAxis = new AmCharts.ValueAxis();	
2	axisColor	轴的颜色		valueAxis.axisColor = "#FF6600" ;
3	axisThickness	轴的宽度		valueAxis.axisThickness = 1;
4	gridAlpha	轴的透明度，值介于 0-1 之间，0 全透明		valueAxis1.gridAlpha = 0.2;
5	tickLength	轴从下到上像左或右伸出来的延长线		valueAxis1.tickLength =0;
6	minimum	轴的最小值，如果不设置那么最小值根据数据动态变化		valueAxis1.minimum = - 100;
7	maximum	轴的最大值，如果不设置那么最大值根据数据动态变化		valueAxis1.maximum = 100;
8	title	轴的名称		valueAxis1.title= " 哈哈 " ;
9	logarithmic	是否为对数函数分布，一般轴的刻度是均匀划分的，当该属性设置为 true 的时候，刻度分布呈对数形式		valueAxis1.logarithmic = false ;

		分布		
10	integersOnly	是否只显示整数，如果为 true 轴的刻度只显示整数形式		valueAxis1.integersOnly = true ;
11	gridCount	最大刻度个数		valueAxis1.gridCount = 10;
12	unit	单位		valueAxis1.unit = "%" ;
13	labelsEnabled	是否显示轴标签，默认值为 true		valueAxis1.labelsEnabled = true ;
14	inside	轴的刻度值显示在表里面还是外面		valueAxis1.inside = true ;
15	position	轴的位置，默认在左侧		valueAxis1.position = "left" ;
16	stackType			valueAxis.stackType = "0%";

2、 categoryAxis （图表线，相当于 X轴）

序号	属性名 / 方法名	作用	对象获取方法 / 常用属性值	示例
1	valueAxis 对象	图表的线，一个图表中可以有多条，每个对应一个 Y 轴或者共同拥有一个 Y 轴	var categoryAxis = chart.categoryAxis;	
2	parseDates	是否以日期为 x 轴的值	True 、 false	categoryAxis.parseDates = false ;
3	minPeriod	当以日期为 x 轴的时候 x 轴显示的最小范围	SS : 分钟 DD : 天	categoryAxis.minPeriod = "SS"
4	dashLength	破折线长度，默认为 0 是实心线		categoryAxis.dashLength = 1;
5	gridAlpha	网格的透明度，垂直 x 轴的刻度线形成网格		categoryAxis.gridAlpha = 0.15;

6	axisColor	轴的颜色		categoryAxis.axisColor = "#DADADA" ;
7	position	轴的位置，默认在最下方	top ：显示在上方    left: 左侧    right ：右侧	categoryAxis.position = "top" ;
8	gridPosition	网格位置		categoryAxis.gridPosition = "start" ;
9	startOnAxis	是否从轴上开始绘制，默认为 false ，即第一个点绘制是从中间开始的，当设置为 true 的时候，第一个点开始总是从 Y 轴上开始，结束总是在最后一个跟 Y 轴平行的轴上结束	true 、 false	categoryAxis.startOnAxis = true ;
10	gridColor	网格颜色		categoryAxis.gridColor = "#FFFFFF" ;
11	dateFormats	日期格式，将数据格式化对应的日期格式		categoryAxis.dateFormats = [{ period:'DD', format: 'DD'}, {period:'WW', format: 'MMM DD'}, {period: 'MM',format:'MMM'}, period: 'YYYY', format: 'YYYY' }];
12				

3、Legend（图例）

序号	属性名 / 方法名	作用	对象获取方法 / 常用属性值	示例
----	-----------	----	----------------	----

1	legend 对象	在图表的上方或者下方显示图例，图例的颜色就是对应线条的颜色	var legend = new AmCharts.AmLegend();	
2	align	排列样式	center	legend.align = "center" ;
3	marginLeft	左边缘		legend.marginLeft = 0;
4	title	标题		legend.title= " 测试 " ;
5	horizontalGap	水平间隔，一个图表可以有多个图例，图例之间的间隔用此属性		legend.horizontalGap = 10;
6	equalWidths	是否等宽		legend.equalWidths = false ;
7	valueWidth	值的宽度，在图例的右侧会显示该线或者图表的当前选中的值，设置为 0 时则不显示值		legend.valueWidth = 120;
8	switchType	暂时没明白什么意思		legend.switchType = "v" ;

4、Guide（向导线）

序号	属性名 / 方法名	作用	对象获取方法 / 常用属性值	示例
1	guide 对象	向导线可以是一条跟 Y 轴平行的线，也可以是一个矩形区域	var guide = new AmCharts.Guide();	
2	fillAlpha	区域透明度		guide.fillAlpha = 0.1;
3	lineAlpha	线透明度		guide.lineAlpha = 0;
4	value	其实值，其实指对应 Y 坐标的值		guide.value = 50;
5	toValue	到达值，其实指对应 Y 坐标的值，跟上面属性共同确定了一个从 value 到 toValue 的区域，宽度为图表的宽度，高度为（toValue-value）的绝对值		guide.toValue = 0;
6	lineColor	相导线的颜色		guide.lineColor =

				"#CC0000" ;
7	dashLength	破折长度，默认为 0 为实心线条，设置值后为破折线		guide.dashLength = 4;
8	label	标签，就是给向导线显示一个名字		guide.label = " 平均值 " ;
9	inside	是否让向导线显示在图形里面，默认为 true	True , false	guide.inside = true ;

5、 Scrollbar （滚动条）

序号	属性名 / 方法名	作用	对象获取方法 / 常用属性值	示例
1	scrollbar 对象	滚动条可以选择图表显示的区域	var chartScrollbar = new AmCharts.ChartScrollbar();	
3	backgroundAlpha	滚动条背景透明度		chartScrollbar.backgroundAlpha = 0.1 ;
4	backgroundColor	滚动条背景颜色		chartScrollbar.backgroundColor = "#000000" ;
5	graphLineAlpha	图像线条的透明度		chartScrollbar.graphLineAlpha = 0.1 ;
6	graphFillAlpha	图像的填充透明度		chartScrollbar.graphFillAlpha = 0 ;
7	selectedGraphFillAlpha	选中图像的填充色的透明度		chartScrollbar.selectedGraphFillAlpha = 0 ;
8	selectedGraphLineAlpha	选中区域的图像线条透明度		chartScrollbar.selectedGraphLineAlpha = 0.25 ;
9	scrollbarHeight	滚动条高度		chartScrollbar.scrollbarHeight = 30 ;
10	selectedBackgr	选中区域的背景颜色		chartScrollbar.selectedBa

	oundColor			ckgroundColor = "#FFFFFF" ;

## 6、 Graph ( 图表 )

序号	属性名 / 方法名	作用	对象获取方法 / 常用属性值	示例
1	graph 对象	图像对象，必须有该属性	var graph1 = new AmCharts.AmGraph();	
2	valueAxis	图像的 Y 轴，一个 chart 可以添加多个 graph，一个 graph 只能有一个 valueAxis		graph1.valueAxis = valueAxis1;
3	valueField	指定一个字段作为 Y 坐标值		graph1.valueField = "visits" ;
4	bullet	图像的节点类型		graph1.bullet = "round" ;
5	dashLength	绘制图像时延时，默认为 0 秒，设置为正整数时可以看到动态生成效果		graph1.dashLength = 0;
6	hideBulletsCount	一个图像中当节点大于一定数值后隐藏节点形状		graph1.hideBulletsCount = 10;
7	balloonText	节点显示的文本内容		graph1.balloonText = "[[date]] ([[visits]])" ;
8	lineColor	图像线颜色		graph1.lineColor = "#d1655d" ;
9	connect	是否连接起来，是指如果数据有 x 轴值，但是 y 轴值丢失的时候，如果设置为 true 则忽略该点，设置为 false 则线条在此点处断开		graph1.connect = false ;
10	bulletBorderColor	节点边框颜色		graph1.bulletBorderColor = "#FFFFFF" ;
11	bulletBorderThickness	节点边框宽度		graph1.bulletBorderThickness = 2;
12	customBulletField	用户自定义节点字段		graph.customBulletField =

	eld			"bullet";
13	bulletOffset	节点偏移量		graph.bulletOffset = 16;
14	cornerRadiusTop			graph.cornerRadiusTop = 8 ;
	p			
15	bulletSize	节点大小		graph.bulletSize = 14;
16	colorField	颜色字段，颜色可以从数据中读取		graph1.colorField = "color" ;
17	type	图像类型，有 line 、column、smoothedLine 类型，第一种为线形图，第二种为柱状图	line /column/smoothedLine	graph1.type = "line";
18	fillAlphas	填充区透明度，默认为 0，最大值为 1，当设置值时，在线条跟 x 轴之间的区域会填充颜色		graph1.fillAlphas = 0.3;
19	negativeLineColor	当数值为负数时线条的颜色		graph1.negativeLineColor = "#efcc26" ;

7、Chart （amcharts 对象）

序号	属性名 / 方法名	作用	对象获取方法 / 常用属性值	示例
1	chart 对象	Amcharts 的核心对象	var chart = new AmCharts.AmSerialChart();	
2	pathToImages	指定 chart 图片的引用地址		chart.pathToImages = "amcharts/images/" ;
3	zoomOutButton	设置放大 / 缩小按钮的背景色和透明度		chart.zoomOutButton = {  backgroundColor: '#000000' ,



				backgroundAlpha: 0.15 };
4	dataProvider	指定数据来源，一般指向一个数组对象		chart.dataProvider = chartData;
5	categoryField	指定 X 轴由哪个字段决定		chart.categoryField = "date" ;
6	autoMargins	自动调整边距，如果设置为 true 则边距设置不起效		chart.autoMargins = true ;
7	fontSize	字体大小		chart.fontSize = 14;
8	color	图标颜色		chart.color = "#FFFFFF" ;
9	marginTop	上边距		chart.marginTop = 100 ;
10	marginLeft	左边距		chart.marginLeft = 50 ;
11	marginRight	右边距		chart.marginRight = 30 ;
12	addGraph(graph)	添加一个图形，可以添加多个，想要绘制图形，必须有此方法		chart.addGraph(graph1);
13	validateNow ( div )	当数据改变时或者属性改变时，想要重新绘图，可以调用该方法		chart.validateNow('chartd iv') ;
14	chart.write( 'chartd iv' );	将 amcharts 对象写到一个 div 中，最常用方法		chart.chart.write( 'chartd iv' );;
15	addListener('dataUpdated', zoomChart)	添加一个监听函数，第一个参数是指定事件，第二个是调用的函数名	chart.addListener('zoomed', handleZoom) ;	chart.addListener('dataUpdated', zoomChart) ;
16	rotate	图像是否 xy 轴互换，默认为 false，如果想让图像顺时针旋转 90 °，则设置为 true	False , true	chart.rotate = false ;
17	depth3D	设置为 3d 图像的厚度值		chart.depth3D = 50
18	angle	角度，当设置图像为 3d 图时使用该属性，默认为 0		chart.angle = 40



19	startDuration			chart.startDuration = 2
20	plotAreaBorderColor	绘图区域边框颜色		chart.plotAreaBorderColor = "#000000" ;
21	plotAreaBorderAlpha	绘图区域边框透明度		chart.plotAreaBorderAlpha = 5;
22	backgroundImage	设置背景图片的地址		chart.backgroundImage = "amcharts/images/bg.jpg" ;
23	addChartScrollbar(chartScrollbar)	添加滚动条，只能添加一个		chart.addChartScrollbar(chartScrollbar) ;
24	addLegend(legend)	添加图例，可以添加多个		chart.addLegend(legend);
25	addValueAxis(valueAxis1)	添加 Y 轴。可以添加多个		chart.addValueAxis(valueAxis1);
26	addChartCursor(chartCursor)	添加鼠标形状		chart.addChartCursor(chartCursor);
27				
28				
29				
30				
31				
32				

8、 ChartCursor （光标）

序号	属性名 / 方法名	作用	对象获取方法 / 常用属性值	示例
----	-----------	----	----------------	----

1	chartCursor 象	对 设置光标的形状和样式	var chartCursor = new AmCharts.ChartCursor();	
2	zoomable	是否可以缩放，设为 true 的时候，按住鼠标 左键划线可以方法图像	True/false	chartCursor.zoomable = false ;
3	cursorAlpha	光标透明度		chartCursor.cursorAlpha = 0;
4	cursorPosition	光标位置		chartCursor.cursorPositio n = "mouse" ;
5	pan	默认为 false ，设置为 true 时，光标变为四个 向外的箭头形状，按住左键滑动鼠标可以将图 像向左移动向右移动		chartCursor.pan = true ;
6	categoryBalloonDateFormat	设置光标节点显示的日期格式		chartCursor.categoryBalloonDateFormat = "JJ:NN, DD MMMM";

9、 动态图表示例

1、需要在 html 页面增加一个 div ，同时设置 div 的 id 和样式 ， amcharts 将图表显示在指定的 id 的 div 中

```
<div id="chartdiv" style="width: 50%; height: 300px"></div>
```

2、引用 amcharts js 库和 css 样式

```
<script src = "amcharts/amcharts.js" type = "text/javascript" ></ script >
```

```
<link rel = "stylesheet" href = "style.css" type = "text/css" >
```

### 3、 设定一个定时器，循环调用函数

```
window.setInterval(show,2000);//      每隔 2 秒调用一次 show ( ) 方法， show 方法完成绘图功能
function show(){
    var chart =      new AmCharts.AmSerialChart();
    var valueAxis1 =      new AmCharts.ValueAxis();
    var graph1 =      new AmCharts.AmGraph();
    var categoryAxis = chart.categoryAxis;
    var guide =      new AmCharts.Guide();
    var legend =      new AmCharts.AmLegend();
    var chartCursor =      new AmCharts.ChartCursor();

    // 设定最大显示数值个 数
    generateChartData();
    chart.pathToImages =      "amcharts/images/"      ;
    chart.zoomOutButton = {
        backgroundColor:      '#000000'      ,
        backgroundAlpha: 0.15
    };
    chart.dataProvider = chartData;
    chart.categoryField =      "date"      ;
    categoryAxis.parseDates =      false      ; // as our data is date-based, we set parseDates to tru      e
        categoryAxis.dashLength = 1;
    categoryAxis.gridAlpha = 0.15;
    categoryAxis.axisColor =      "#DADADA"      ;
        // categoryAxis.position = "top"      ;
```

```

categoryAxis.gridPosition = "start" ;
categoryAxis.startOnAxis = true ;
categoryAxis.gridColor = "#FFFFFF" ;
// CURSOR R

chartCursor.zoomable = false ; // as the chart displays not too many values, we disabled zoomin g
chartCursor.cursorAlpha = 0;
chartCursor.cursorPosition = "mouse" ;
chartCursor.pan = true ; // set it to fals if you want the cursor to work in "select" mod e
chart.addChartCursor(chartCursor);

valueAxis1.axisColor = "#FF6600" ;
valueAxis1.axisThickness = 1;
valueAxis1.gridAlpha = 0;
valueAxis1.tickLength =0;
valueAxis1.minimum = - 100;
valueAxis1.maximum = 100;
valueAxis1.title= " 哈哈 " ;
valueAxis1.logarithmic = false ; // this line makes axis logarithmi c
valueAxis1.integersOnly = true ;
valueAxis1.gridCount = 10;
valueAxis1.unit = "%" ;
valueAxis1.labelsEnabled = true ;
valueAxis1.inside = true ;
valueAxis1.position = "left" ;

```

```
chart.addValueAxis(valueAxis1);

// LEGEND
legend.align = "center" ;
legend.marginLeft = 0;
legend.title= " 测试 " ;
legend.horizontalGap = 10;
legend.equalWidths = false ;
legend.valueWidth = 120;
chart.addLegend(legend);

guide.fillAlpha = 0.1;
guide.lineAlpha = 0;
guide.value = 50;
guide.toValue = 0;
guide.lineColor = "#CC0000" ;
guide.dashLength = 4;
guide.label = " 平均值 " ;
guide.inside = true ;
guide.lineAlpha = 1;

var guide1 = new AmCharts.Guide();
guide1.lineColor = "#CC0000" ;
guide1.lineAlpha = 1;
guide1.dashLength = 2;
guide1.inside = true ;
```

```

guide1.labelRotation = 90;
categoryAxis.addGuide(guide1);
valueAxis1.addGuide(guide);

graph1.valueAxis = valueAxis1;           // we have to indicate which value axis should be use      d
graph1.title = "      红色 " ;
graph1.valueField = "visits" ;
graph1.bullet = "round" ;
graph1.dashLength = 0;
graph1.hideBulletsCount = 10;
graph1.balloonText = "[[date]] ([[visits]])" ;
graph1.lineColor = "#d1655d" ;
graph1.connect = false ;
graph1.negativeLineColor = "#efcc26" ;
graph1.bulletBorderColor = "#FFFFFF" ;
graph1.bulletBorderThickness = 2;
graph1.type = "smoothedLine" ; // this line makes the graph smoothed line .
graph1.fillAlphas = 0.3; // setting fillAlphas to > 0 value makes it area grap      h
chart.addGraph(graph1);

chart.addTitle(" 测试动态报表图 ", 5);
chart.plotAreaBorderColor = "#000000" ;
chart.plotAreaBorderAlpha = 5;
chart.autoMargins = true ;
chart.fontSize = 14;
chart.write('chartdiv' );

```

```
        chart=            null    ;
        valueAxis1 =      null    ;
        graph1 =          null    ;
        categoryAxis =    null    ;
        guide =           null    ;
        legend =          null    ;
    }

function generateChartData() {
    $.ajax({
        type :    "get"    ,
        url :     "${pageContext.request.contextPath}/getData"    ,
        dataType :    "text"    ,
        data : {
        },
        success :    function    (result) {
            parseData(result,20);
        },
        error :      function    (XMLHttpRequest, textStatus, errorThrown) {
            alert("    请求异常，请检查服务器是否正常运行！    " + XMLHttpRequest.status + "
                + XMLHttpRequest.readyState + " " + textStatus)    ;
        }
    });
}
```



```
function parseData(data,num){
    var tempDate = [];
    tempDate = data.split(" ");
    chartData.push({
    date:tempDate[0],
    visits:tempDate[1]
    });
    var newChartData=[];
    var len=chartData.length;
    if (len>num){
        for ( var m=0;m <num;m++){
            newChartData[m]=chartData[len - num+m];
        }
        chartData = newChartData;
        newChartData= null ;
    }
    len= null ;
    visits= null ;
}
```

#### 4、写一个 servlet 程序，给 amcharts 提供动态数据

```
public class DataProviderServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    public static Integer i =0;
    public DataProviderServlet() {
```

```
        super ();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
        StringBuilder sb = new StringBuilder();
        sb.append( i ++).append( " " ).append( new Random().nextInt(100)*( i %2==0?(-1):(1)));
        System. out .println(sb.toString());
        response.getWriter().write(sb.toString());

    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
        doGet(request, response);
    }
}
```

## 5、效果截图

测试动态报表图

