

Git&黑马就业数据平台



黑马程序员
www.itheima.com

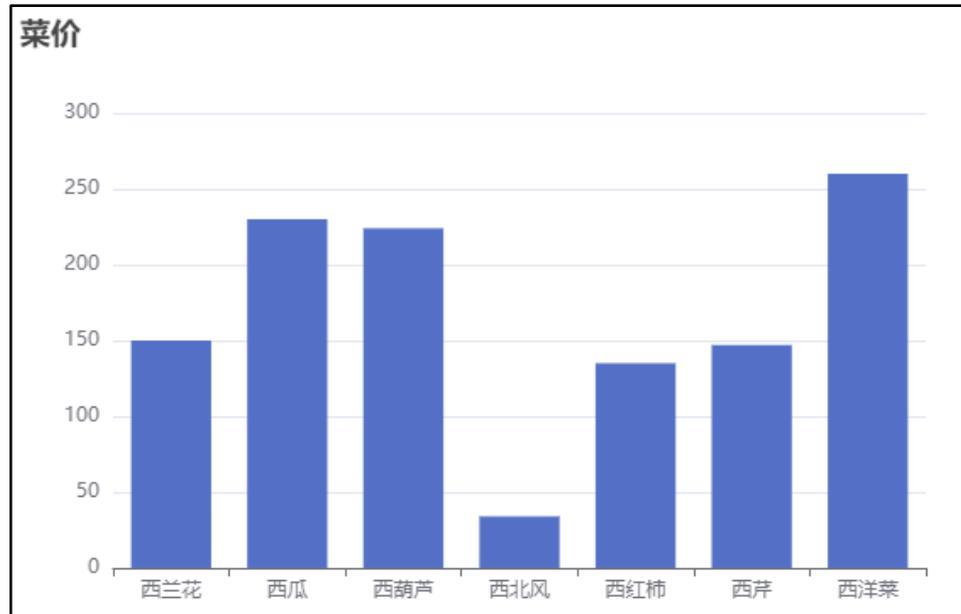
传智教育旗下
高端IT教育品牌

数据可视化

数据可视化: 主要旨在借助于图形化手段，清晰有效地传达于沟通信息

作用: 将数据转换为图形，数据特点更加突出

```
const foodArr = [  
  { name: '西兰花', price: 150 },  
  { name: '西瓜', price: 230 },  
  { name: '西葫芦', price: 224 },  
  { name: '西北风', price: 34 },  
  { name: '西红柿', price: 135 },  
  { name: '西芹', price: 147 },  
  { name: '西洋菜', price: 260 },  
]
```



echarts简介

[官网](#): 一个基于 JavaScript 的开源可视化图表库

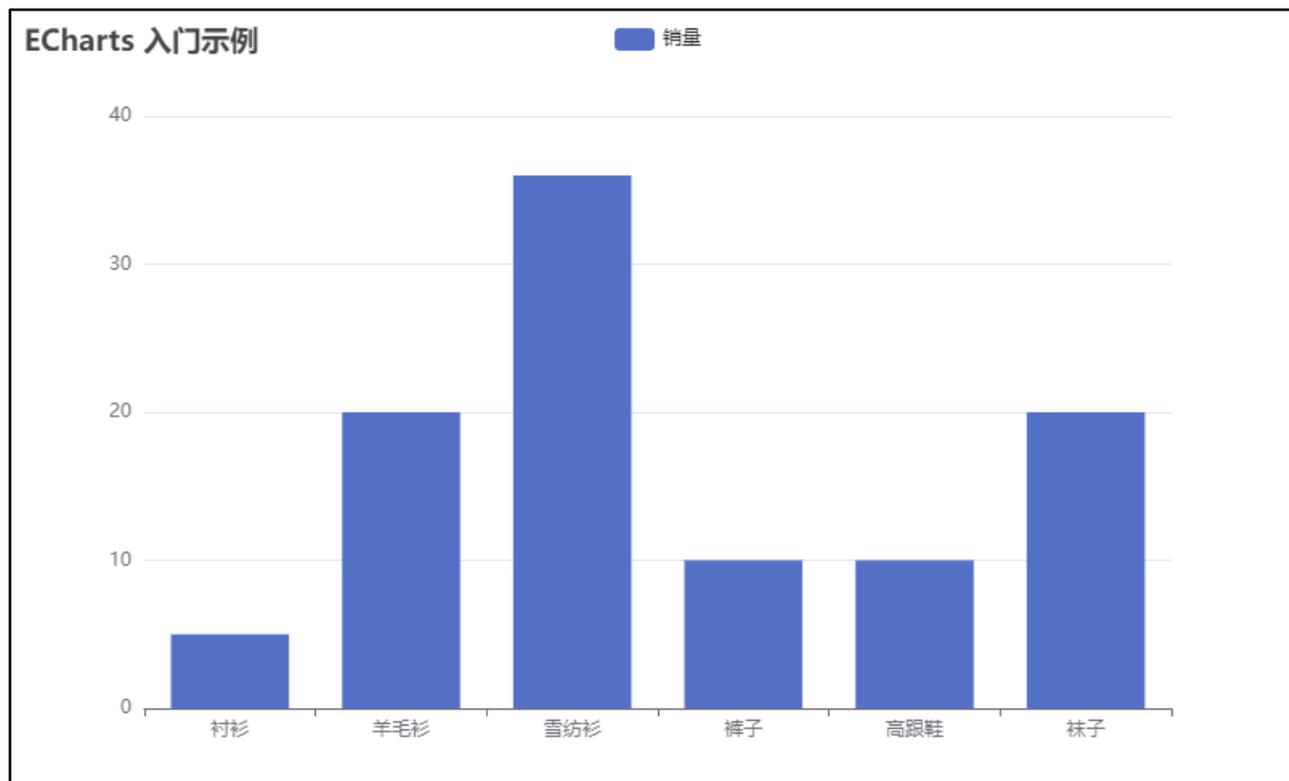
特点:

1. 性能好，流畅运行于 PC 和 移动端
2. 兼容主流浏览器
3. 提供非常多的常用图表，且支持定制



echarts快速上手

需求: 基于[文档](#)生成基础图表



总结

1. 数据可视化:

将数据转换为图形，数据特点更加突出

2. 日常开发中如何绘制图表:

借助图表库完成绘制，比如ECharts

3. ECharts使用准备

```
<!-- 准备工作: 准备定义了宽高的dom容器 -->
<div id="main" style="width: 600px;height:400px;"></div>
<!-- 准备工作: 下包+导包 -->
<script src="../lib/echarts.min.js"></script>
```

4. ECharts核心使用步骤

```
// 基于准备好的dom, 初始化echarts实例
const myChart = echarts.init(dom元素)

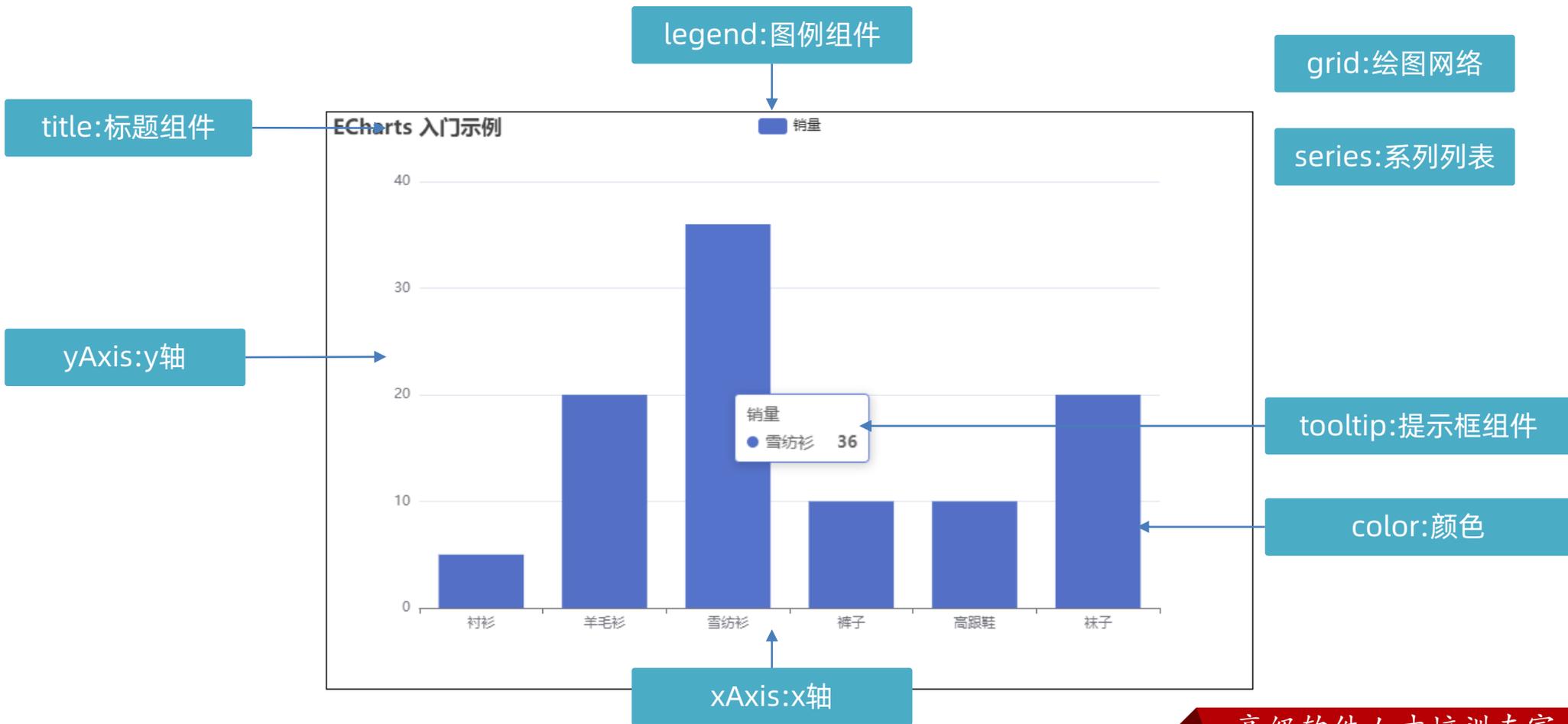
// 指定图表的配置项和数据
const option = {
  //....
}

// 使用刚指定的配置项和数据显示图表。
myChart.setOption(option)
```

echarts基础配置

通过不同配置项的组合，即可实现不同的图表

```
const option = {  
  // 标题  
  title: {},  
  // 图例  
  legend: {},  
  // 绘图网络  
  grid: {},  
  // x轴  
  xAxis: {},  
  // y轴  
  yAxis: {},  
  // 提示框  
  tooltip: {},  
  // 系列图表  
  series: [],  
  // 颜色  
  color: []  
}
```



echarts基础配置

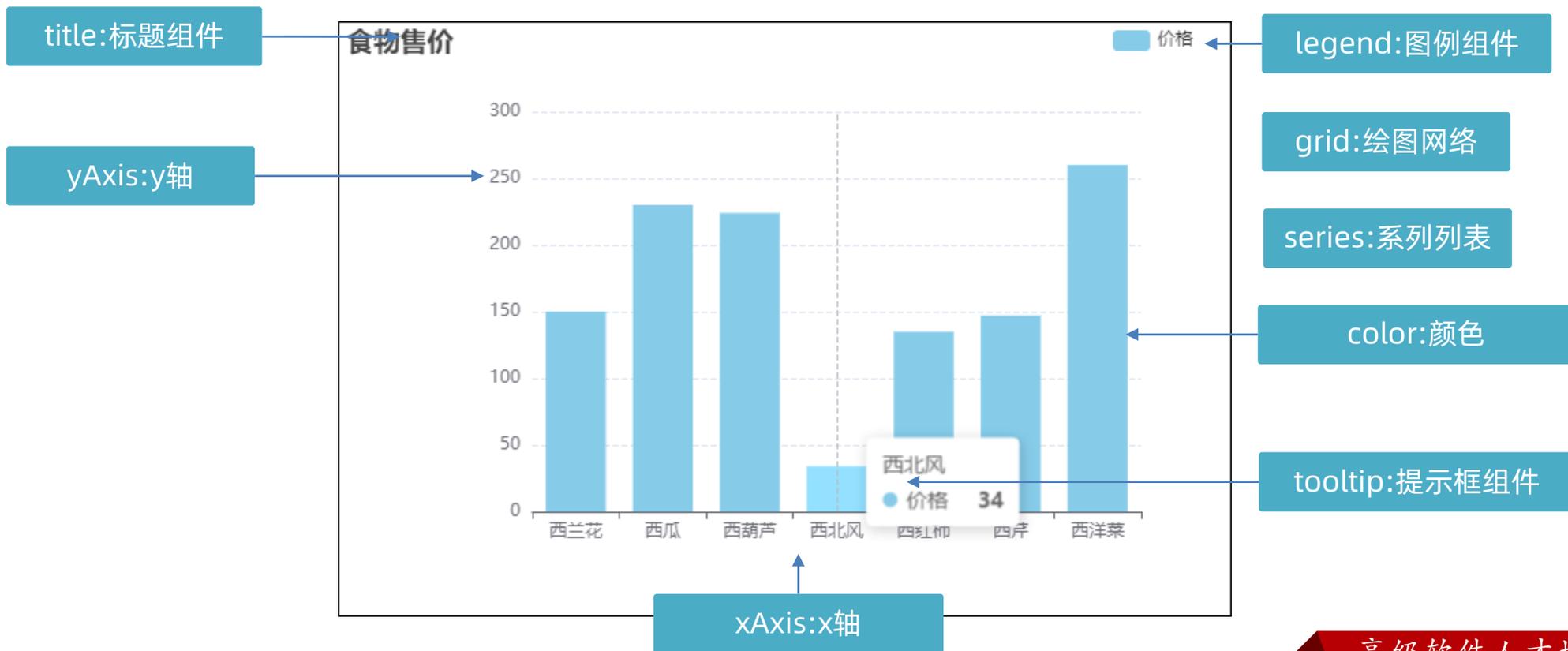
需求: 将上一节的图形, 调整为如下外观

核心步骤:

查文档



调整设置



echarts基础配置

需求: 将上一节的图形, 调整为如下外观

核心步骤:

查文档



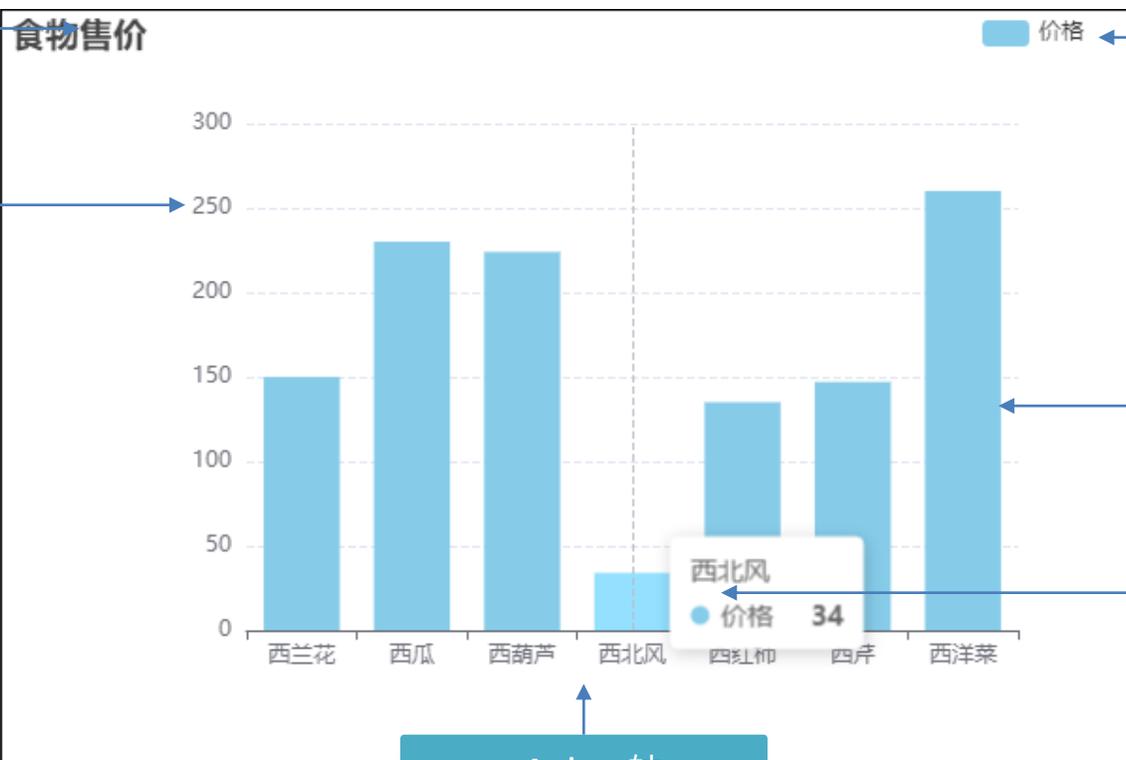
调整设置

```
const option = {  
  title: {  
    text: ''  
  },  
  legend: {  
    data:[],  
    right: '',  
  },  
}
```

title:标题组件

yAxis:y轴

食物售价



legend:图例组件

grid:绘图网络

series:系列列表

color:颜色

tooltip:提示框组件

xAxis:x轴

echarts基础配置

需求: 将上一节的图形, 调整为如下外观

核心步骤:

查文档

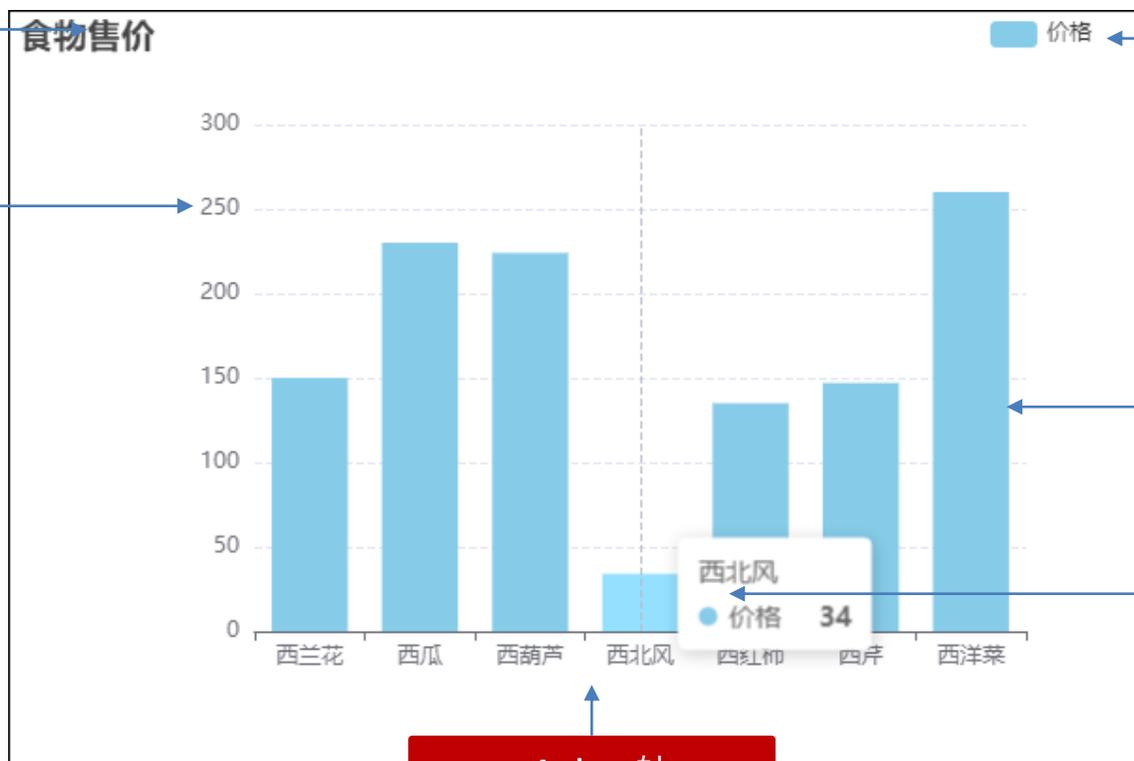
调整设置

```
const option = {  
  grid:{  
    left:''  
  },  
  xAxis:{  
    data:[]  
  }  
}
```

title:标题组件

yAxis:y轴

食物售价



legend:图例组件

grid:绘图网络

series:系列列表

color:颜色

tooltip:提示框组件

xAxis:x轴

echarts基础配置

需求: 将上一节的图形, 调整为如下外观

核心步骤:

查文档

调整设置

```
const option = {  
  yAxis: {  
    splitLine: {  
      lineStyle: {  
        type: ''  
      }  
    }  
  },  
  tooltip: {  
    trigger: ''  
  }  
}
```

title:标题组件

yAxis:y轴

食物售价

价格

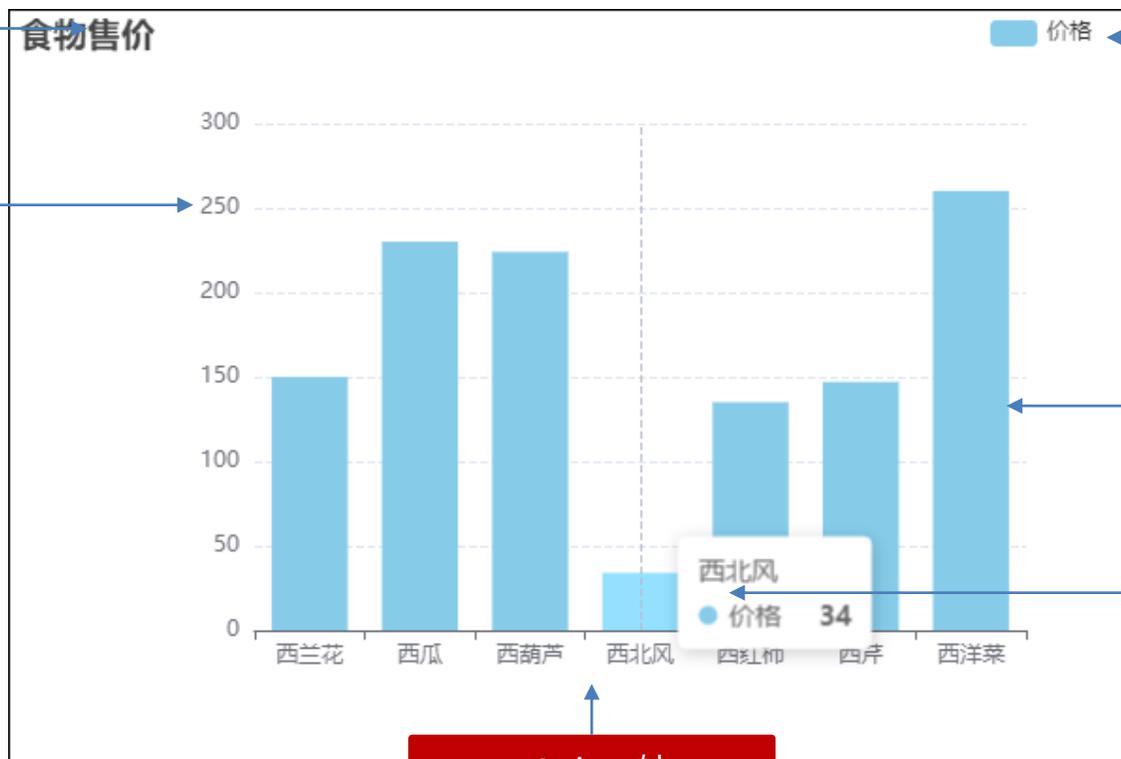
legend:图例组件

grid:绘图网络

series:系列列表

color:颜色

tooltip:提示框组件



xAxis:x轴

echarts基础配置

需求: 将上一节的图形, 调整为如下外观

核心步骤:

查文档

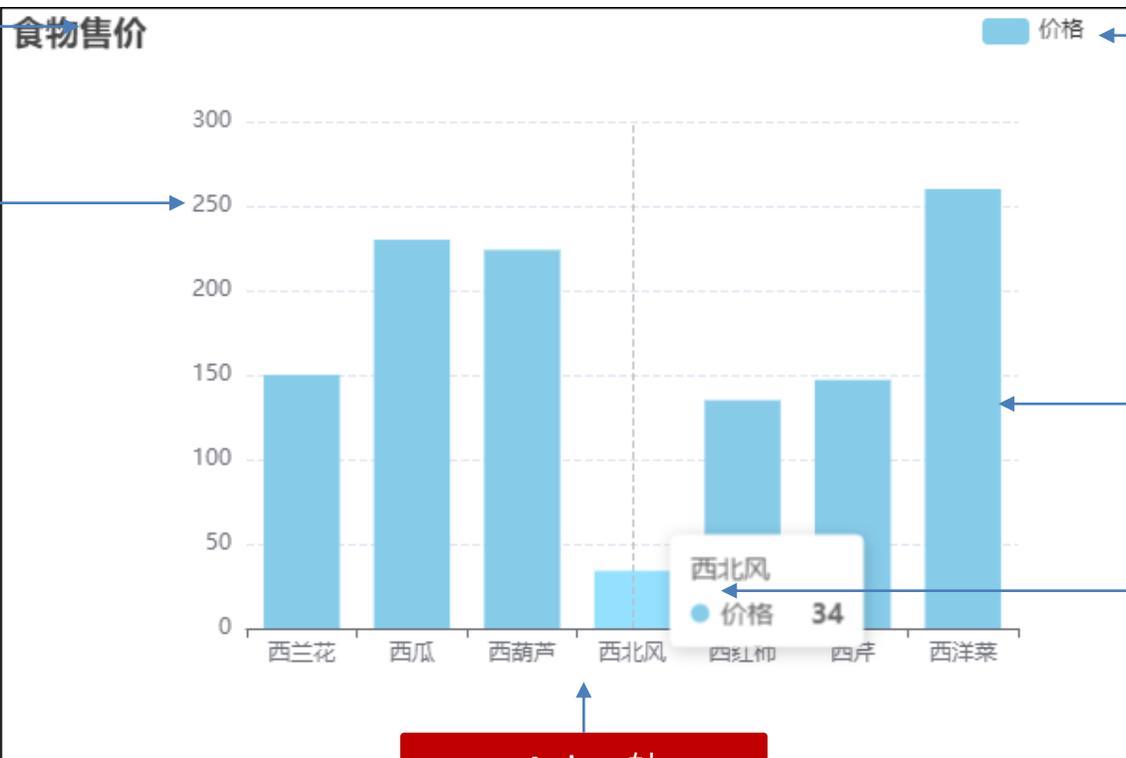
调整设置

```
const option = {  
  series: [  
    {  
      name: '',  
      type: '',  
      data: []  
    }  
  ],  
  color: []  
}
```

title:标题组件

yAxis:y轴

食物售价



legend:图例组件

grid:绘图网络

series:系列列表

color:颜色

tooltip:提示框组件

xAxis:x轴

数据看板-函数抽取

核心步骤:

抽取函数



调用并传递数据

```
async function getData() {  
  const res = await axios({  
    url: '/dashboard',  
  })  
  // 渲染顶部数据...  
}
```

```
{  
  message: '获取面板数据成功', data: {...}  
  data: {  
    groupData: {1: Array(10), 2: Array(1)},  
    overview: {salary: 16276, student_co},  
    provinceData: (29) [{...}, {...}, {...}, {...}],  
    salaryData: (4) [{...}, {...}, {...}, {...}],  
    year: (12) [{...}, {...}, {...}, {...}, {...}],  
    [[Prototype]]: Object  
  }  
  message: "获取面板数据成功"  
}
```



数据看板-薪资走势

核心步骤:

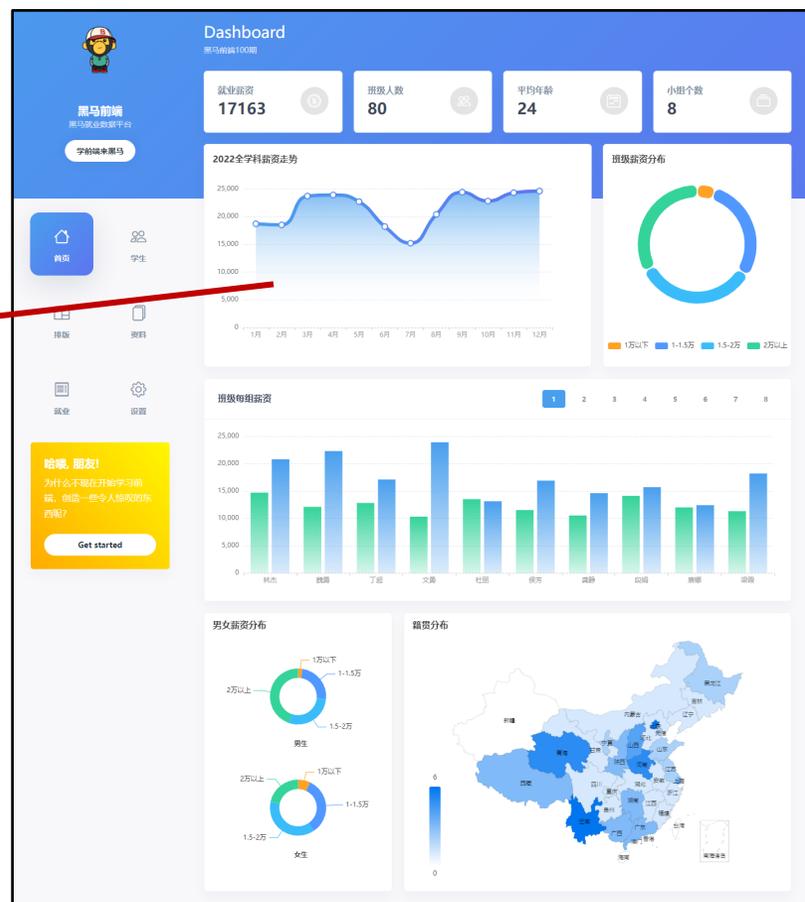
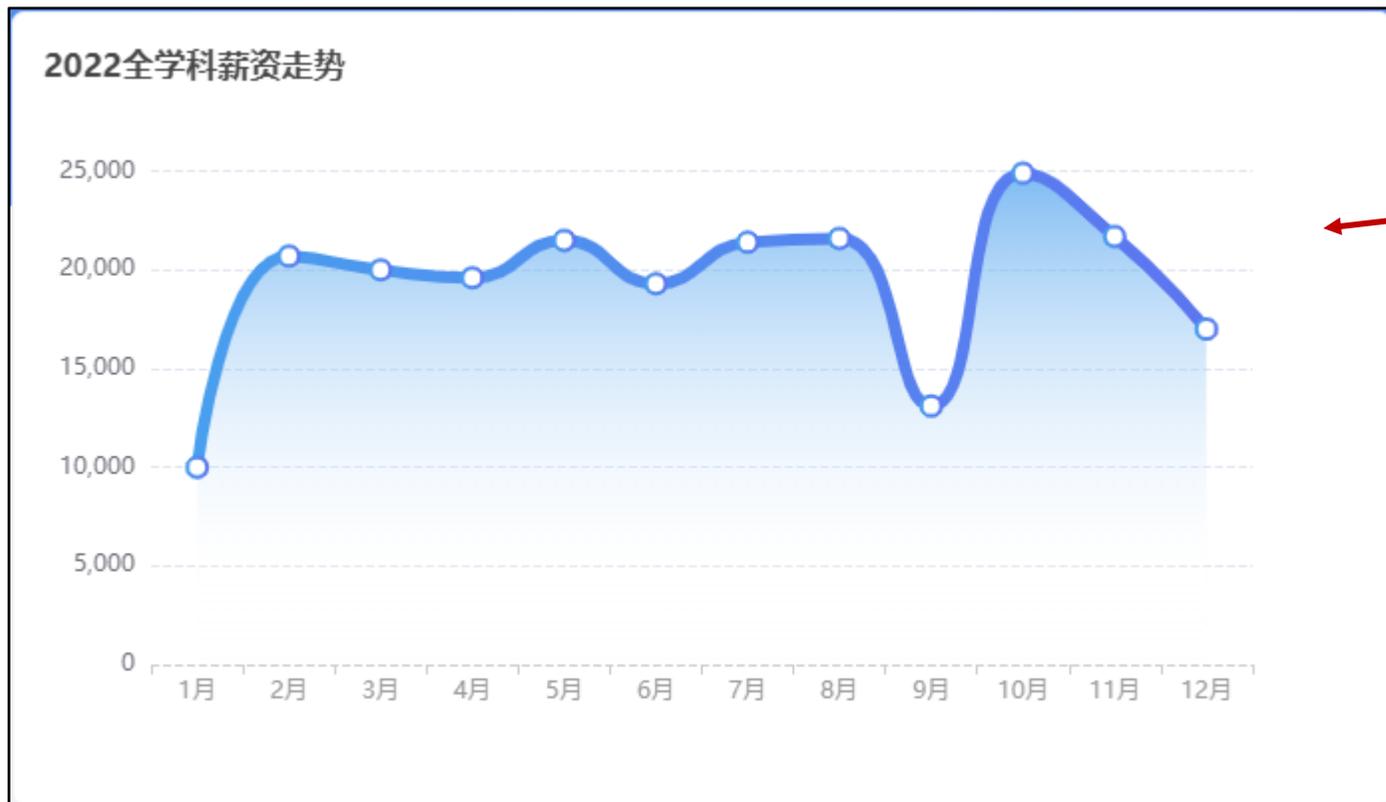
找示例



整合到项目



调整设置



数据看板-薪资走势

核心步骤:

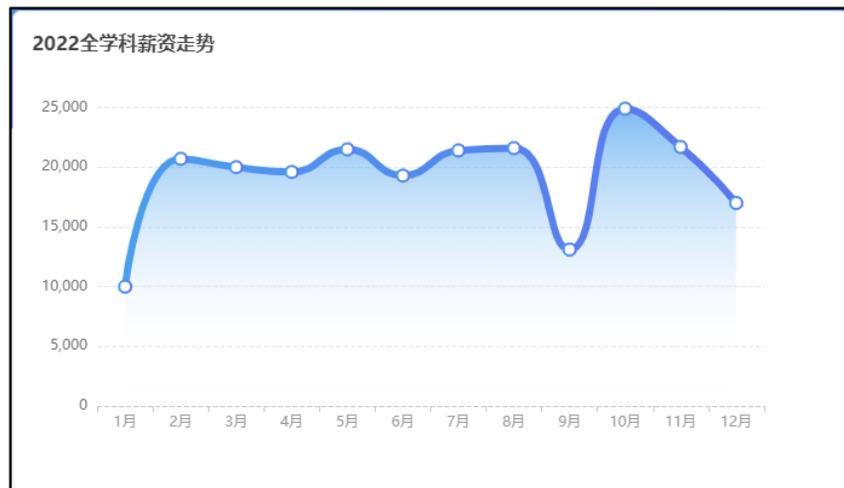
找示例



整合到项目



调整设置



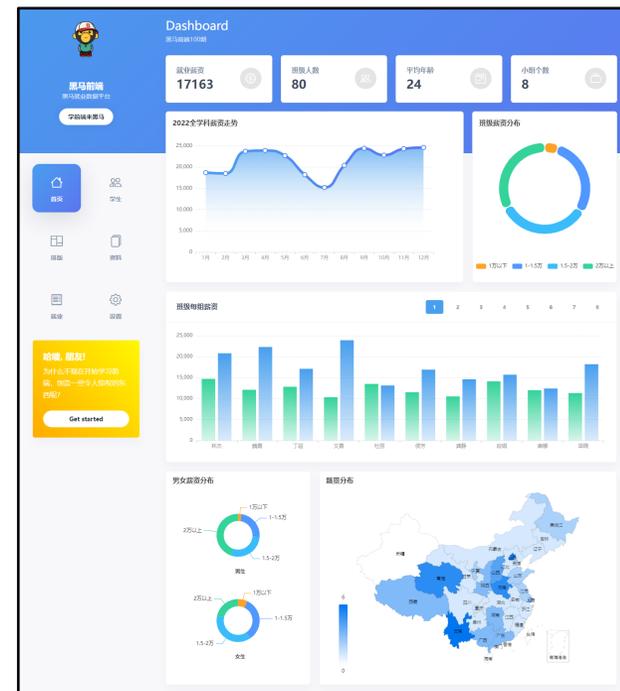
抽取函数



调用并传递数据



整合示例



数据看板-薪资走势

核心步骤:

找示例

整合到项目

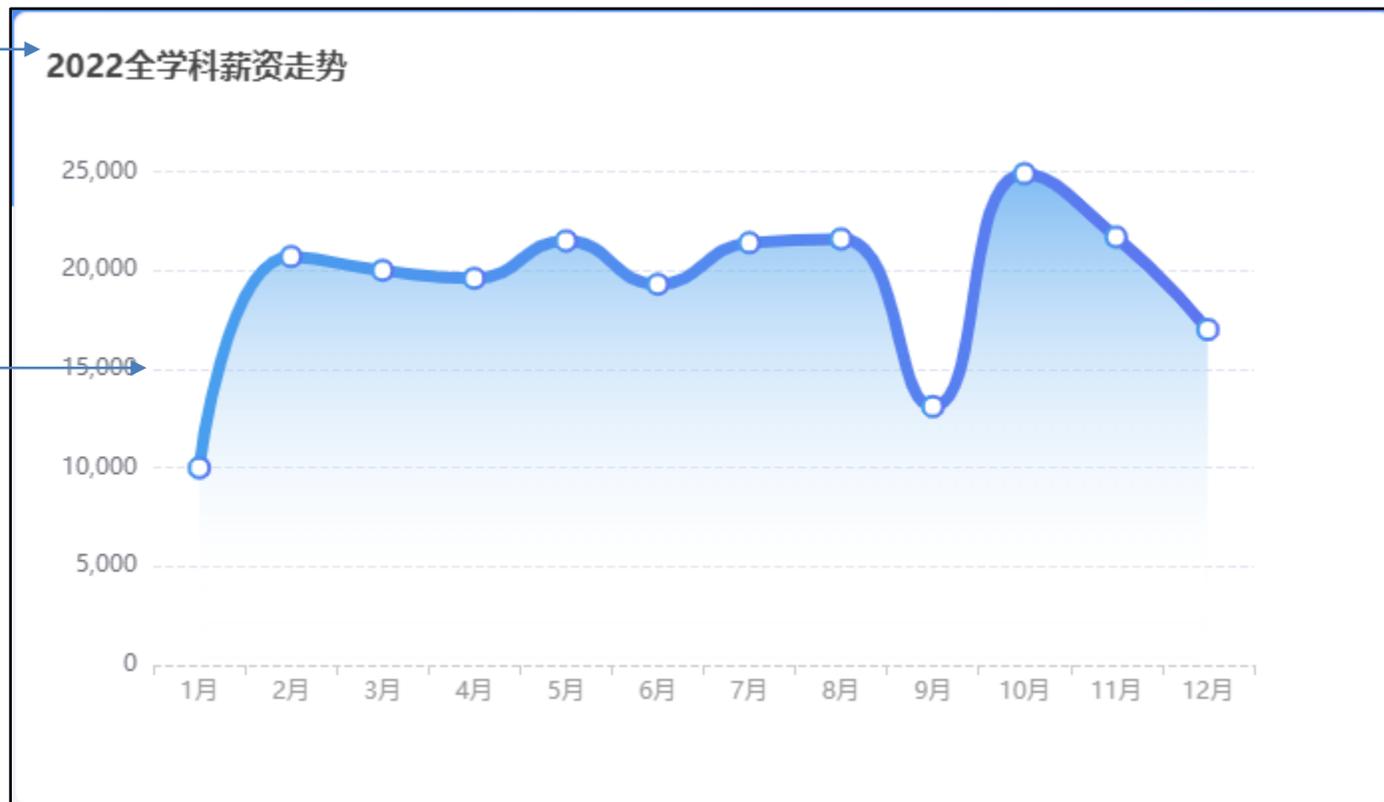
调整设置

```
const option = {  
  title: {  
    text: '',  
    left: '',  
    top: ''  
  },  
  grid: {  
    top: ''  
  },  
  yAxis: {  
    splitLine: {  
     LineStyle: {  
        type: ''  
      }  
    },  
  },  
}
```

title:标题组件

grid:绘图网络

yAxis:y轴



数据看板-薪资走势

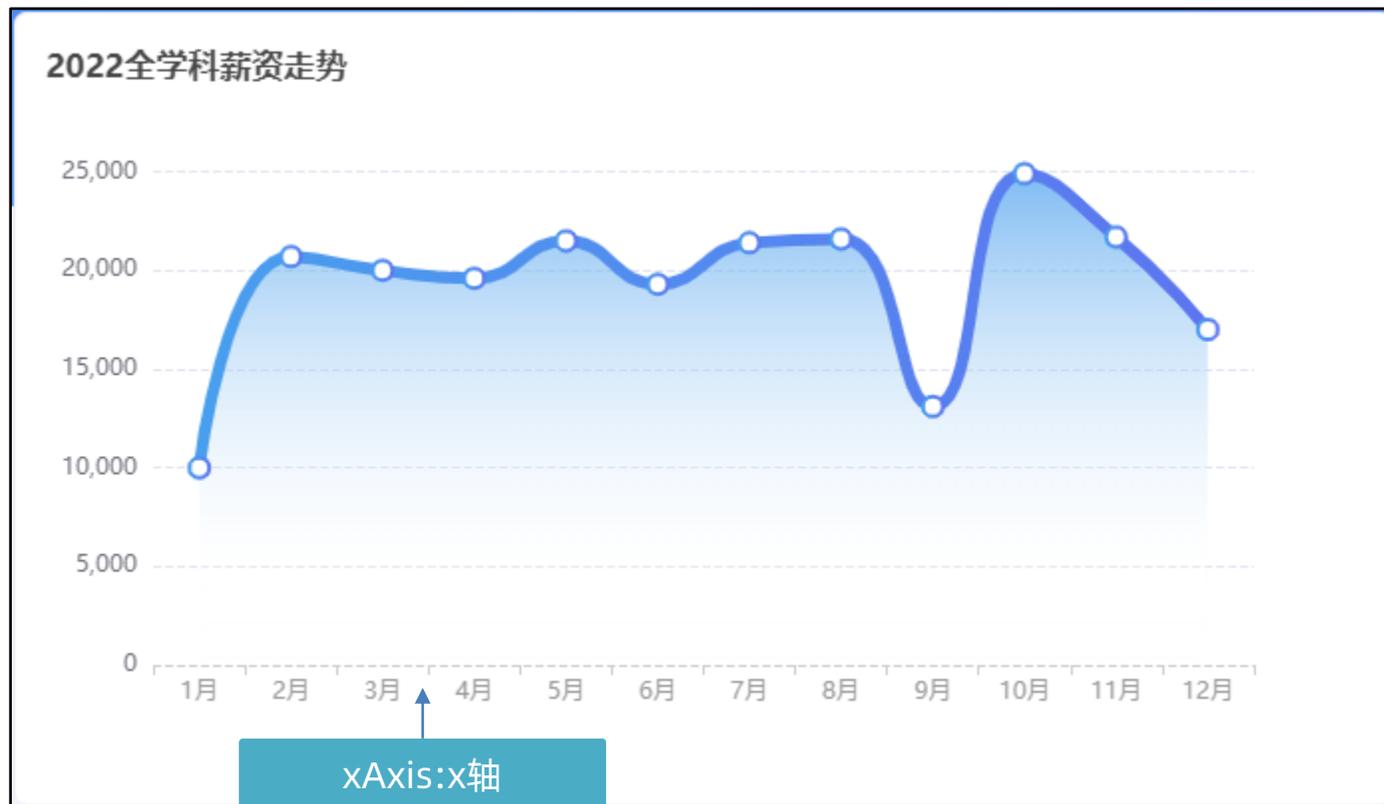
核心步骤:

找示例

整合到项目

调整设置

```
const option = {  
  xAxis: {  
    axisLine: {  
     LineStyle: {  
        color: '',  
        type: ''  
      }  
    },  
    data: []  
  },  
}
```



数据看板-薪资走势

核心步骤:

找示例



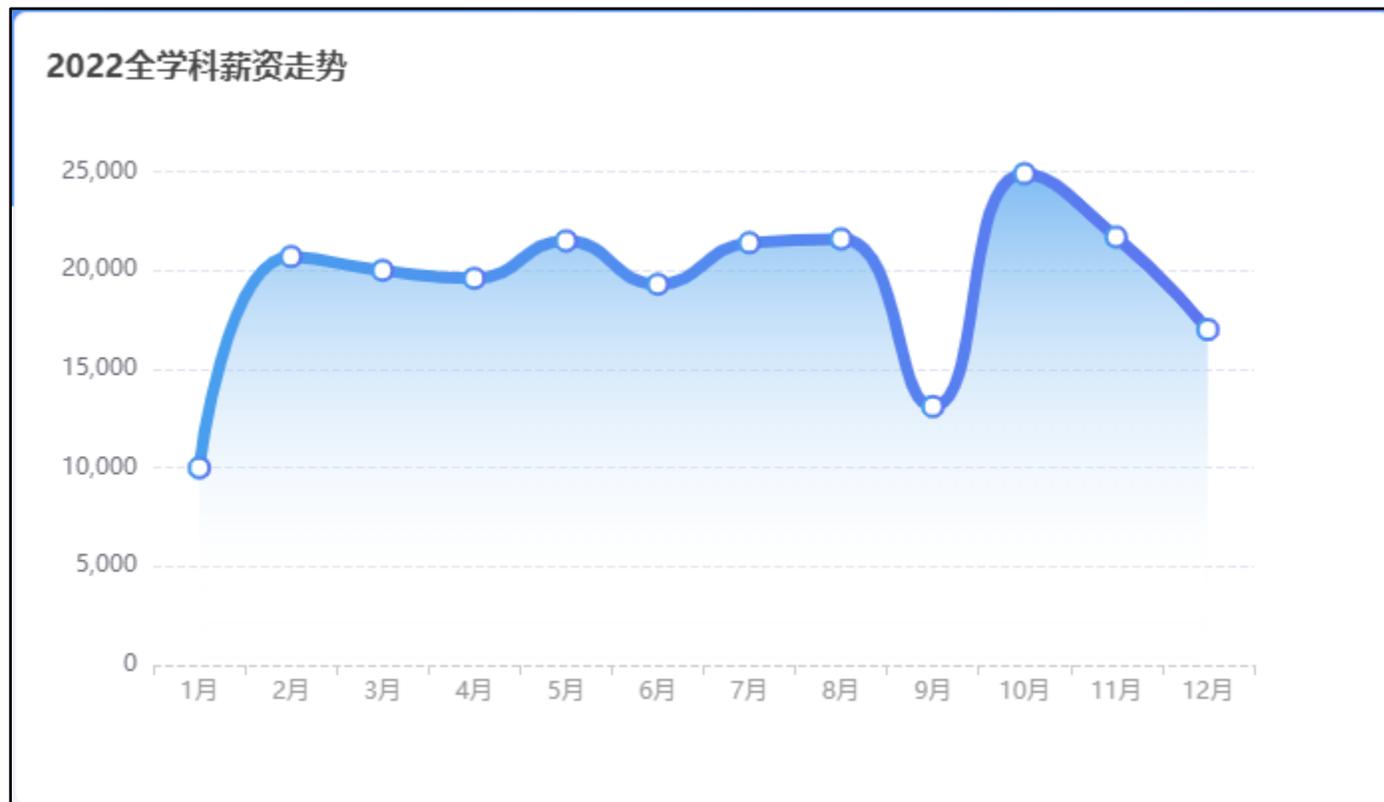
整合到项目



调整设置

```
const option = {  
  series: [  
    {  
      data: [],  
      symbolSize: '',  
      lineStyle: {  
        width: '',  
      }  
    }  
  ]  
}
```

series:系列列表



数据看板-薪资走势

核心步骤:

找示例



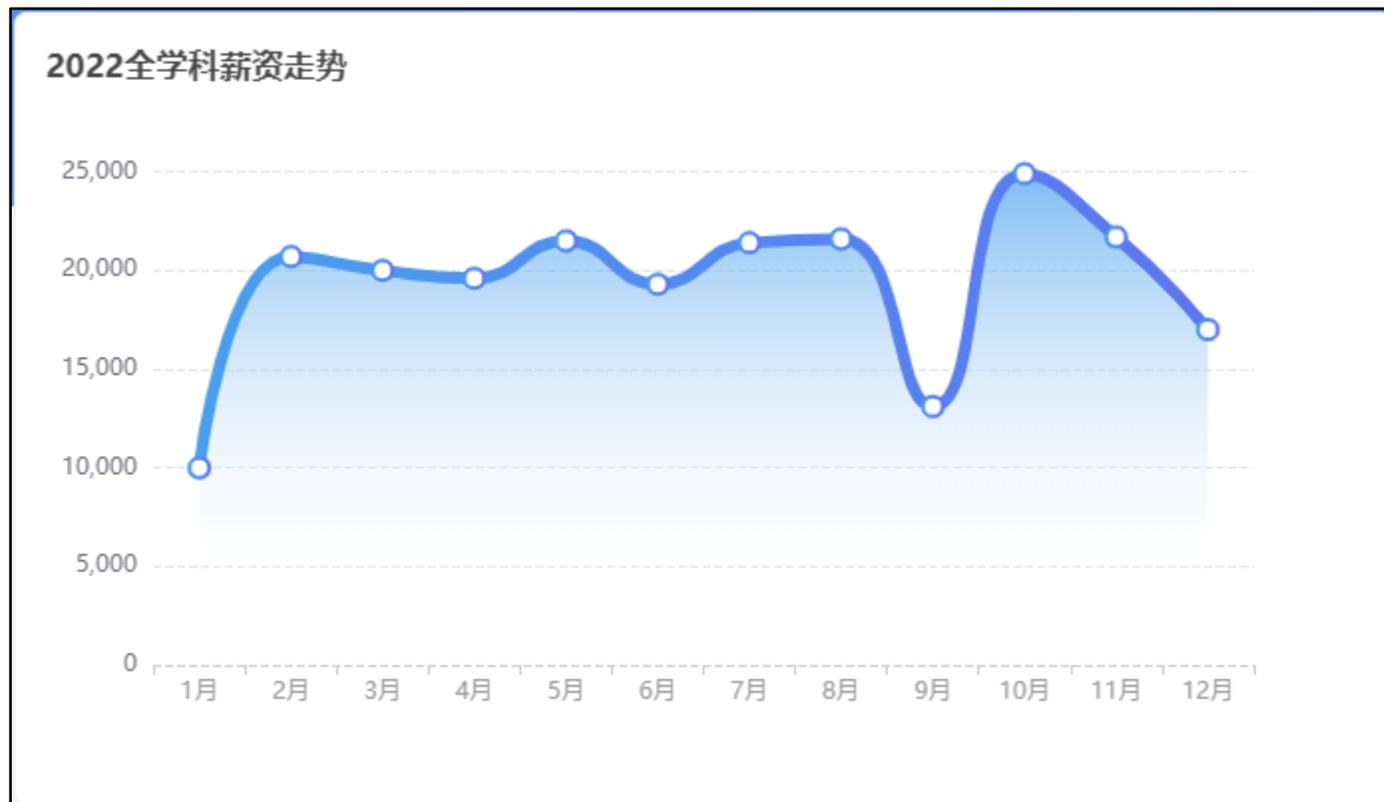
整合到项目



调整设置

```
const option = {  
  series: [  
    {  
      areaStyle: {  
        color: ''  
      },  
      lineStyle: {  
        color: ''  
      }  
    }  
  ]  
}
```

series:系列列表



数据看板-薪资走势

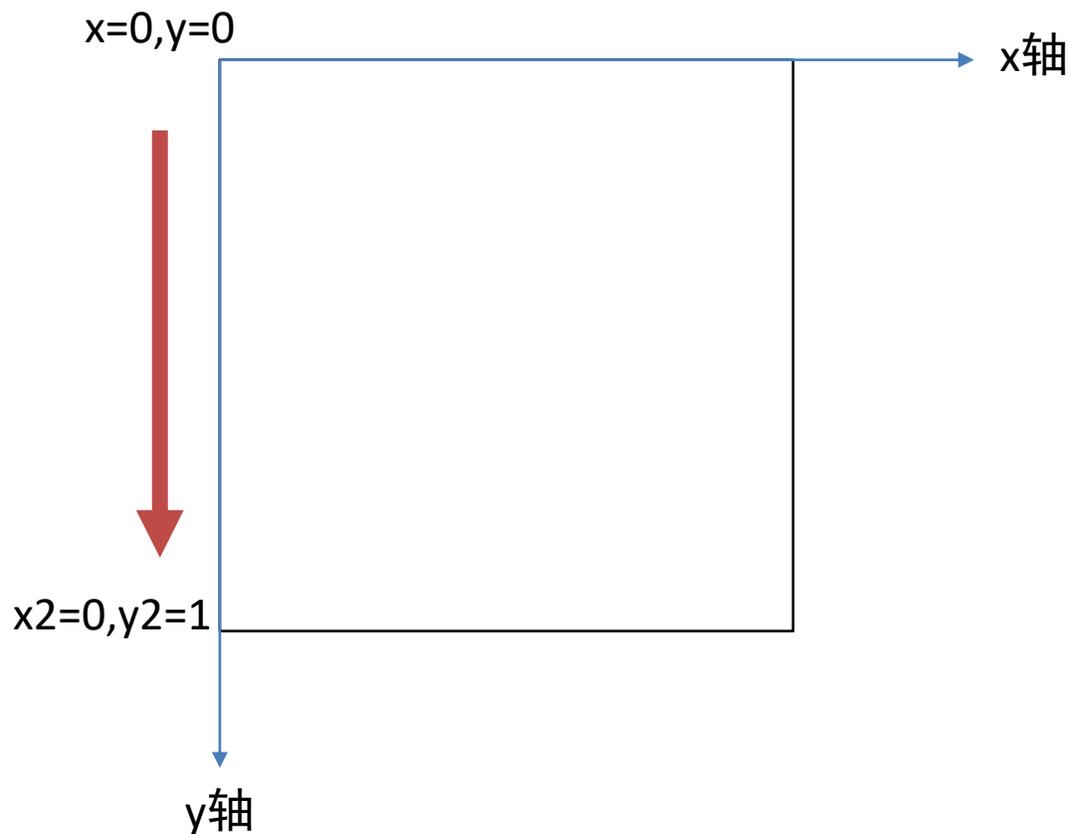
核心步骤:

找示例

整合到项目

调整设置

```
// x0, y0, x2, y2, 范围从 0 - 1, 相当于在图形包围  
// 盒中的百分比  
{  
  type: 'linear',  
  x: 0,  
  y: 0,  
  x2: 0,  
  y2: 1,  
  colorStops: [{  
    offset: 0, color: 'red' // 0% 处的颜色  
  }, {  
    offset: 1, color: 'blue' // 100% 处的颜色  
  }],  
  global: false // 缺省为 false  
}
```



数据看板-薪资走势

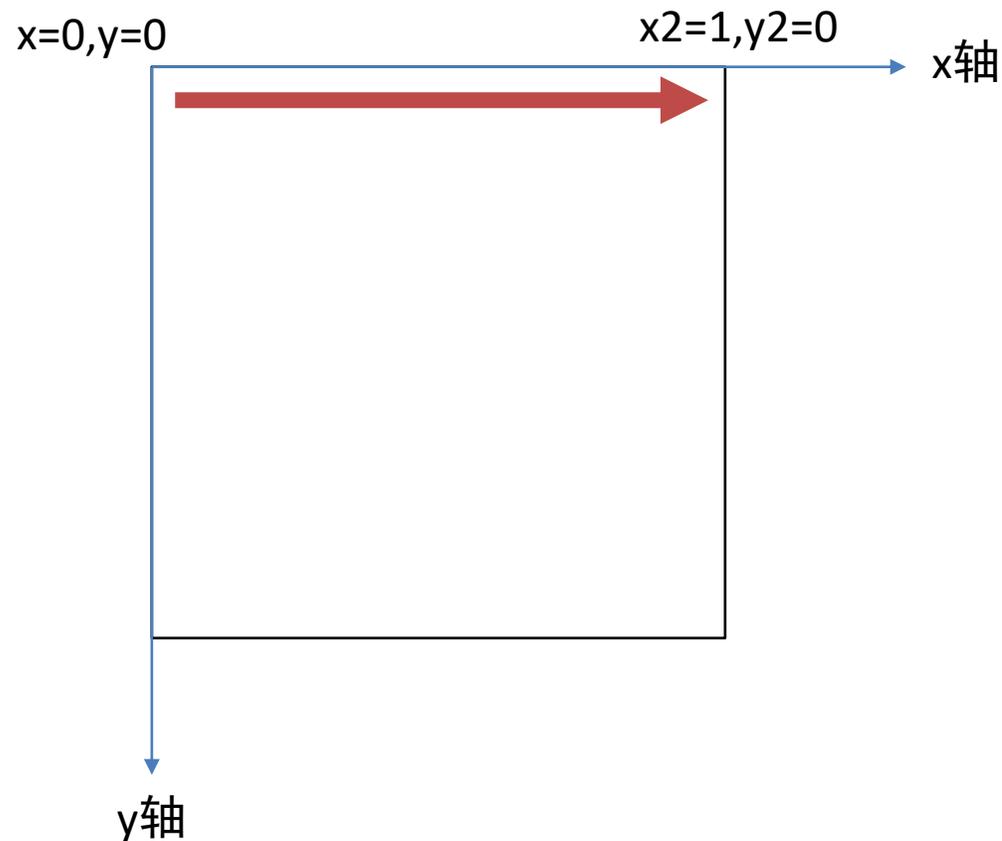
核心步骤:

找示例

整合到项目

调整设置

```
// x0, y0, x2, y2, 范围从 0 - 1, 相当于在图形包围  
// 盒中的百分比  
{  
  type: 'linear',  
  x: 0,  
  y: 0,  
  x2: 1,  
  y2: 0,  
  colorStops: [{  
    offset: 0, color: 'red' // 0% 处的颜色  
  }, {  
    offset: 1, color: 'blue' // 100% 处的颜色  
  }],  
  global: false // 缺省为 false  
}
```



数据看板-薪资走势

核心步骤:

找示例



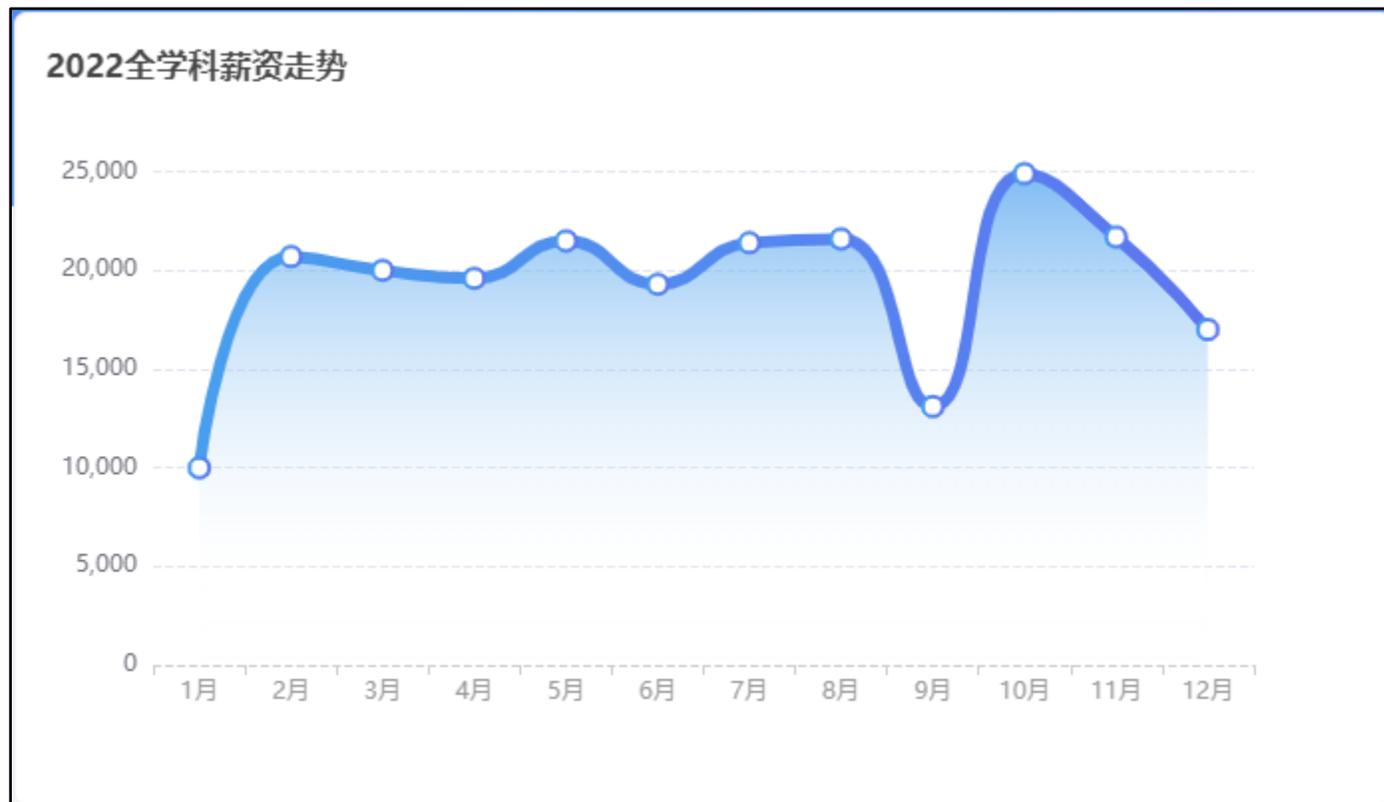
整合到项目



调整设置

```
const option = {  
  series: [  
    {  
      areaStyle: {  
        color: ''  
      },  
      lineStyle: {  
        color: ''  
      }  
    }  
  ]  
}
```

series:系列列表



数据看板-薪资走势

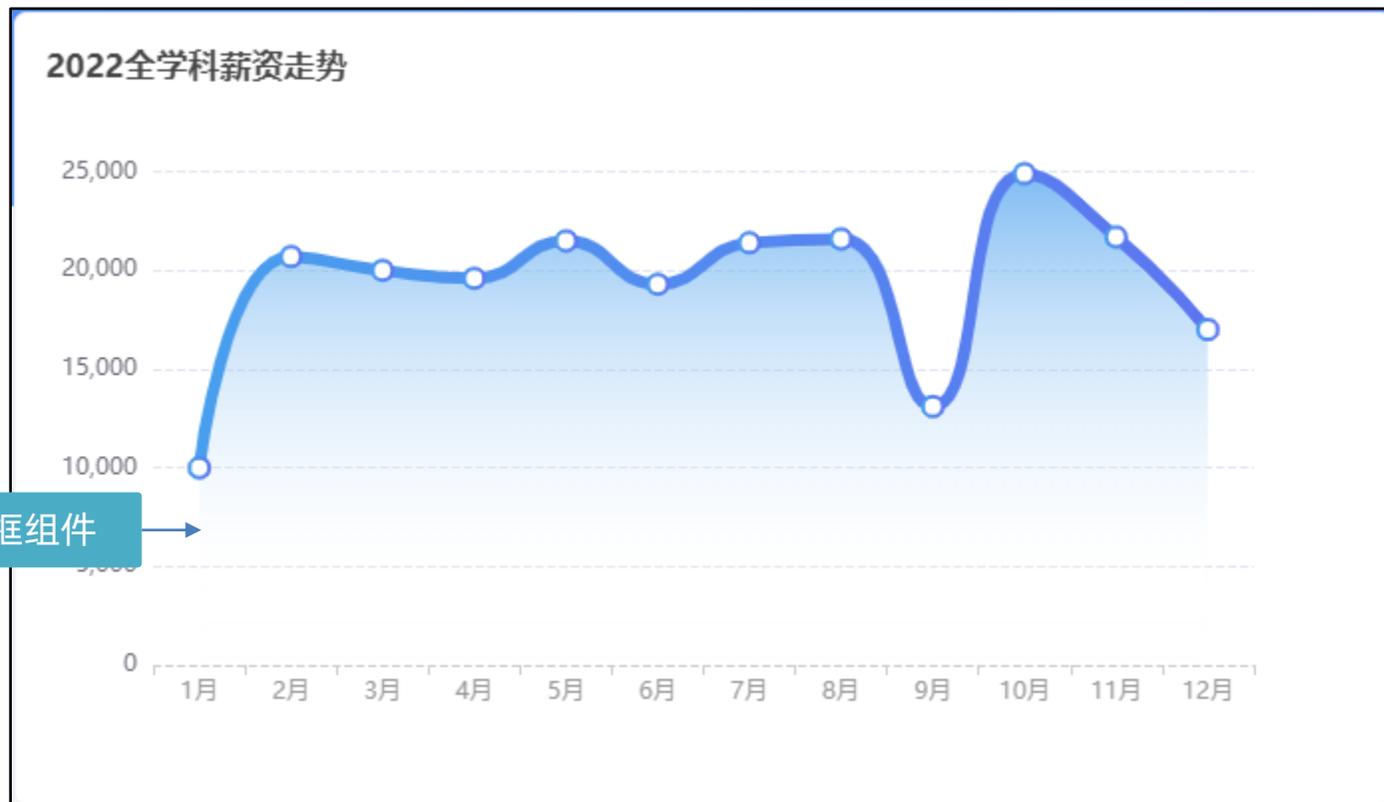
核心步骤:

找示例

整合到项目

调整设置

```
const option = {  
  tooltip: {  
    show: '',  
    trigger: ''  
  }  
}
```



数据看板-薪资走势

核心步骤:

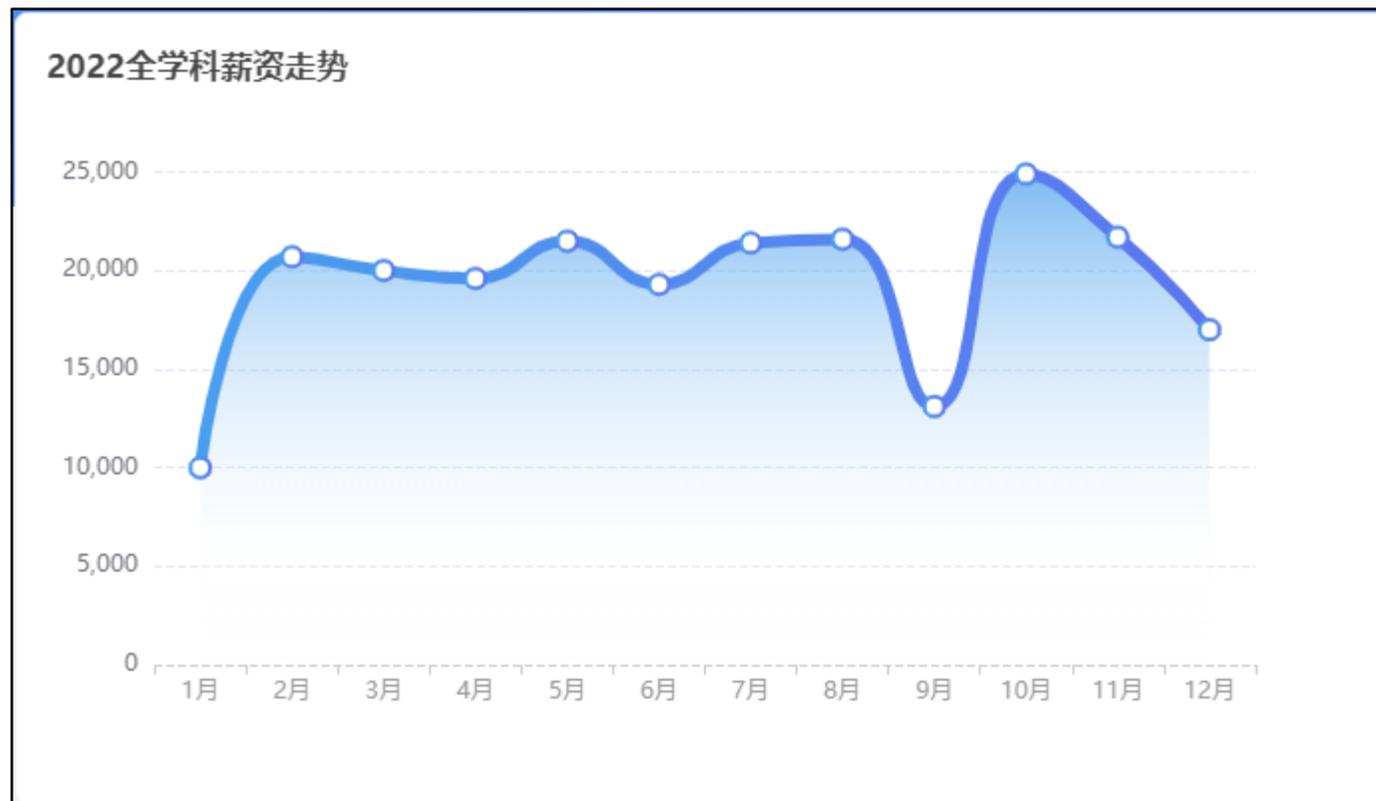
找示例



整合到项目



调整设置



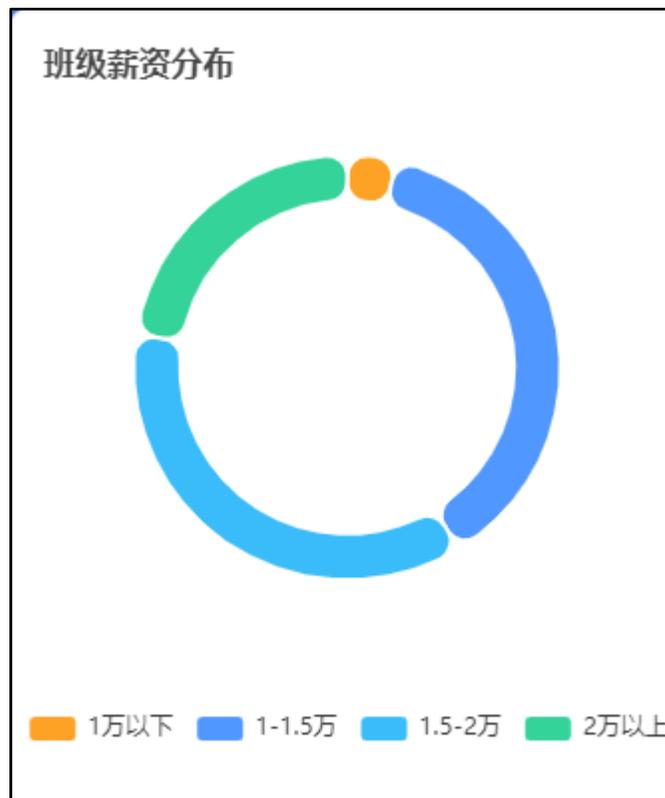
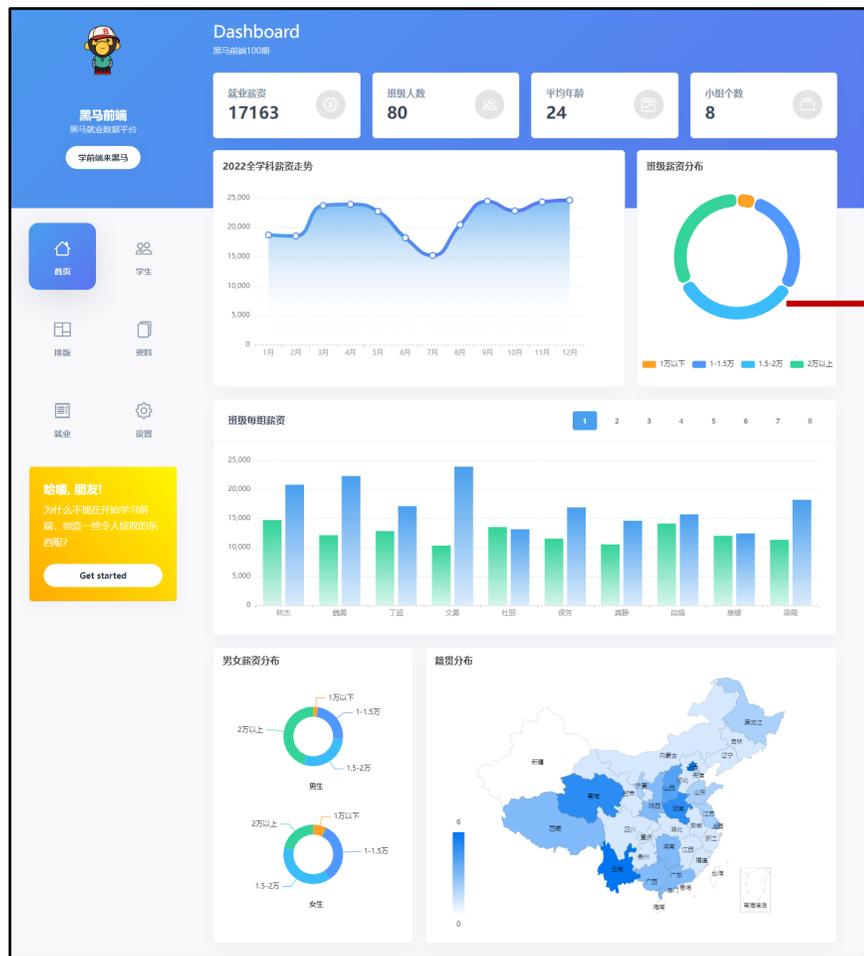
数据看板-薪资分布

核心步骤:

找示例

整合到项目

调整设置



数据看板-薪资分布

核心步骤:

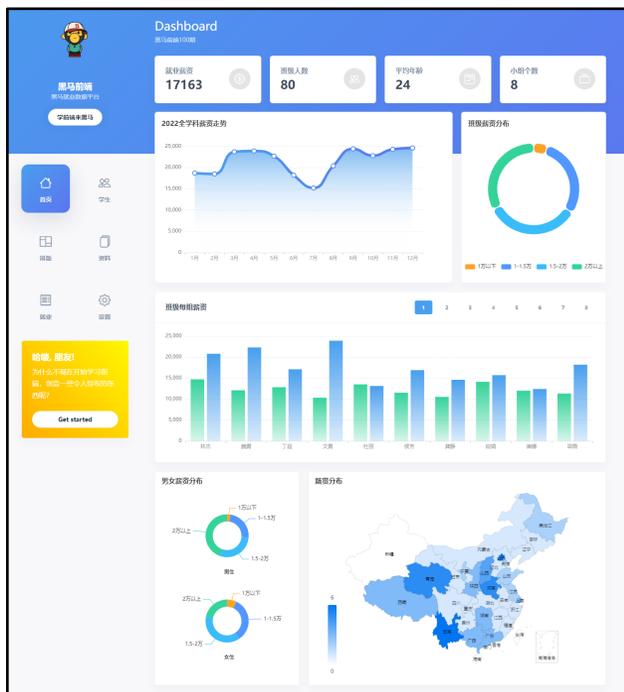
找示例



整合到项目



调整设置



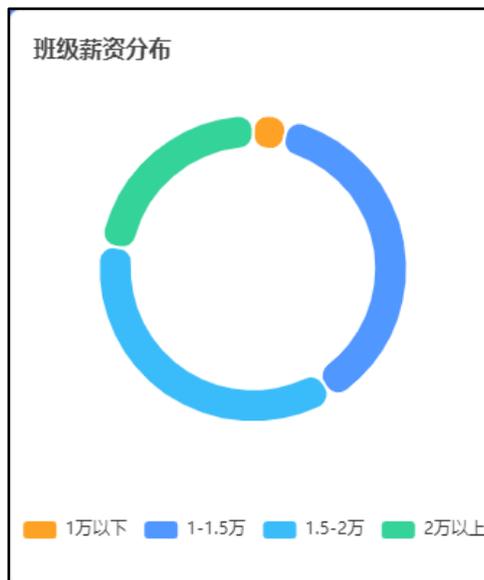
抽取函数



调用并传递数据



整合示例



数据看板-薪资分布

核心步骤:

找示例

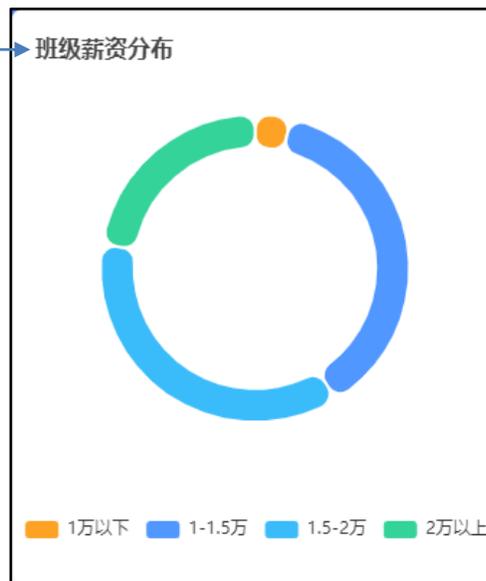
整合到项目

调整设置

```
const option = {  
  title: {  
    left: '',  
    top: ''  
  },  
  series: [  
    {  
      name: '',  
      type: '',  
      radius: '',  
      itemStyle: {}  
    }  
  ]  
}
```

title:标题组件

series:系列列表



数据看板-薪资分布

核心步骤:

找示例



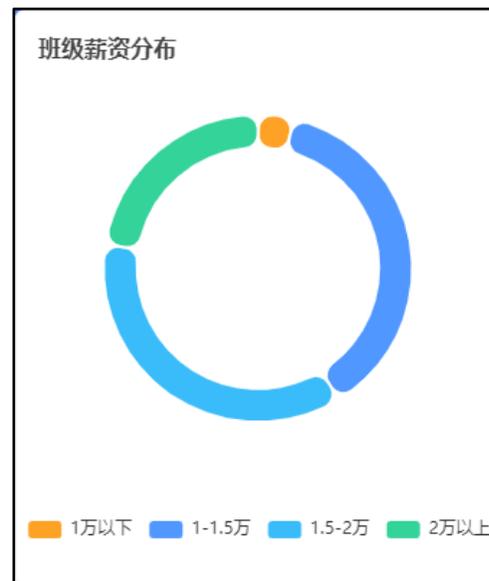
整合到项目



调整设置

```
const option = {  
  series: [  
    {  
      emphasis: {},  
      label: {},  
      labelLine: {},  
      avoidLabelOverlap: '',  
    }  
  ]  
}
```

series:系列列表



数据看板-薪资分布

核心步骤:

找示例



整合到项目

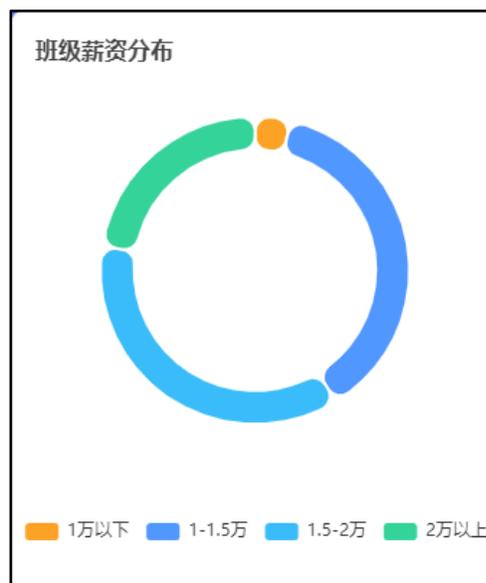


调整设置

```
const option = {  
  series: [  
    {  
      data: []  
    }  
  ],  
  color: []  
}
```

series:系列列表

color:颜色



数据看板-每组薪资

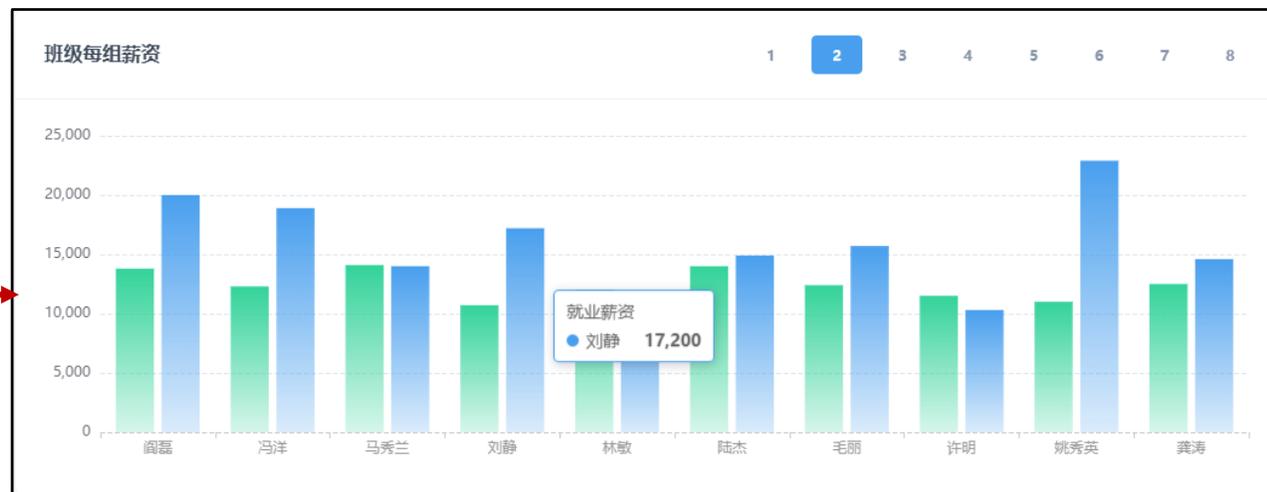
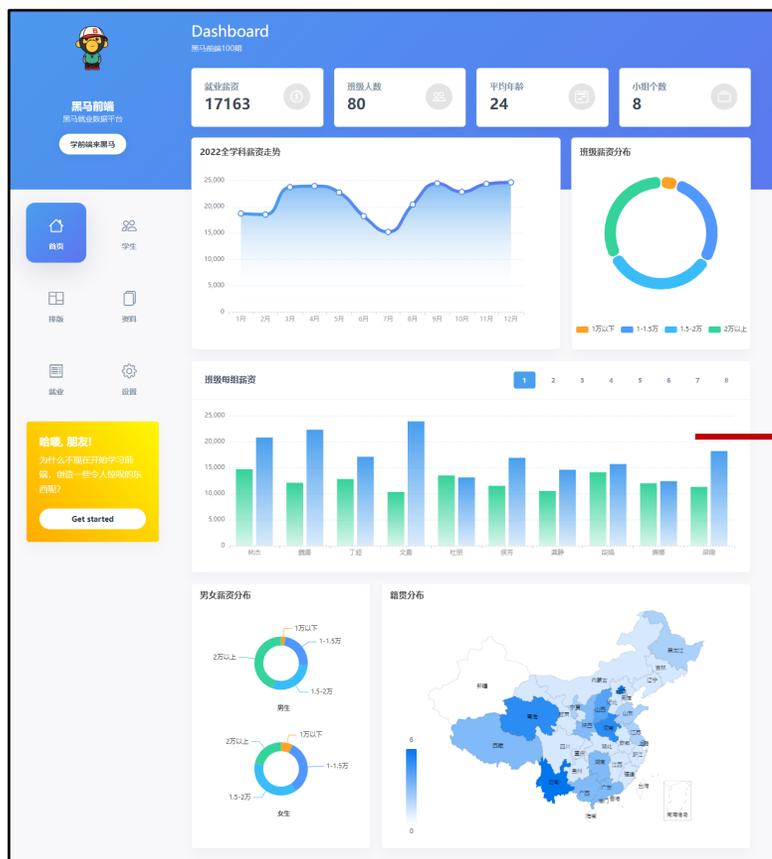
核心步骤:

找示例

整合到项目

调整设置

切换效果



数据看板-每组薪资

核心步骤:

找示例

整合到项目

调整设置

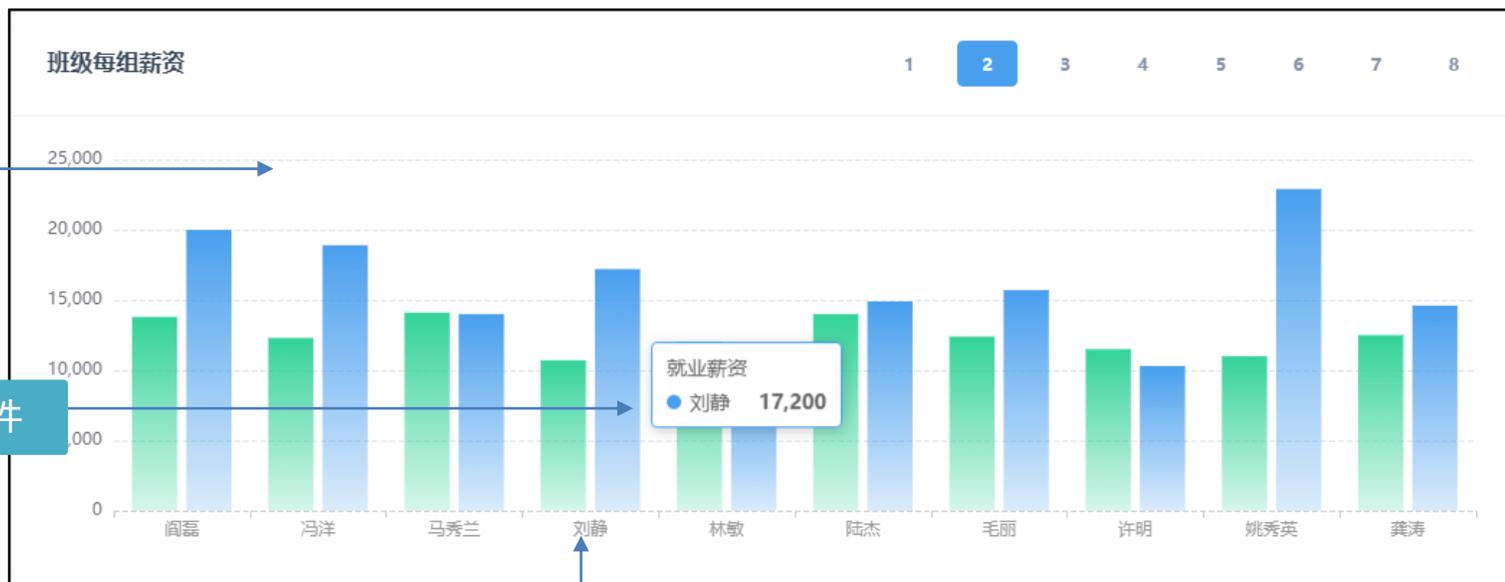
切换效果

```
const option = {  
  tooltip: {},  
  xAxis: {  
    axisLine: {  
     LineStyle: {  
        color: '',  
        type: ''  
      }  
    }  
  },  
  yAxis: {  
    splitLine: {  
     LineStyle: {  
        type: ''  
      }  
    }  
  }  
}
```

yAxis:y轴

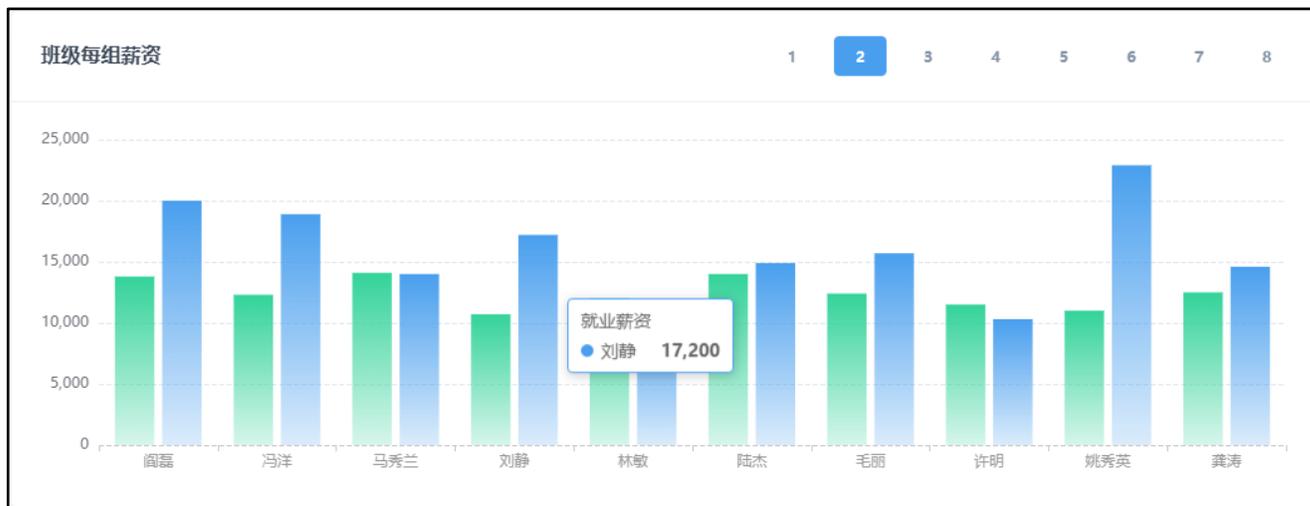
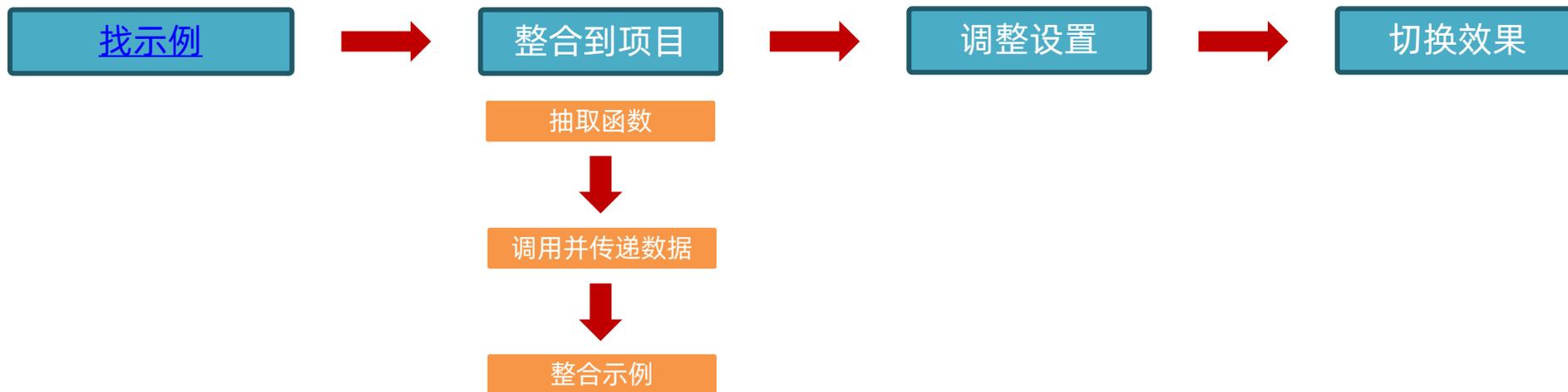
tooltip:提示框组件

xAxis:x轴



数据看板-每组薪资

核心步骤:



数据看板-每组薪资

核心步骤:

找示例

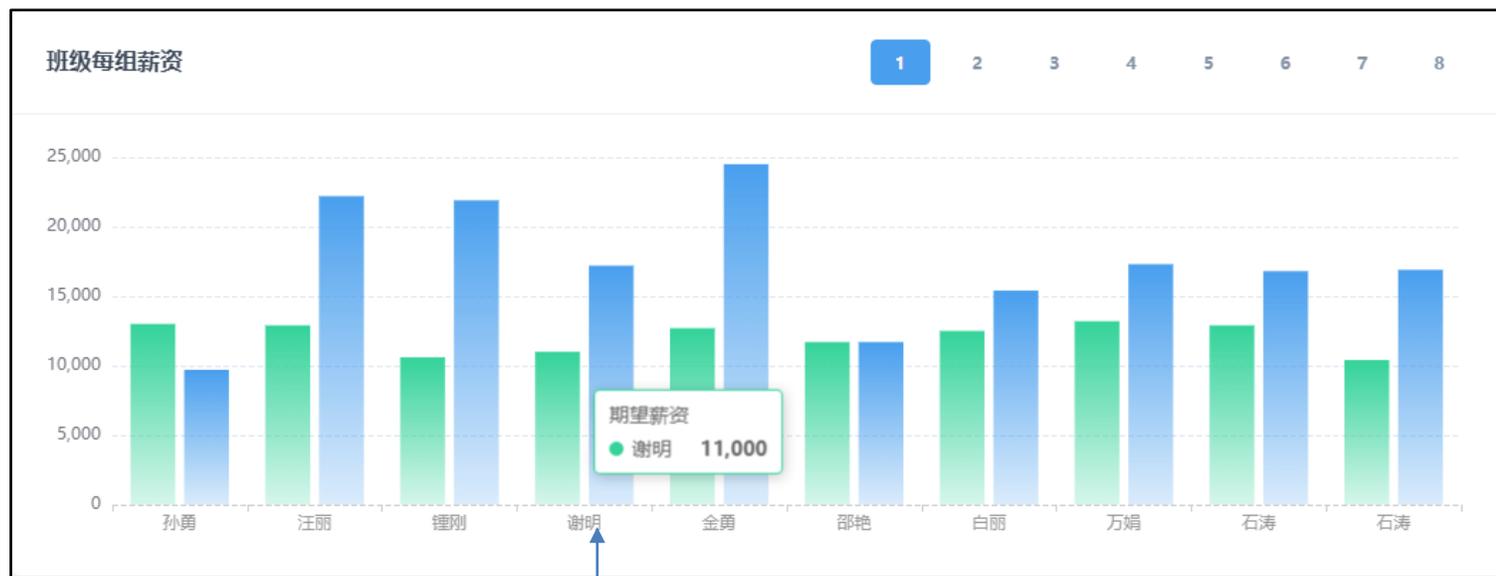
整合到项目

调整设置

切换效果

```
const option = {  
  grid: {  
    left: 70,  
    top: 30,  
    right: 30,  
    bottom: 50,  
  },  
  xAxis: {  
    data: [],  
    axisLabel: {  
      color: '#999'  
    }  
  }  
}
```

grid:绘图网络



xAxis:x轴

数据看板-每组薪资

核心步骤:

找示例

整合到项目

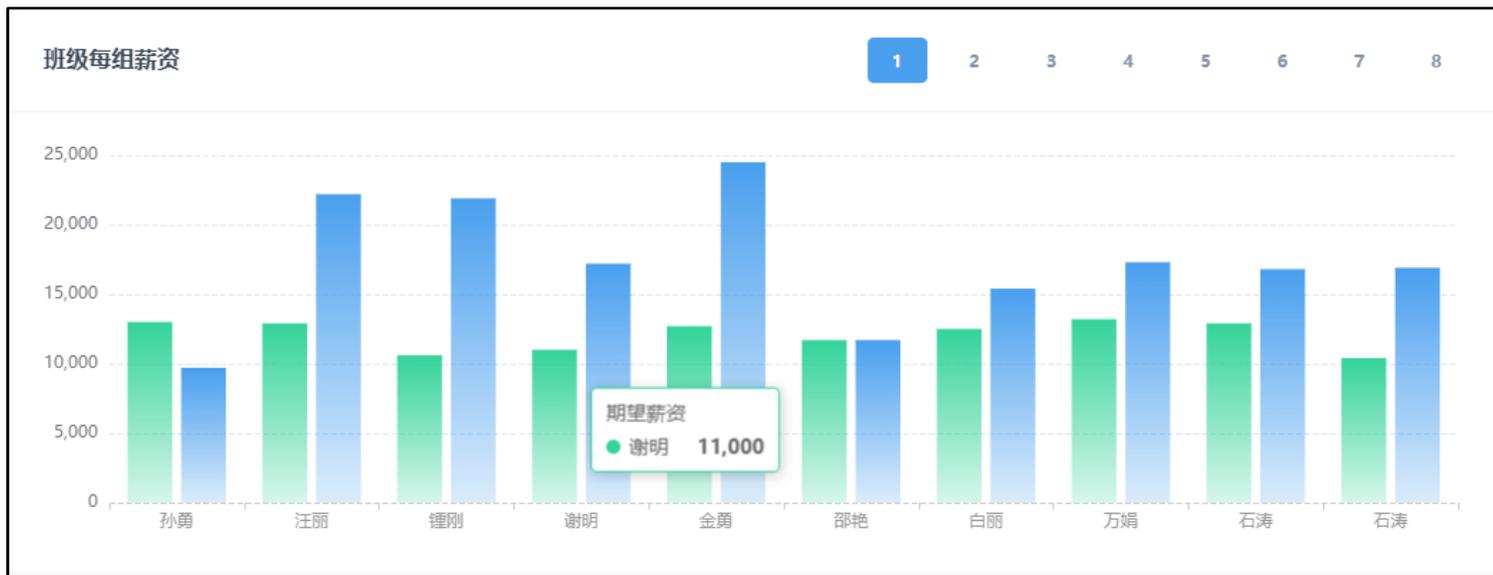
调整设置

切换效果

```
const option = {  
  series: [  
    {  
      name: '',  
      data:[],  
      itemStyle: {  
        color: ''  
      }  
    }  
  ]  
}
```

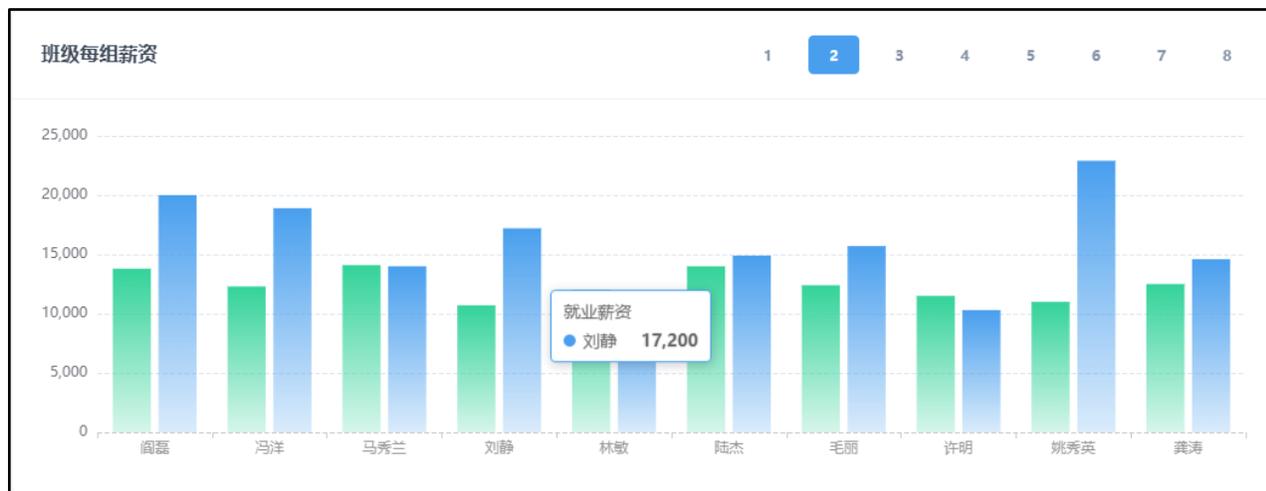
```
// 左侧0%颜色 #34D39A  
// 左侧100%颜色 rgba(52,211,154,0.2)  
// 右侧0%颜色 #499FEE  
// 右侧100%颜色 rgba(73,159,238,0.2)
```

series:系列列表



数据看板-每组薪资

核心步骤:



高亮切换

数据切换

重新渲染

```
myChart.setOption(option)
```

数据看板-男女薪资分布

核心步骤:

找示例

整合到项目

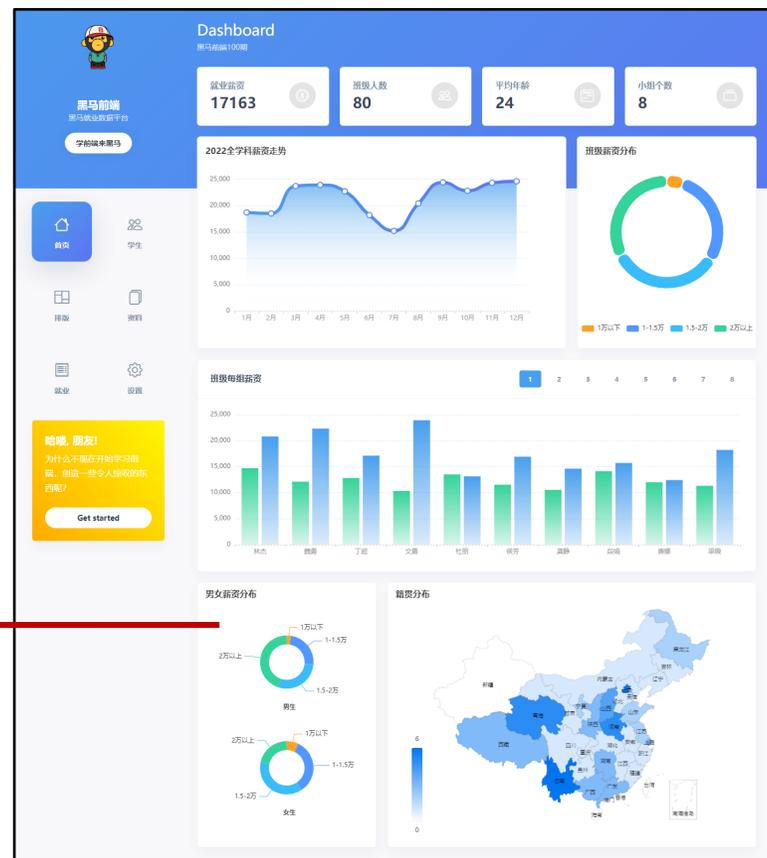
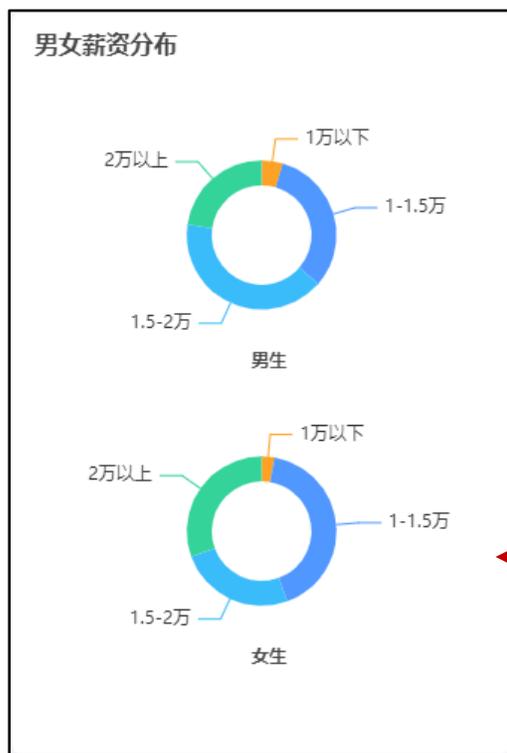
调整设置

legend:图例组件

color:颜色

title:标题组件

series:系列列表



数据看板-男女薪资分布

核心步骤:

找示例



整合到项目



调整设置

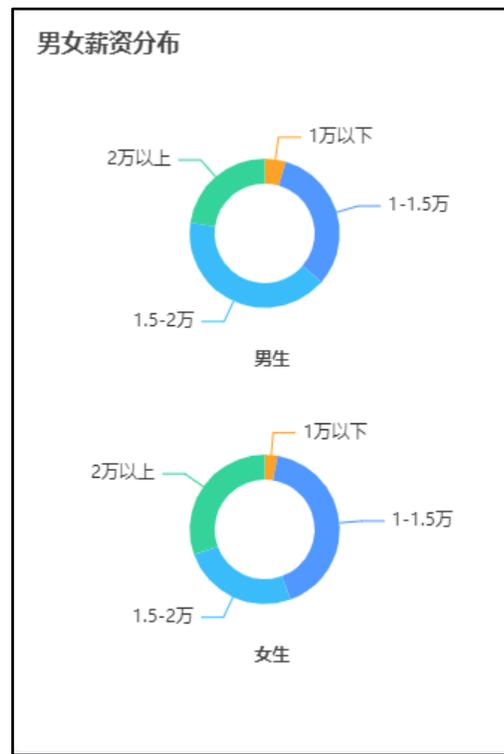
抽取函数



调用并传递数据



整合示例



数据看板-男女薪资分布

核心步骤:

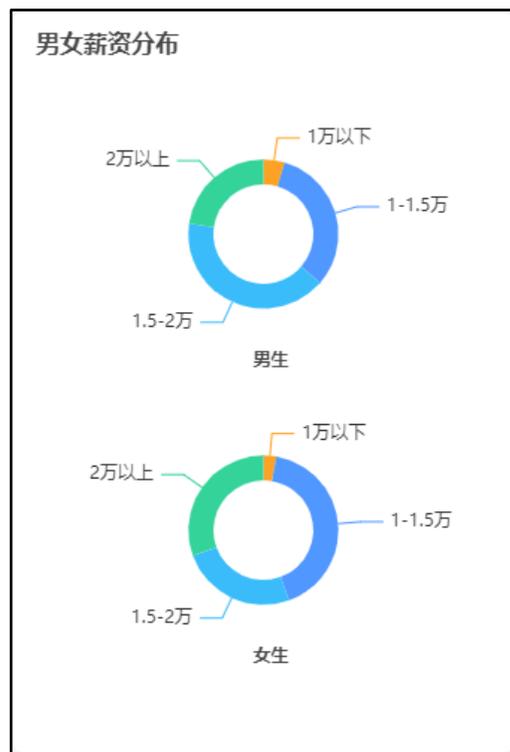
找示例

整合到项目

调整设置

```
const option = {  
  title: [  
    {  
      left: 10,  
      top: 10,  
      textStyle: {  
        fontSize: 16  
      }  
    },  
    {  
      left: '50%',  
      top: '45%',  
      textStyle: {  
        fontSize: 12  
      }  
    },  
    {  
      left: '50%',  
      top: '85%',  
      textStyle: {  
        fontSize: 12  
      }  
    }  
  ],  
}
```

title:标题组件



数据看板-男女薪资分布

核心步骤:

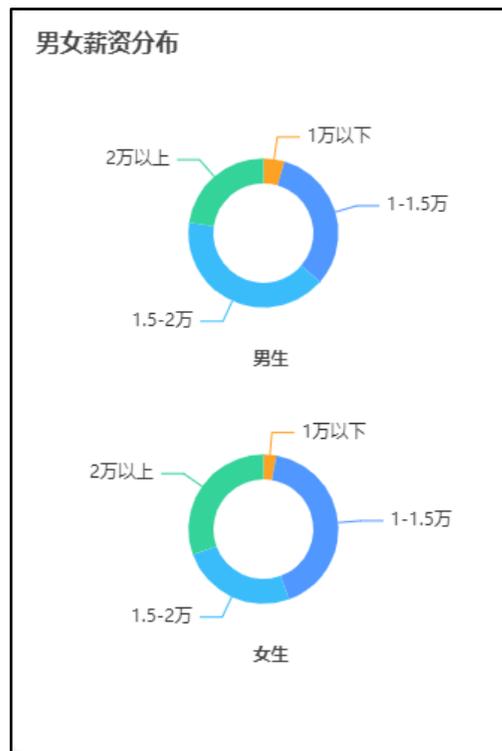
找示例

整合到项目

调整设置

```
const option = {  
  series: [  
    {  
      radius: ['20%', '30%'],  
      center: ['50%', '30%'],  
      data: []  
    },  
    {  
      radius: ['20%', '30%'],  
      center: ['50%', '70%'],  
      data: []  
    }  
  ]  
}
```

series:系列列表



数据看板-籍贯分布

核心步骤:

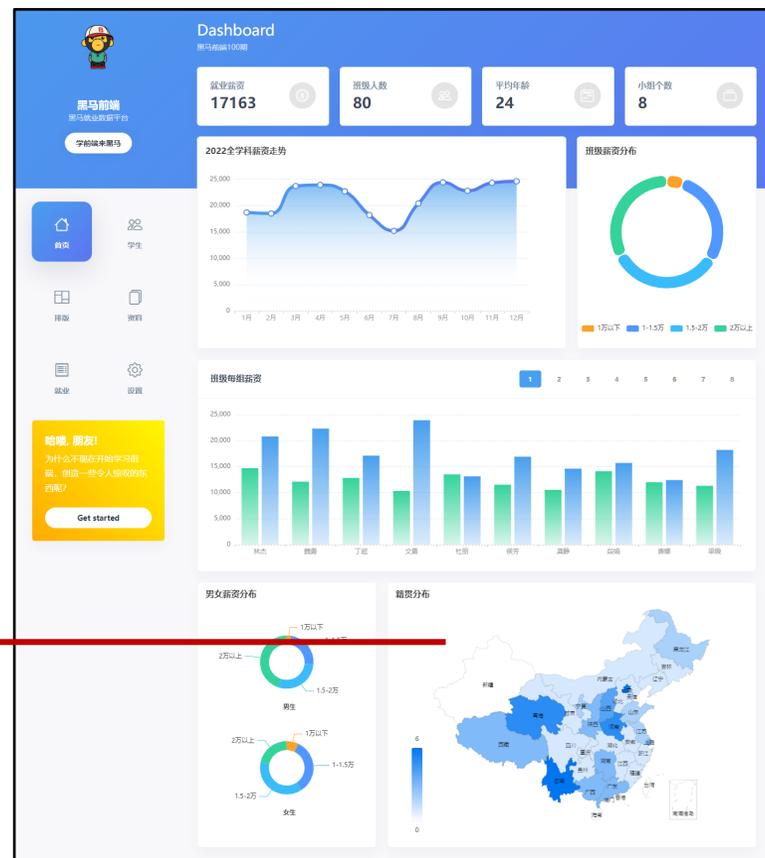
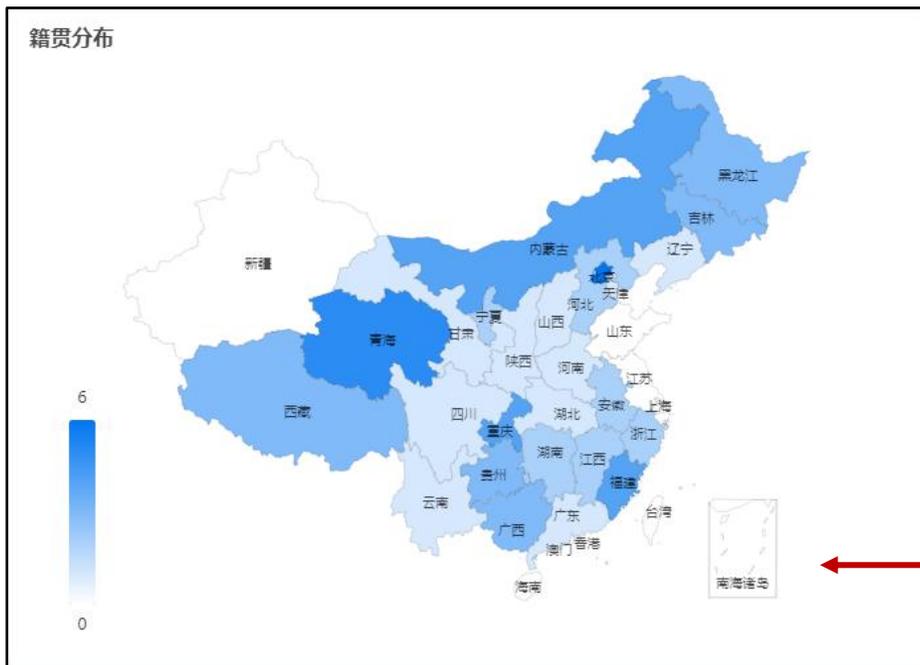
社区找示例



整合到项目



调整设置



数据看板-籍贯分布

核心步骤:

社区找示例



整合到项目



调整设置

筛选数据



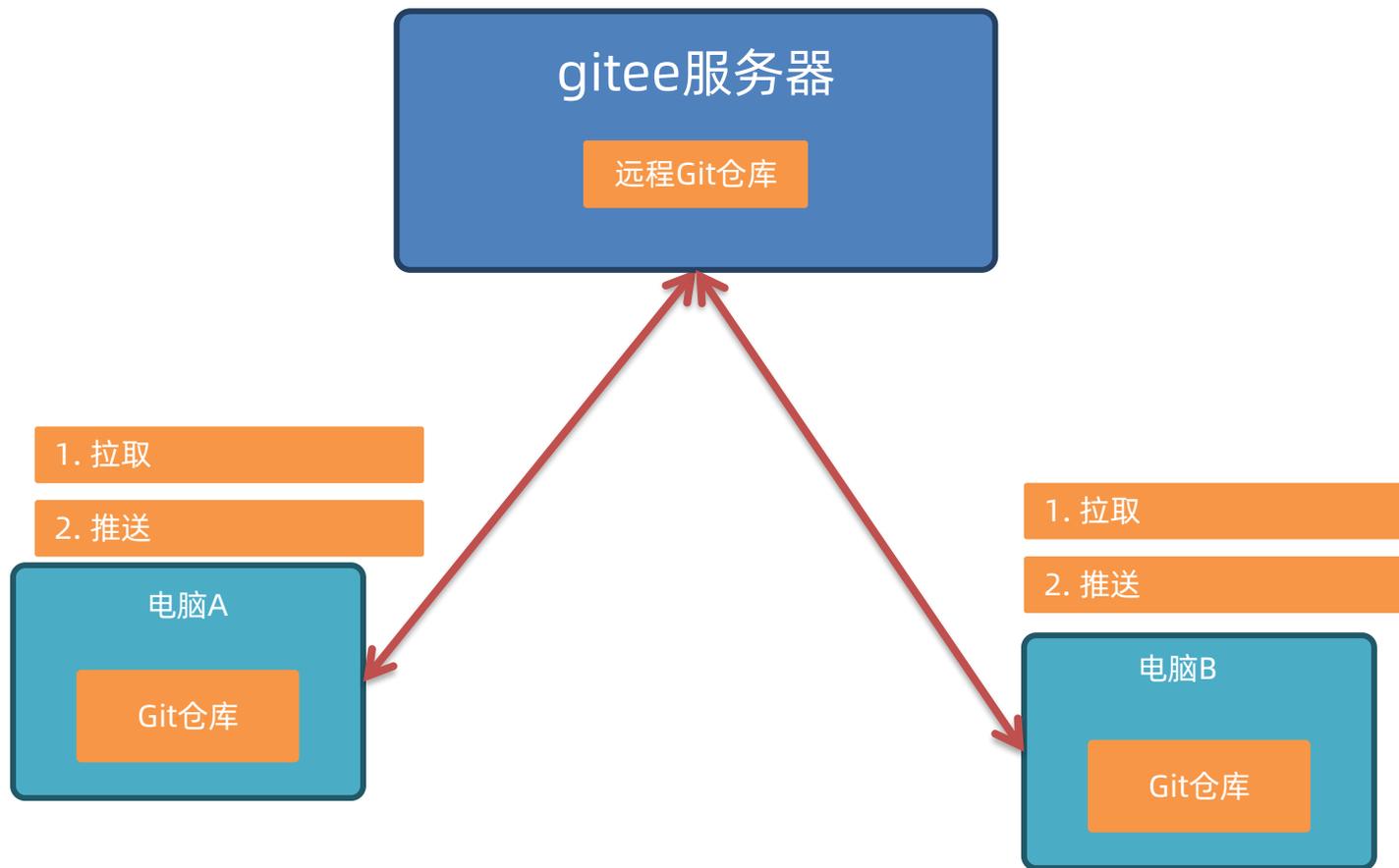
数据赋值

```
const dataList = [  
  { name: '南海诸岛', value: 0 },  
  { name: '北京', value: 0 },  
  { name: '天津', value: 0 },  
  { name: '上海', value: 0 },  
  { name: '重庆', value: 0 },  
  { name: '河北', value: 0 },  
  { name: '河南', value: 0 },  
  { name: '云南', value: 0 },  
  { name: '辽宁', value: 0 },  
  { name: '黑龙江', value: 0 },  
  { name: '湖南', value: 0 },  
  { name: '安徽', value: 0 },  
  { name: '山东', value: 0 },  
  // ....略  
]
```

```
provinceData: Array(27)  
▶ 0: {name: '江西省', value: 2}  
▶ 1: {name: '北京', value: 23}  
▶ 2: {name: '吉林省', value: 3}  
▶ 3: {name: '福建省', value: 4}  
▶ 4: {name: '贵州省', value: 3}  
▶ 5: {name: '重庆', value: 4}  
▶ 6: {name: '浙江省', value: 2}  
▶ 7: {name: '安徽省', value: 2}  
▶ 8: {name: '河北省', value: 2}  
▶ 9: {name: '湖南省', value: 2}  
▶ 10: {name: '青海省', value: 5}  
▶ 11: {name: '甘肃省', value: 1}  
▶ 12: {name: '新疆维吾尔自治区', value: 3}  
▶ 13: {name: '天津', value: 1}  
▶ 14: {name: '黑龙江省', value: 3}
```



数据看板-代码上传





传智教育旗下高端IT教育品牌