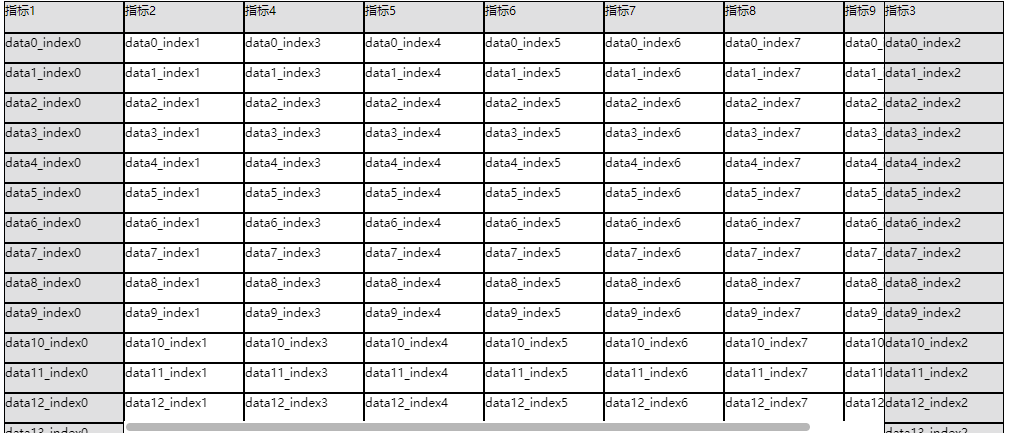
1. 需求分析

通过对需求分析,需要10万个格子以上数据显示,进行技术选型,决定采用react-window虚拟组件,考虑到固定行列功能, 页面架构决定采用如下架构,Tbody使用react-window实现

经过方案预演,实现了Table1.0版本,如下所示

发现几个问题,1.由于三者table分开,滚动需要同步,采用scrollTo方式进行三者同步,存在三者同步不协调,卡顿问题。2.下方滚动条只处于中间table处。

通过分析,决定放弃此方案, 采用如下新页面架构方案(利用css的sticky属性)

经过方案预演,实现了初步的2.0版本,具备可行性,对需求进行如下细化拆分:

Thead:实现固定行, 支持列排序(前端), 列支持"数值"类型,可设置小数位, 列支持"日期"类型，可设置日期格式, 支持千分位, 列宽可调整, 列顺序可调整(拖拽), 列居左,中,右功能

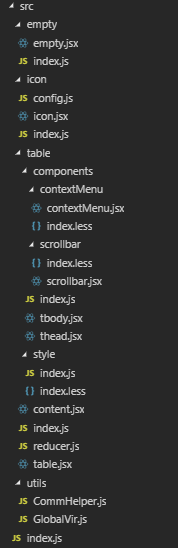
左侧固定列,右侧固定列:实现固定列功能

Tbody,滚动条组件:实现10万个格子以上数据显示(行定高), 滚动整行/整列功能

菜单:实现行可删除,列可删除,单元格可编辑,单元格可选中, 行可选中, 列可选中, 复制选中, 可旋转功能

1. 项目结构

项目采用如下结构:



Empty:空组件,当table数据为空的时候,显示此组件

Icon:图标组件,目前支持上下箭头(排序功能),以及菜单需要的图标,可以通过config.js进行配置,随时添加图标,如下图所示。



Table:

包括菜单组件,滚动条组件,考虑到这些组件数据需要与table交互,所以没有单独提出来,后续可以考虑单独隔离出来。

Tbody表格内容,

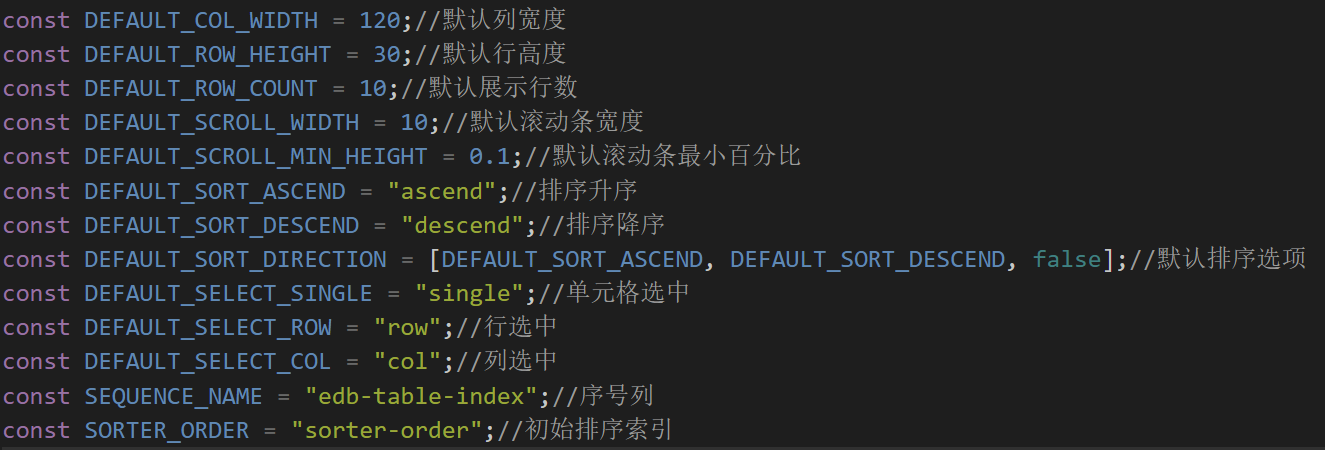
thead表头,

reducer.js,使用reduce进行状态管理

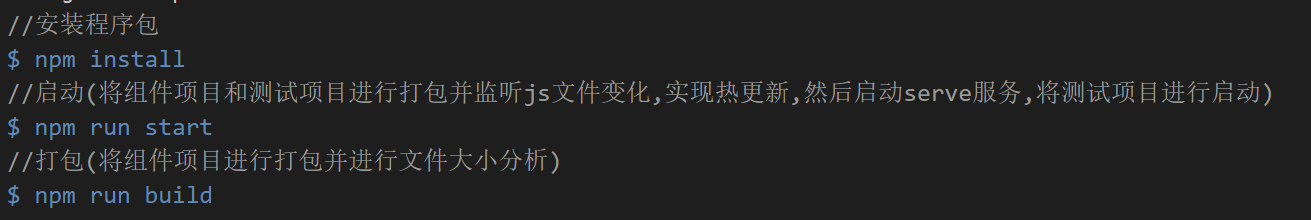
Utils:

CommHelper.js:通用方法

GlobalVir.js:全局变量,包括一些组件的基础配置,如下图所示。



1. 启动及打包方法



1. 基本使用方法

import { Table } from "Table";

let columns = [

{

title: "姓名",

dataIndex: "name",

},

{

title: "出生日期",

dataIndex: "date"

},

{

title: "年龄",

dataIndex: "age"

},

{

title: "住址",

dataIndex: "address"

},

];

let dataSource = [

{ name: "胡彦斌", date: "1988-03-11", age: 32, address: "西湖区湖底公园1号" },

{ name: "吴彦祖", date: "1978-04-06", age: 42, address: "浙江省杭州市西湖区湖底公园2号" },

];

<Table columns={columns} dataSource={dataSource}></Table>

1. API





1. 程序截图

