

山东大学 计算机科学与技术 学院

可视化技术 课程实验报告

学号：202300130242	姓名：王启源	班级：数据 23
实验题目：电子表格实践		
实验学时：2	实验日期：2025-10-29	
实验目的：对 x-spreadsheet 进行修改，添加一个可视化数据的功能		
硬件环境： 计算机一台		
软件环境： Linux 或 Windows		
实验步骤与内容： 1. 在 src 文件夹中添加了 visualization 文件夹来实现可视化功能 2. 其中添加了 get_data 函数，用于将选中区域的数据得到，并返回给可视化函数。同时添加了 vis 函数，用于将得到的数据进行一个可视化 3. 在 src/component/contextmenu.js 中找到 menuItems，在其中添加可视化的功能键，点击后可以得到可视化结果 4. 在 src/sheet.js 中有将点击按键和相应的功能联系在一起的代码，在文件的以下部分		

```

698 // contextmenu
699 contextMenu.itemClick = (type) => {
700   // console.log('type:', type);
701   if (type === 'validation') {
702     modalValidation.setValue(this.data.getSelectedValue);
703   } else if (type === 'copy') {
704     copy.call(this);
705   } else if (type === 'cut') {
706     cut.call(this);
707   } else if (type === 'paste') {
708     paste.call(this, 'all');
709   } else if (type === 'paste-value') {
710     paste.call(this, 'text');
711   } else if (type === 'paste-format') {
712     paste.call(this, 'format');
713   } else if (type === 'hide') {
714     hideRowsOrCols.call(this);
715   } else if (type === 'insert-row' || type === 'insert-column') {
716     insertDeleteRowColumn.call(this, type);
717   } else if (type === 'vis'){
718     const rows_data = get_data(this);
719     vis(rows_data);
720   }
721 };

```

在其中添加可视化的逻辑：判断该点击的该按钮是不是可视化，如果是的话，调用对应的可视化函数实现可视化结果

5. 以下是可视化函数的具体位置

```

└─ visualization
   └─ JS charts.js
   └─ JS get_data.js
   └─ JS confirm.js

```

结论分析与体会：

了解了在对于一个崭新的项目时，如何对该项目进行学习和修改
进一步熟悉了可视化操作