**更新记录**

目录

[一、 全局替换插件静态文件路径 2](#_Toc20153)

[二、添加图表可拖动区域高亮 2](#_Toc19838)

[三、添加顶部工具栏的保存按钮 4](#_Toc1861)

[四、添加Luckysheet单元格上拖拽停止的事件 7](#_Toc27421)

[五 预览时，合并单元格返回数据后，报错 9](#_Toc12732)

[六 增加弹窗创建图表的回调 9](#_Toc22106)

[七 完善全图表配置信息的获取 20](#_Toc21054)

[八 完善图表拖动或改变大小后更新图表位置和大小信息 21](#_Toc26566)

[九 添加设置sheet表配置hyperlink属性的方法 24](#_Toc30789)

## 全局替换插件静态文件路径

1）描述

替换编译后的luckysheet源码中所有的以下路径

原：expendPlugins/chart/chartmix.umd.min.js

目标：/luckysheet/expendPlugins/chart/chartmix.umd.min.js

## 二、添加图表可拖动区域高亮

1）描述

鼠标经过图标区域时，如果是支持拖拽，则显示十字，否则显示箭头。

2）JS代码

第一步



/\*\*

 \* 设置图表可拖动区域高亮效果，鼠标经过可拖动区域时鼠标显示“十字”，不可拖动区域显示箭头

 \* @param {JQuery} $container 图表的容器DIV

 \*/

function setChartMoveableEffect($container) {

  $container.find('.luckysheet-modal-dialog-content').hover(function () {

    $container.removeClass("chart-moveable");

  }, function () {

    $container.addClass("chart-moveable");

  });

  $container.hover(function () {

    $container.addClass("chart-moveable");

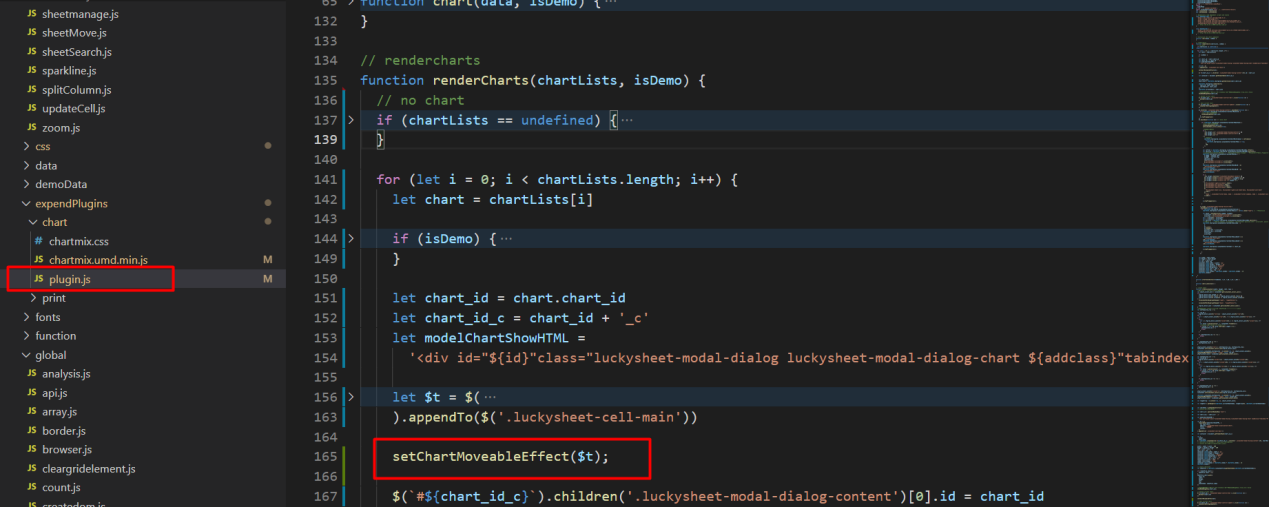
  }, function () {

    $container.removeClass("chart-moveable");

  });

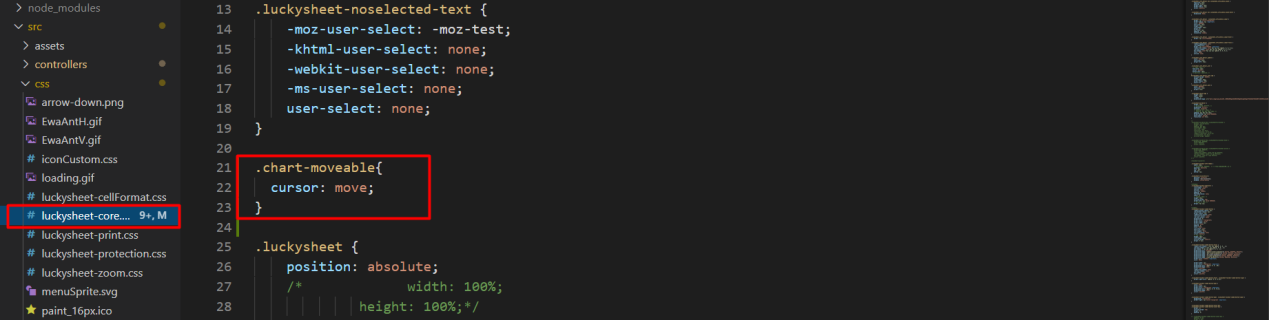
}

第二步





3）CSS代码

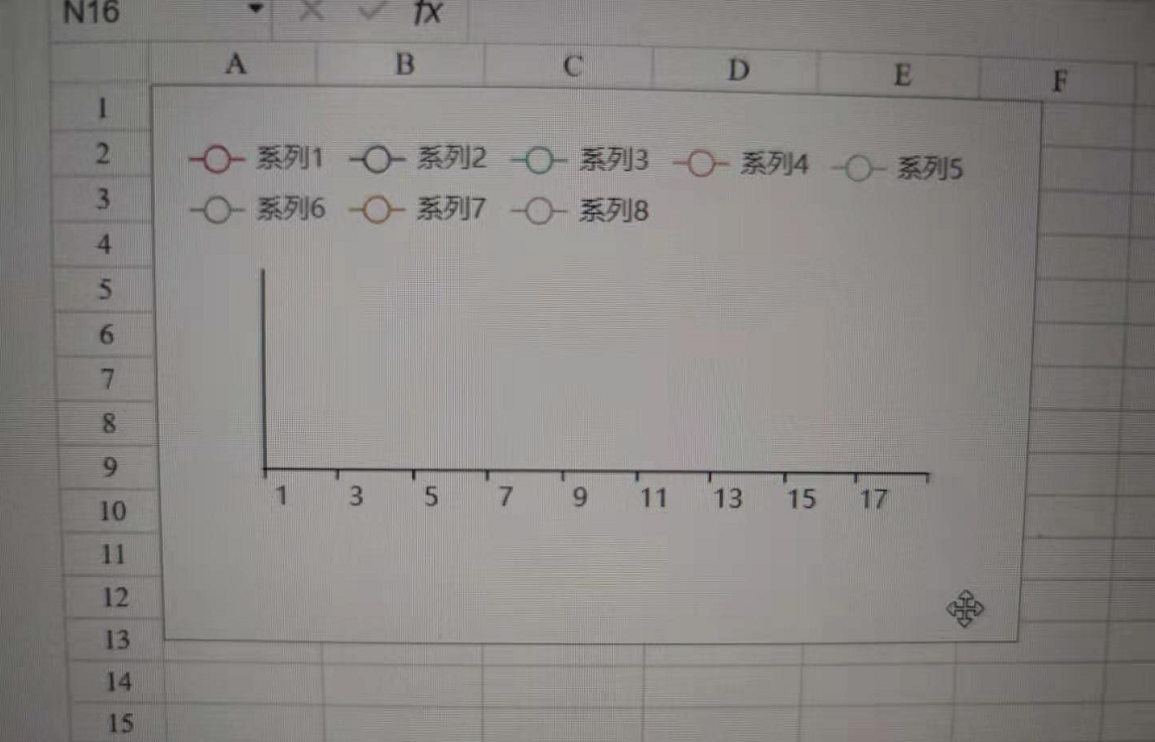


.chart-moveable{

  cursor: move;

}

3）效果



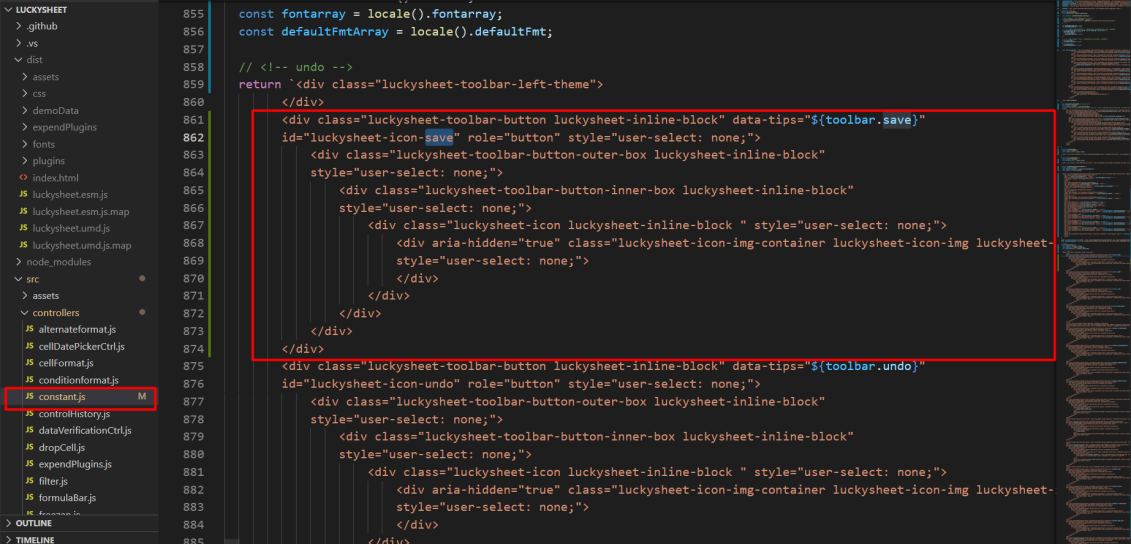
## 三、添加顶部工具栏的保存按钮

1）描述

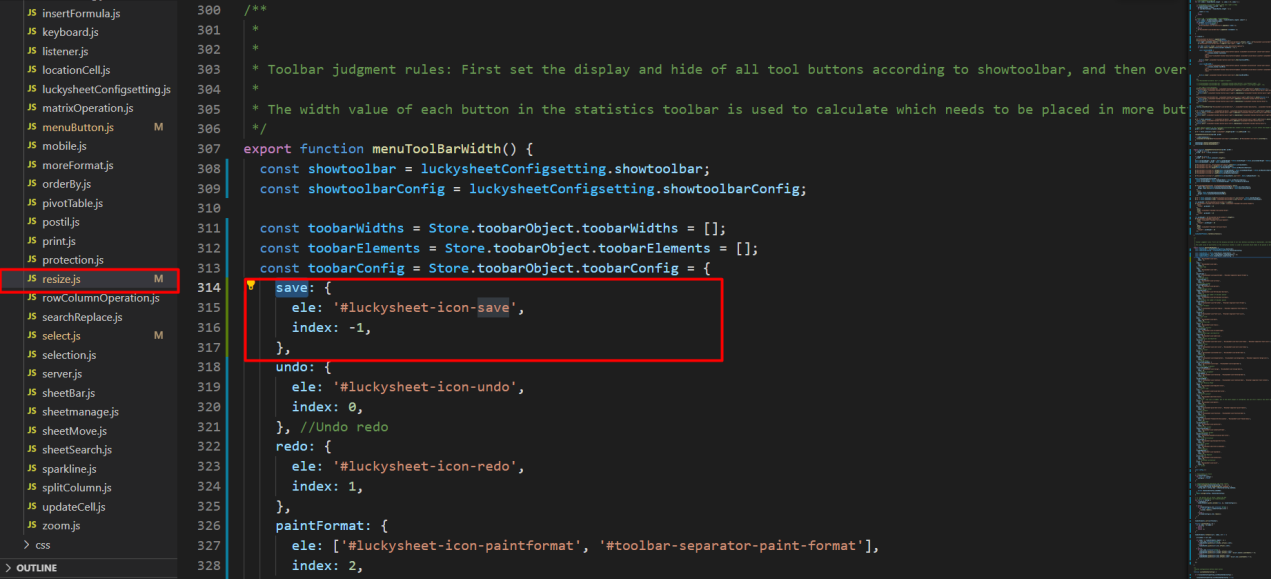
添加顶部工具栏的保存按钮，并实现保存的钩子。

2）代码

第一步：添加按钮的Html代码



第二步：添加按钮的配置，包括序号（按钮的位置），按钮对应Dom的ID



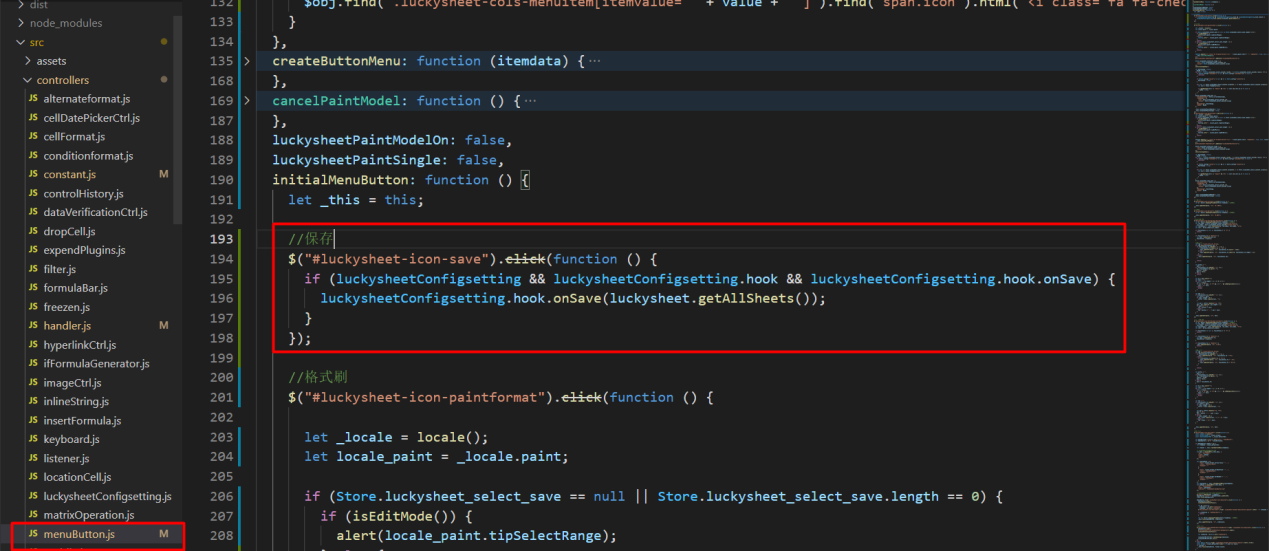
save: {

      ele: '#luckysheet-icon-save',

      index: -1,

    },

第三步：添加按钮的Hook



//保存

    $("#luckysheet-icon-save").click(function () {

      if (luckysheetConfigsetting && luckysheetConfigsetting.hook && luckysheetConfigsetting.hook.onSave) {

        luckysheetConfigsetting.hook.onSave(luckysheet.getAllSheets());

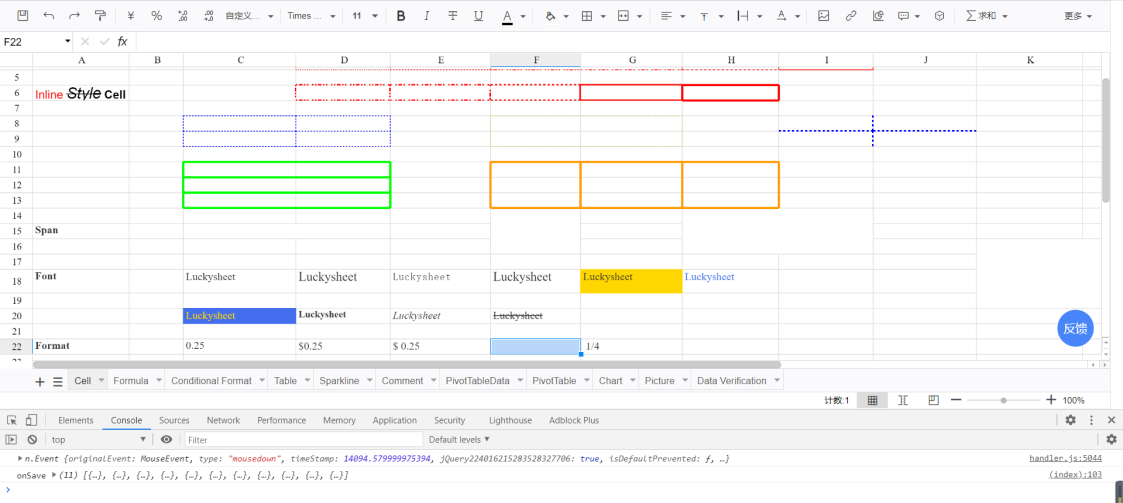
      }

});

第四步：配置国际化



3）效果

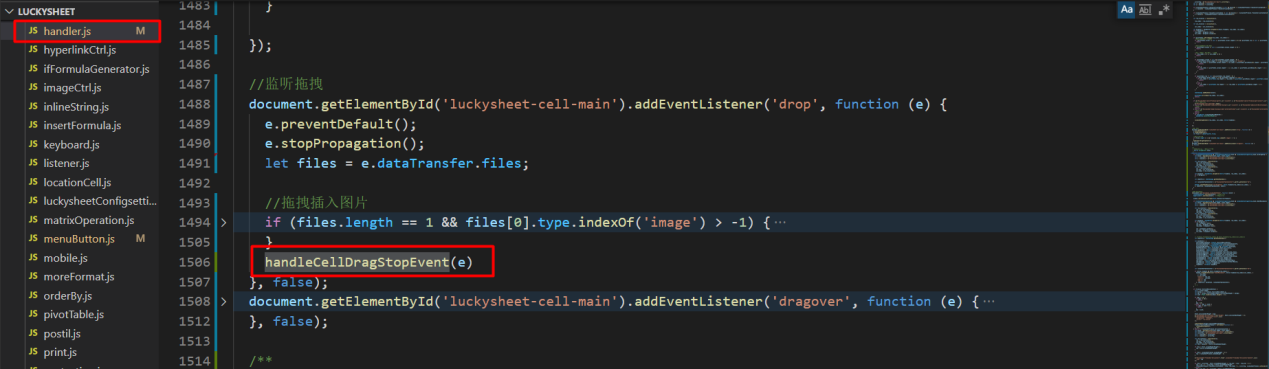


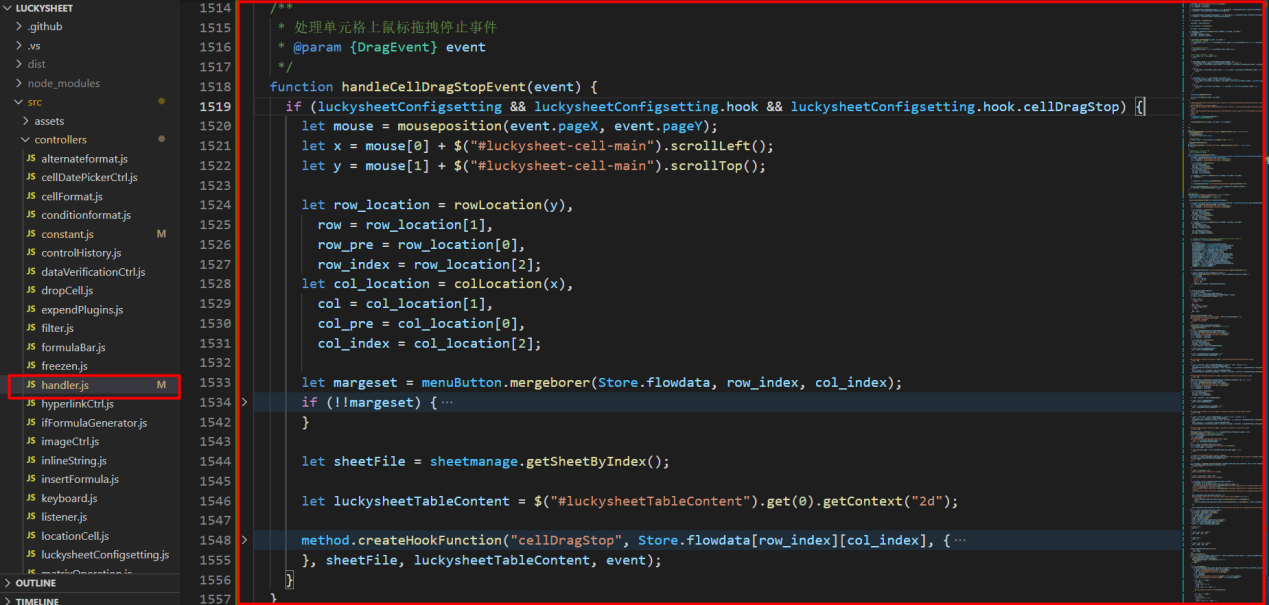
## 四、添加Luckysheet单元格上拖拽停止的事件

1）描述

需要拖拽数据源字段到Luckysheet，因此需要在Luckysheet上监听拖拽停止的事件，获取停止时鼠标所在的单元格。

2）JS代码





handleCellDragStopEvent(e)

/\*\*

   \* 处理单元格上鼠标拖拽停止事件

   \* @param {DragEvent} event

   \*/

  function handleCellDragStopEvent(event) {

    if (luckysheetConfigsetting && luckysheetConfigsetting.hook && luckysheetConfigsetting.hook.cellDragStop) {

      let mouse = mouseposition(event.pageX, event.pageY);

      let x = mouse[0] + $("#luckysheet-cell-main").scrollLeft();

      let y = mouse[1] + $("#luckysheet-cell-main").scrollTop();

      let row\_location = rowLocation(y),

        row = row\_location[1],

        row\_pre = row\_location[0],

        row\_index = row\_location[2];

      let col\_location = colLocation(x),

        col = col\_location[1],

        col\_pre = col\_location[0],

        col\_index = col\_location[2];

      let margeset = menuButton.mergeborer(Store.flowdata, row\_index, col\_index);

      if (!!margeset) {

        row = margeset.row[1];

        row\_pre = margeset.row[0];

        row\_index = margeset.row[2];

        col = margeset.column[1];

        col\_pre = margeset.column[0];

        col\_index = margeset.column[2];

      }

      let sheetFile = sheetmanage.getSheetByIndex();

      let luckysheetTableContent = $("#luckysheetTableContent").get(0).getContext("2d");

      method.createHookFunction("cellDragStop", Store.flowdata[row\_index][col\_index], {

        r: row\_index,

        c: col\_index,

        "start\_r": row\_pre,

        "start\_c": col\_pre,

        "end\_r": row,

        "end\_c": col

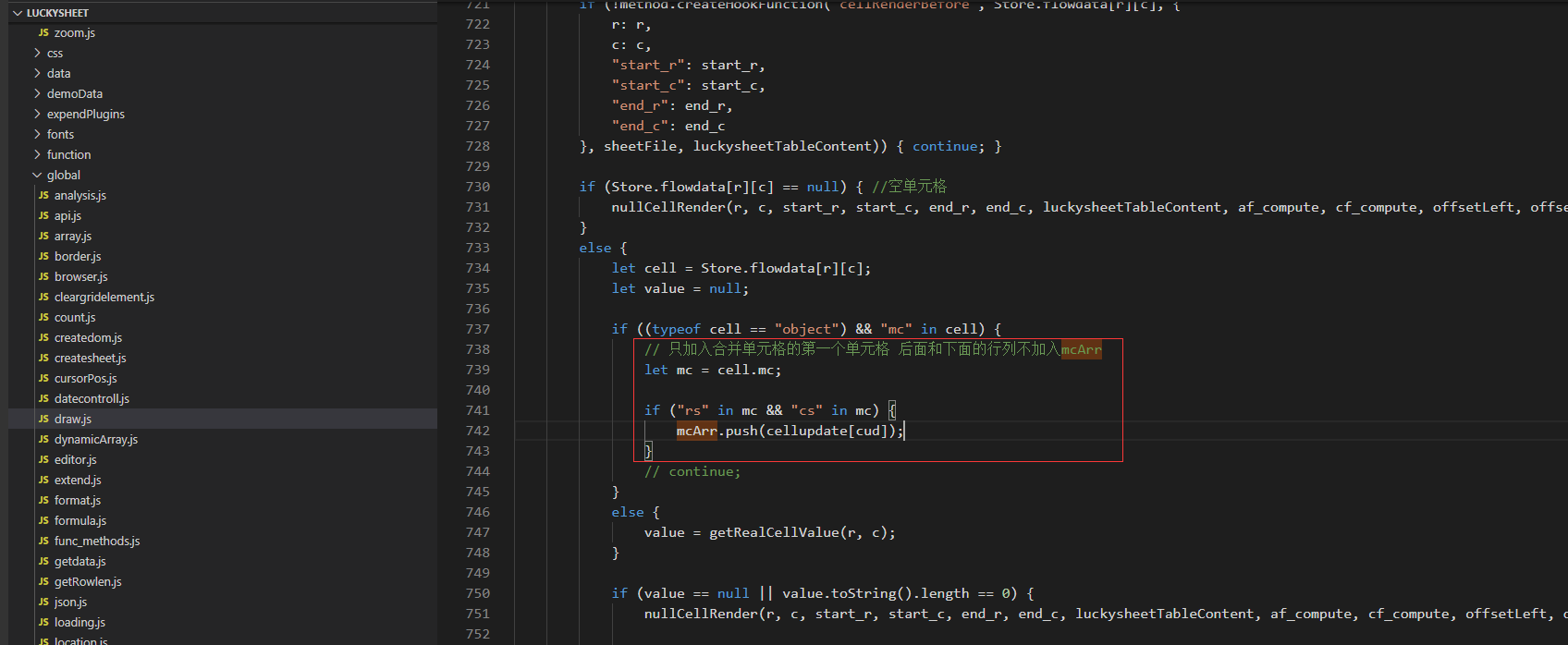
      }, sheetFile, luckysheetTableContent, event);

    }

  }

## 五 预览时，合并单元格返回数据后，报错

原因：代码中使用单元格内是否包含”mc”属性作为判断条件，几个单元格合并成一个单元格后，均会带有“mc”属性，只有第一行第一列的单元格带有“rs”、“cs”属性，所以加入判断条件，当“rs”、“cs”属性均存在时，才加入mcAry。

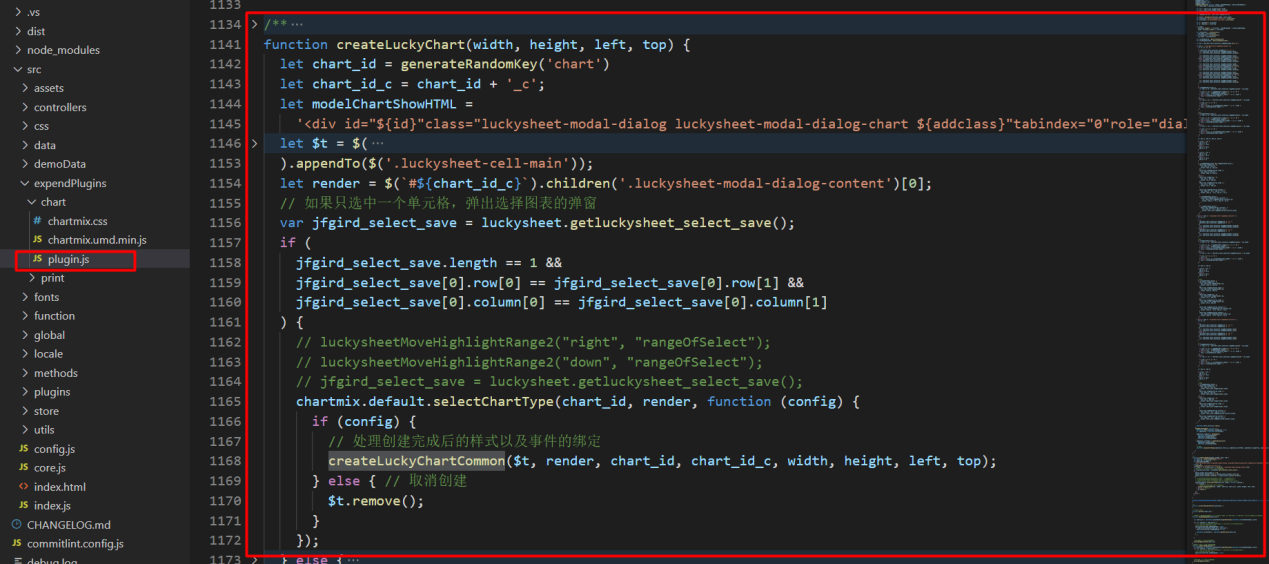


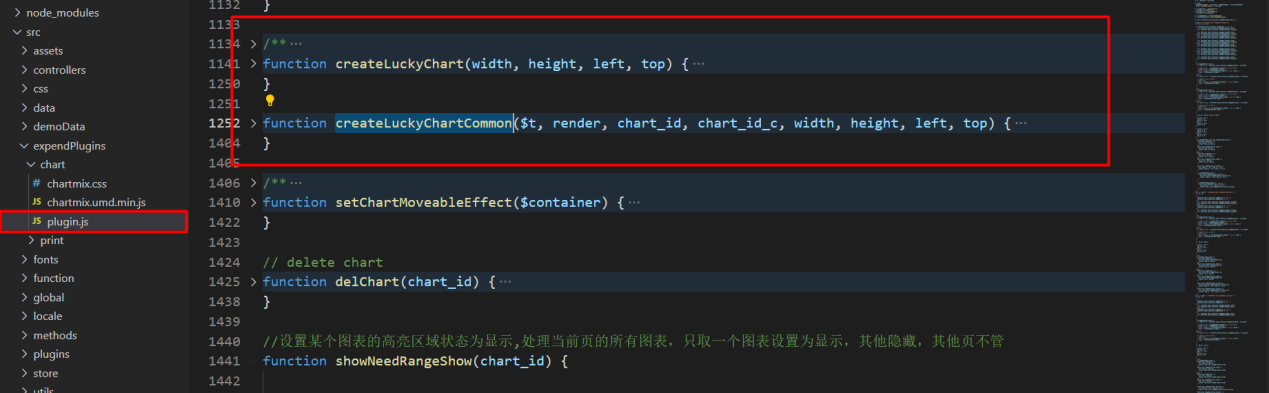
## 六 增加弹窗创建图表的回调

1）描述

改动较大。在只选中一个单元格时，使用弹窗方式选择创建图表的类型。否则按原来提供的创建表格方式进行创建。

2）代码





/\*\*

 \* 创建图表

 \* @param {number} width

 \* @param {number} height

 \* @param {number} left

 \* @param {number} top

 \*/

function createLuckyChart(width, height, left, top) {

  let chart\_id = generateRandomKey('chart')

  let chart\_id\_c = chart\_id + '\_c';

  let modelChartShowHTML =

    '<div id="${id}"class="luckysheet-modal-dialog luckysheet-modal-dialog-chart ${addclass}"tabindex="0"role="dialog"aria-labelledby=":41e"dir="ltr"><div class="luckysheet-modal-dialog-resize"><div class="luckysheet-modal-dialog-resize-item luckysheet-modal-dialog-resize-item-lt"data-type="lt"></div><div class="luckysheet-modal-dialog-resize-item luckysheet-modal-dialog-resize-item-mt"data-type="mt"></div><div class="luckysheet-modal-dialog-resize-item luckysheet-modal-dialog-resize-item-lm"data-type="lm"></div><div class="luckysheet-modal-dialog-resize-item luckysheet-modal-dialog-resize-item-rm"data-type="rm"></div><div class="luckysheet-modal-dialog-resize-item luckysheet-modal-dialog-resize-item-rt"data-type="rt"></div><div class="luckysheet-modal-dialog-resize-item luckysheet-modal-dialog-resize-item-lb"data-type="lb"></div><div class="luckysheet-modal-dialog-resize-item luckysheet-modal-dialog-resize-item-mb"data-type="mb"></div><div class="luckysheet-modal-dialog-resize-item luckysheet-modal-dialog-resize-item-rb"data-type="rb"></div></div><div class="luckysheet-modal-dialog-controll"><span class="luckysheet-modal-controll-btn luckysheet-modal-controll-update"role="button"tabindex="0"aria-label="修改图表"title="修改图表"><i class="fa fa-pencil"aria-hidden="true"></i></span><span class="luckysheet-modal-controll-btn luckysheet-modal-controll-max"role="butluckysheet\_chartIns\_indexton"tabindex="0"aria-label="最大化"title="最大化"><i class="fa fa-window-maximize"aria-hidden="true"></i></span><span class="luckysheet-modal-controll-btn luckysheet-modal-controll-del"role="button"tabindex="0"aria-label="删除"title="删除"><i class="fa fa-trash"aria-hidden="true"></i></span></div><div class="luckysheet-modal-dialog-content">${content}</div></div>'

  let $t = $(

    replaceHtml(modelChartShowHTML, {

      id: chart\_id\_c,

      addclass: 'luckysheet-data-visualization-chart',

      title: '图表生成',

      content: ''

    })

  ).appendTo($('.luckysheet-cell-main'));

  let render = $(`#${chart\_id\_c}`).children('.luckysheet-modal-dialog-content')[0];

  // 如果只选中一个单元格，弹出选择图表的弹窗

  var jfgird\_select\_save = luckysheet.getluckysheet\_select\_save();

  if (

    jfgird\_select\_save.length == 1 &&

    jfgird\_select\_save[0].row[0] == jfgird\_select\_save[0].row[1] &&

    jfgird\_select\_save[0].column[0] == jfgird\_select\_save[0].column[1]

  ) {

    // luckysheetMoveHighlightRange2("right", "rangeOfSelect");

    // luckysheetMoveHighlightRange2("down", "rangeOfSelect");

    // jfgird\_select\_save = luckysheet.getluckysheet\_select\_save();

    chartmix.default.selectChartType(chart\_id, render, function (config) {

      if (config) {

        // 处理创建完成后的样式以及事件的绑定

        createLuckyChartCommon($t, render, chart\_id, chart\_id\_c, width, height, left, top);

      } else { // 取消创建

        $t.remove();

      }

    });

  } else {

    //处理右边的空白单元格，自动略过并修改选区 ---------------start

    var shiftpositon\_row = -1;

    var row\_ed =

      jfgird\_select\_save[0]["row"][1] - jfgird\_select\_save[0]["row"][0];

    for (

      var r = jfgird\_select\_save[0]["row"][0]; r <= jfgird\_select\_save[0]["row"][1]; r++

    ) {

      for (

        var c = jfgird\_select\_save[0]["column"][0]; c <= jfgird\_select\_save[0]["column"][1]; c++

      ) {

        var value = getcellvalue(r, c, luckysheet.flowdata());

        //console.log("value,r,c",value,r,c);

        if (value != null && value.toString().length > 0) {

          shiftpositon\_row = r;

          break;

        }

      }

      if (shiftpositon\_row !== -1) {

        break;

      }

    }

    if (shiftpositon\_row == -1) {

      shiftpositon\_row = 0;

    }

    jfgird\_select\_save[0]["row"] = [shiftpositon\_row, shiftpositon\_row];

    luckysheet.setluckysheet\_select\_save(jfgird\_select\_save);

    chartInfo.luckysheet\_shiftpositon = $.extend(true, {}, jfgird\_select\_save[0]);

    luckysheetMoveEndCell("down", "range", false, row\_ed);

    jfgird\_select\_save = luckysheet.getluckysheet\_select\_save();

    var shiftpositon\_col = -1;

    var column\_ed =

      jfgird\_select\_save[0]["column"][1] - jfgird\_select\_save[0]["column"][0];

    for (

      var c = jfgird\_select\_save[0]["column"][0]; c <= jfgird\_select\_save[0]["column"][1]; c++

    ) {

      for (

        var r = jfgird\_select\_save[0]["row"][0]; r <= jfgird\_select\_save[0]["row"][1]; r++

      ) {

        var value = getcellvalue(r, c, luckysheet.flowdata());

        if (value != null && value.toString().length > 0) {

          shiftpositon\_col = c;

          break;

        }

      }

      if (shiftpositon\_col !== -1) {

        break;

      }

    }

    if (shiftpositon\_col == -1) {

      shiftpositon\_col = 0;

    }

    jfgird\_select\_save[0]["column"] = [shiftpositon\_col, shiftpositon\_col];

    luckysheet.setluckysheet\_select\_save(jfgird\_select\_save);

    chartInfo.luckysheet\_shiftpositon = $.extend(true, {}, jfgird\_select\_save[0]);

    luckysheetMoveEndCell("right", "range", false, column\_ed);

    jfgird\_select\_save = luckysheet.getluckysheet\_select\_save()

    var rangeArray = $.extend(true, [], jfgird\_select\_save);

    var rangeTxt = getRangetxt(chartInfo.currentSheetIndex, rangeArray[0], chartInfo.currentSheetIndex)

    let chartData = getdatabyselection();

    chartInfo.createChart(render, chartData, chart\_id, rangeArray, rangeTxt);

    createLuckyChartCommon($t, render, chart\_id, chart\_id\_c, width, height, left, top);

  }

}

function createLuckyChartCommon($t, render, chart\_id, chart\_id\_c, width, height, left, top) {

  let container = document.getElementById(chart\_id\_c)

  width = width ? width : 400

  height = height ? height : 250

  left = left ? left : 0

  top = top ? top : 0

  container.style.width = width + 'px'

  container.style.height = height + 'px'

  container.style.position = 'absolute'

  container.style.background = '#fff'

  container.style.left = left + 'px'

  container.style.top = top + 'px'

  render.style.width = '100%'

  render.style.height = '100%'

  container.style.zIndex = chartInfo.zIndex ? chartInfo.zIndex : 15

  chartInfo.zIndex++

  // insert chartinfo

  let sheetFile = chartInfo.luckysheetfile[getSheetIndex(chartInfo.currentSheetIndex)];

  if (!sheetFile.chart) {

    sheetFile.chart = [];

  }

  sheetFile.chart.push({

    chart\_id,

    width,

    height,

    left,

    top,

    sheetIndex: sheetFile.index

  })

  //处理区域高亮框参数，当前页中，只有当前的图表的needRangShow为true,其他为false

  showNeedRangeShow(chart\_id);

  // delete current chart

  $(`#${chart\_id}\_c .luckysheet-modal-controll-del`).click(function (e) {

    delChart(chart\_id)

  })

  setChartMoveableEffect($t);

  // edit current chart

  $(`#${chart\_id}\_c .luckysheet-modal-controll-update`).click(function (e) {

    showChartSettingComponent()

  })

  $t.children('.luckysheet-modal-dialog-content').mousedown(function (e) {

    if (!chartInfo.chartparam.luckysheetCurrentChartMaxState) {

      //当前图表显示区域高亮

      showNeedRangeShow(chart\_id);

    }

    e.stopPropagation()

  })

  $t.mousedown(function (e) { //move chart

      if (!chartInfo.chartparam.luckysheetCurrentChartMaxState) {

        //当前图表显示区域高亮

        showNeedRangeShow(chart\_id);

        setluckysheet\_scroll\_status(true);

        //允许拖动渲染框

        if (

          !$(e.target).is(".luckysheet-modal-dialog-controll") &&

          !$(e.target).is(".luckysheet-modal-controll-btn") &&

          !$(e.target).is("i")

        ) {

          // Debounce

          chartInfo.chartparam.luckysheetCurrentChartMoveTimeout = setTimeout(

            function () {

              chartInfo.chartparam.luckysheetCurrentChartMove = true;

            },

            100

          );

        }

        var toffset = chartInfo.chartparam.luckysheetCurrentChartMoveObj.offset();

        var tpsition = chartInfo.chartparam.luckysheetCurrentChartMoveObj.position();

        //luckysheetCurrentChartMoveXy: [鼠标点相对chart框的距离X方向，鼠标点相对chart框的距离Y方向，chart框相对cell-main的距离X方向，chart框相对cell-main的距离Y方向，水平滚动条的位置，垂直滚动条的位置]

        chartInfo.chartparam.luckysheetCurrentChartMoveXy = [

          e.pageX - toffset.left,

          e.pageY - toffset.top,

          tpsition.left,

          tpsition.top,

          $("#luckysheet-scrollbar-x").scrollLeft(),

          $("#luckysheet-scrollbar-y").scrollTop()

        ];

        chartInfo.chartparam.luckysheetCurrentChartMoveWinH = $(

          "#luckysheet-cell-main"

        )[0].scrollHeight;

        chartInfo.chartparam.luckysheetCurrentChartMoveWinW = $(

          "#luckysheet-cell-main"

        )[0].scrollWidth;

        if (

          !$(e.target).hasClass("luckysheet-mousedown-cancel") &&

          $(e.target).filter("[class\*='sp-palette']").length == 0 &&

          $(e.target).filter("[class\*='sp-thumb']").length == 0 &&

          $(e.target).filter("[class\*='sp-']").length == 0

        ) {

          $("#luckysheet-rightclick-menu").hide();

          $("#luckysheet-cols-h-hover").hide();

          $("#luckysheet-cols-menu-btn").hide();

          $("#luckysheet-rightclick-menu").hide();

          $(

            "#luckysheet-sheet-list, #luckysheet-rightclick-sheet-menu, #luckysheet-user-menu"

          ).hide();

          $(

            "body > .luckysheet-filter-menu, body > .luckysheet-filter-submenu, body > .luckysheet-cols-menu"

          ).hide();

        }

        e.stopPropagation();

      }

    }).find(".luckysheet-modal-dialog-resize-item")

    .mousedown(function (e) {

      if (chartInfo.chartparam.luckysheetCurrentChartActive) {

        chartInfo.chartparam.luckysheetCurrentChartResize = $(this).data("type"); //开始状态resize

        var mouse = mouseposition(e.pageX, e.pageY),

          scrollLeft = $("#luckysheet-scrollbar-x").scrollLeft(),

          scrollTop = $("#luckysheet-scrollbar-y").scrollTop();

        var x = mouse[0] + scrollLeft;

        var y = mouse[1] + scrollTop;

        var position = chartInfo.chartparam.luckysheetCurrentChartResizeObj.position();

        //参数：x,y:鼠标位置，$t.width(), $t.height(): chart框宽高， position.left + scrollLeft, position.top + scrollTop ：chart框位置 ，scrollLeft, scrollTop：滚动条位置

        chartInfo.chartparam.luckysheetCurrentChartResizeXy = [

          x,

          y,

          $t.width(),

          $t.height(),

          position.left + scrollLeft,

          position.top + scrollTop,

          scrollLeft,

          scrollTop

        ];

        chartInfo.chartparam.luckysheetCurrentChartResizeWinH = $(

          "#luckysheet-cell-main"

        )[0].scrollHeight;

        chartInfo.chartparam.luckysheetCurrentChartResizeWinW = $(

          "#luckysheet-cell-main"

        )[0].scrollWidth;

        chartInfo.chartparam.luckysheetCurrentChart = chart\_id;

        e.stopPropagation();

      }

    })

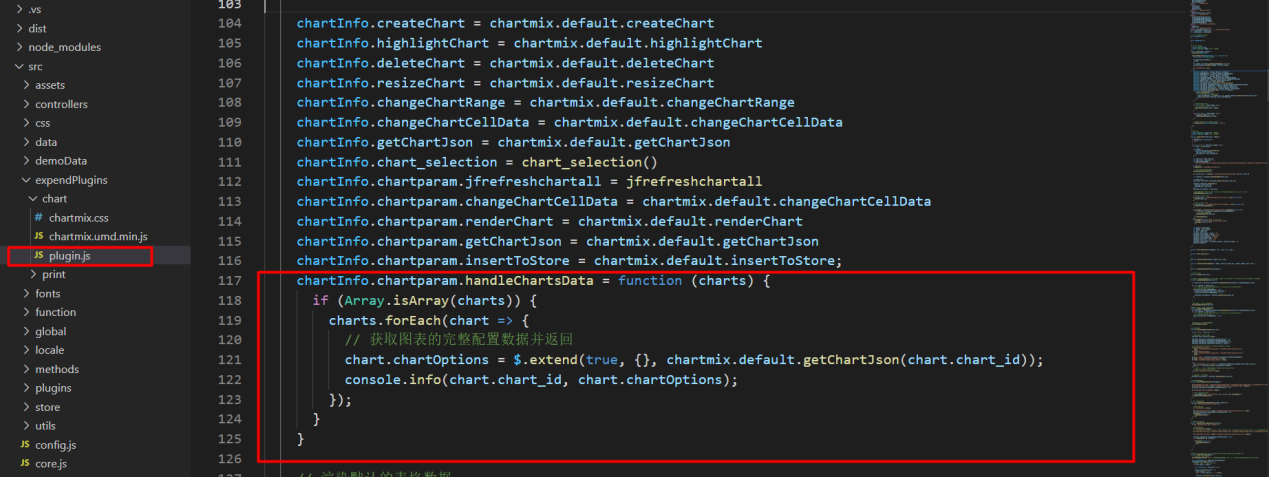
}

## 七 完善全图表配置信息的获取

1）描述

在图表插件plugin.js中添加相应的处理逻辑，然后在getAllSheets中调用这个方法对表格返回数据进行处理。

2）代码1



chartInfo.chartparam.handleChartsData = function (charts) {

      if (Array.isArray(charts)) {

        charts.forEach(chart => {

          // 获取图表的完整配置数据并返回

          chart.chartOptions = $.extend(true, {}, chartmix.default.getChartJson(chart.chart\_id));

          console.info(chart.chart\_id, chart.chartOptions);

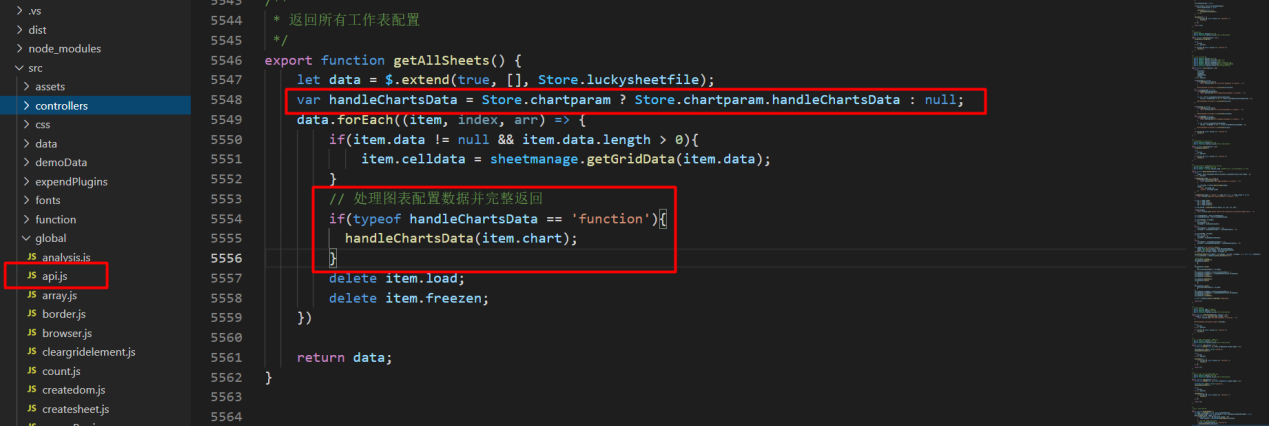
        });

      }

    }

2）代码2

var handleChartsData = Store.chartparam ? Store.chartparam.handleChartsData : null;



var handleChartsData = Store.chartparam ? Store.chartparam.handleChartsData : null;

// 处理图表配置数据并完整返回

        if(typeof handleChartsData == 'function'){

          handleChartsData(item.chart);

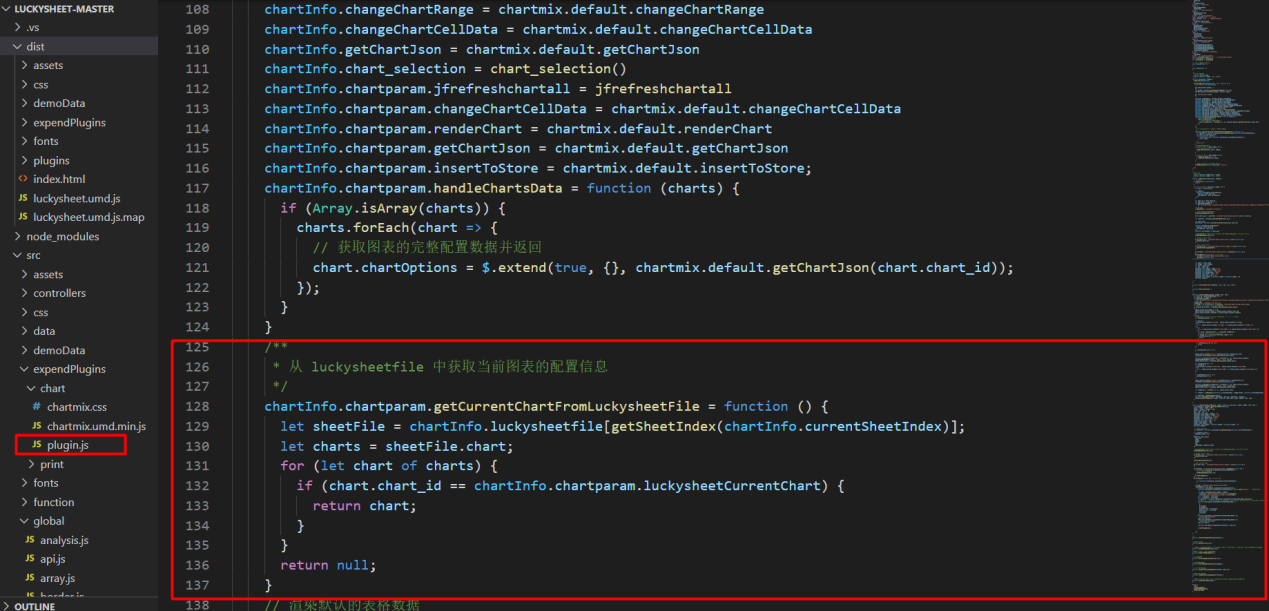
        }

## 八 完善图表拖动或改变大小后更新图表位置和大小信息

1）描述

图表发生拖动或大小发生变化时，Luckysheet没有将最新的大小和位置信息更新到缓存的图标信息中，在这新增这一步操作。

2）获取当前图表配置信息



/\*\*

     \* 从 luckysheetfile 中获取当前图表的配置信息

     \*/

    chartInfo.chartparam.getCurrentChartFromLuckysheetFile = function () {

      let sheetFile = chartInfo.luckysheetfile[getSheetIndex(chartInfo.currentSheetIndex)];

      let charts = sheetFile.chart;

      for (let chart of charts) {

        if (chart.chart\_id == chartInfo.chartparam.luckysheetCurrentChart) {

          return chart;

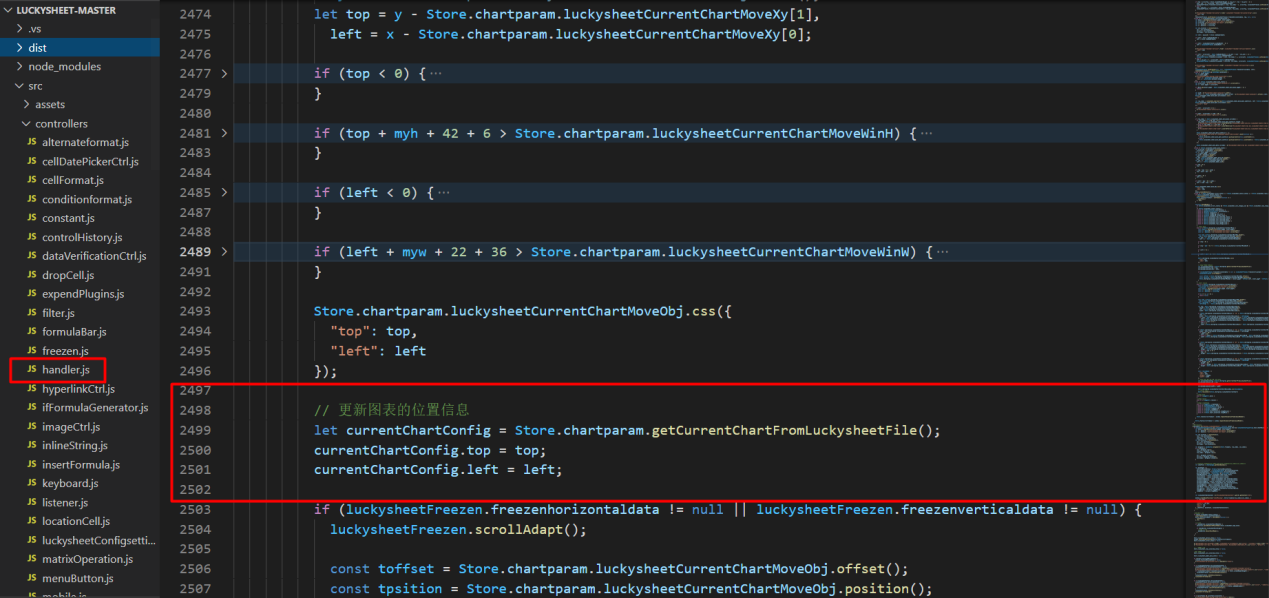
        }

      }

      return null;

    }

3）更新图表位置代码



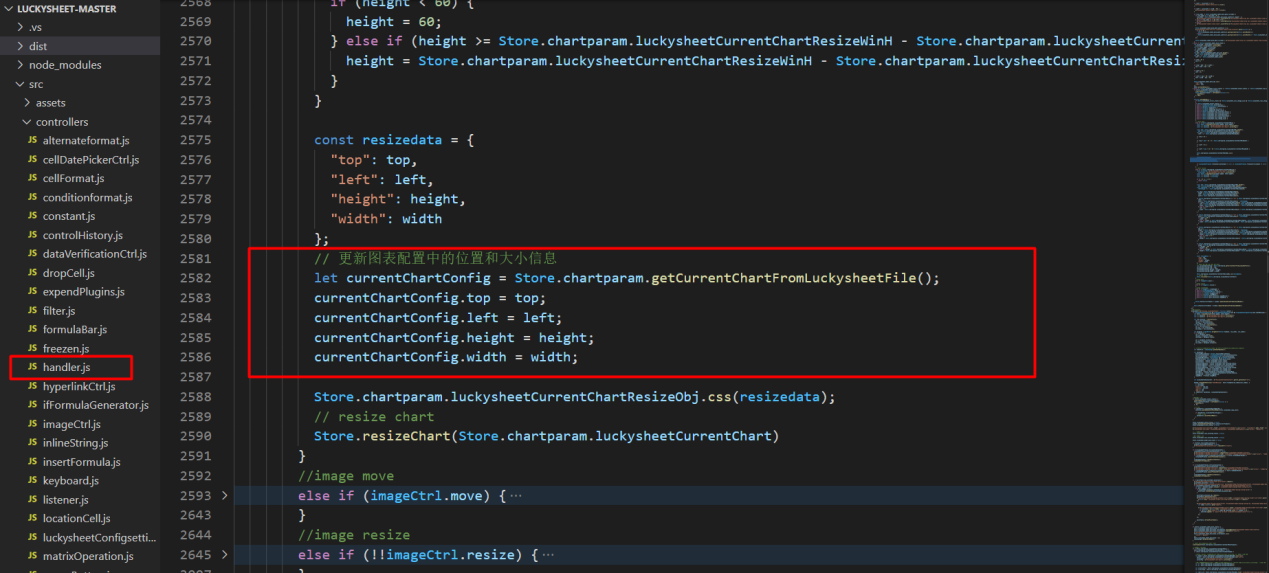
// 更新图表的位置信息

          let currentChartConfig = Store.chartparam.getCurrentChartFromLuckysheetFile();

          currentChartConfig.top = top;

          currentChartConfig.left = left;

3）更新图表大小代码



// 更新图表配置中的位置和大小信息

          let currentChartConfig = Store.chartparam.getCurrentChartFromLuckysheetFile();

          currentChartConfig.top = top;

          currentChartConfig.left = left;

          currentChartConfig.height = height;

          currentChartConfig.width = width;

## 九 添加设置sheet表配置hyperlink属性的方法

1. 对应sheet表配置的属性



2)js调用样例



3)实现代码

