**内管综合管理系统前端设计开发文档**

**文件描述**

|  |  |
| --- | --- |
| **文件名称** | 综合管理系统前端设计开发文档 |
| **文档编号** |  |
| **当前版本号** | V1.0 |

**文档变更**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** | **审核** |
| 20201118 | V1.0 |  | 周可 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1 引言 5](#_Toc23240)

[1.1 编写说明 5](#_Toc15198)

[1.2 适用范围 5](#_Toc25626)

[2 准备工作 5](#_Toc19299)

[2.1 项目结构 5](#_Toc28224)

[2.2 界面路由 5](#_Toc17394)

[3 代码设计 7](#_Toc28334)

[3.1 控件初始化 7](#_Toc30450)

[3.1.1 文本输入框 7](#_Toc24810)

[3.1.2 日期选择框 7](#_Toc1576)

[3.1.3 时间选择框 7](#_Toc27315)

[3.1.4 下拉选择框 8](#_Toc1842)

[3.1.5 文件选择框 9](#_Toc16000)

[3.1.6 按钮事件 10](#_Toc28832)

[3.2 执法机构查控功能程序 10](#_Toc5843)

[4 数据库表设计 11](#_Toc24687)

[5 业务功能概设 12](#_Toc14638)

[5.1 请求ZIP下载 12](#_Toc4013)

[5.1.1 功能概述 12](#_Toc9646)

[5.1.2 流程图示 13](#_Toc9246)

[5.1.3 流程说明 13](#_Toc161)

[5.1.4 实现要点 13](#_Toc22545)

[5.2 止付类 14](#_Toc28040)

[5.2.1 功能概述 14](#_Toc29123)

[5.2.2 流程图示 15](#_Toc3938)

[5.2.3 流程说明 15](#_Toc3847)

[5.2.4 实现要点 16](#_Toc16446)

[5.3 冻结类 16](#_Toc12049)

[5.3.1 功能概述 16](#_Toc25186)

[5.3.2 流程图示 17](#_Toc29575)

[5.3.3 流程说明 17](#_Toc32639)

[5.3.4 实现要点 18](#_Toc6994)

[5.4 划拨交易 18](#_Toc20182)

[5.4.1 功能概述 18](#_Toc3269)

[5.4.2 流程图示 19](#_Toc16696)

[5.4.3 流程说明 20](#_Toc12809)

[5.4.4 实现要点 20](#_Toc27204)

[5.5 账户信息查询类 20](#_Toc16891)

[5.5.1 功能概述 20](#_Toc6224)

[5.5.2 流程图示 21](#_Toc8406)

[5.5.3 流程说明 21](#_Toc14558)

[5.5.4 实现要点 21](#_Toc8040)

[5.6 影像资料推送影像平台 22](#_Toc21910)

[5.6.1 功能概述 22](#_Toc21659)

[5.6.2 流程图示 22](#_Toc5648)

[5.6.3 流程说明 23](#_Toc27800)

[5.6.4 实现要点 23](#_Toc21752)

[5.7 参数配置 23](#_Toc25053)

[5.7.1 功能概述 23](#_Toc2901)

[5.7.2 流程图示 24](#_Toc30695)

[5.7.3 流程说明 24](#_Toc10997)

[5.7.4 实现要点 24](#_Toc19565)

[5.8 待审核信息提醒 25](#_Toc2382)

[5.8.1 功能概述 25](#_Toc15169)

[5.8.2 流程图示 25](#_Toc18005)

[5.8.3 流程说明 25](#_Toc29255)

[5.8.4 实现要点 26](#_Toc14101)

[6 接口设计 26](#_Toc15915)

[6.1 外部接口设计 26](#_Toc23063)

[6.2 内部接口设计 26](#_Toc7610)

1. 引言

## 编写说明

本文档简单阐述了综合管理系统前端部分对表格的系列操作，主要包括对表格数据的增，删，查，改操作。

## 适用范围

本文档的面向综合管理系统中前端开发技术人员。

# 准备工作

## 项目结构

1. 前端界面jsp文件存放地址路径:\icore-sump\systems\sump\src\main\webapp\WEB-INF\jsp\
2. 前端界面对应的静态文件(js文件/图片/样式等静态文件)存放地址路径:\icore-sump\systems\sump\src\main\webapp\pages\

## 界面路由

由于界面的路由是存储在数据库中的，所以需要在数据库表中插入对应的路由信息:

-- 添加主菜单sql

INSERT INTO `sump`.`sif\_sys\_auth`(`auth\_cd`, `auth\_type`, `register\_cd`, `auth\_url`, `iconfg`, `menu\_name`, `parent\_auth\_cd`, `rank`, `show\_flag`, `sortno`, `target\_flag`) VALUES ('1009', '2', '999', NULL, NULL, '司法查冻控', NULL, 1, '1', 9, NULL);

-- 添加一级菜单

INSERT INTO `sump`.`sif\_sys\_auth`(`auth\_cd`, `auth\_type`, `register\_cd`, `auth\_url`, `iconfg`, `menu\_name`, `parent\_auth\_cd`, `rank`, `show\_flag`, `sortno`, `target\_flag`) VALUES ('100901', '2', '999', NULL, NULL, '账户查询', '1009', 2, '1', 1, NULL);

-- 添加二级菜单

INSERT INTO `sump`.`sif\_sys\_auth`(`auth\_cd`, `auth\_type`, `register\_cd`, `auth\_url`, `iconfg`, `menu\_name`, `parent\_auth\_cd`, `rank`, `show\_flag`, `sortno`, `target\_flag`) VALUES ('10

090101', '2', '999', '/judicial/imageQuery', NULL, '协助调阅凭证图像结果', '100901', 3, NULL, NULL, NULL);

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 列名 | auth\_cd | auth\_type | register\_cd | auth\_url | iconfg | menu\_name |
| 说明 | 菜单id | 菜单类型 | 法人号 | 路由地址 |  | 菜单名称 |
| 列名 | parent\_auth\_cd | rank | show\_flag | sortno | target\_flag |  |
| 说明 | 父级菜单 | 菜单级别 |  | 排序 |  |  |

注：auth\_url(路由地址) 需要对应jsp界面文件存放路径下的路径。例如路由地址为/judicial/imageQuery，对应的jsp文件应为\icore-sump\systems\sump\src\main\webapp\WEB-INF\jsp\judicial\imageQuery.jsp

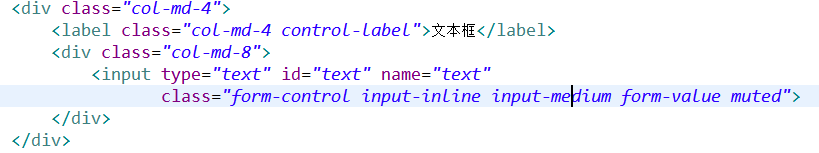
-- 菜单权限配置

INSERT INTO `sump`.`sif\_sys\_role\_auth`(`auth\_cd`, `auth\_type`, `register\_cd`, `role\_cd`) VALUES ('10090101', '2', '999', '管理员');

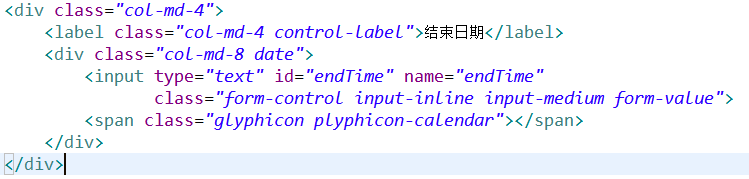
# 代码设计

## 控件初始化

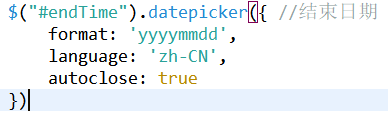
### 文本输入框



### 日期选择框

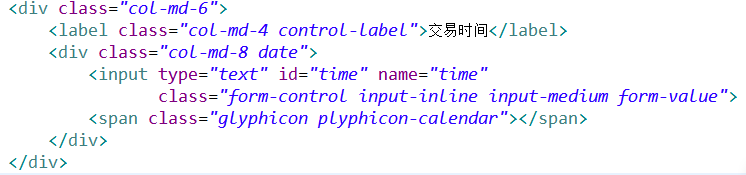


Js:



注：更多日期框组件选项及说明请到官网查询。

### 时间选择框

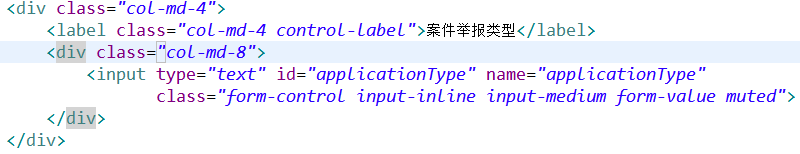


Js:

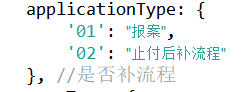


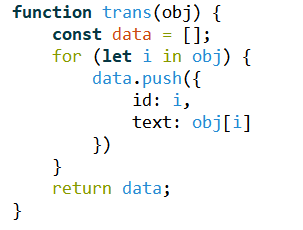
注：更多时间框组件选项及说明请到官网查询。

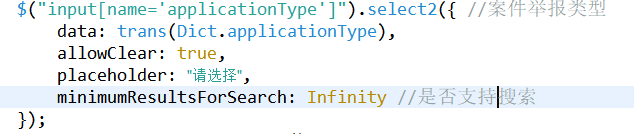
### 下拉选择框



Js:

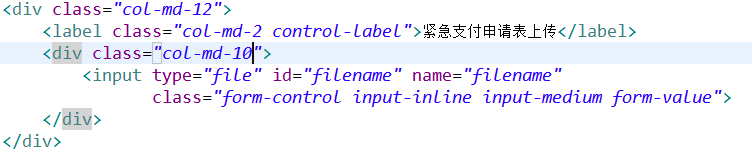






注：更多下拉组件选项及说明请到官网查询。

### 文件选择框



Js:



注：更多文件选择框组件选项及说明请到官网查询。

### 按钮事件



## 表格

### 表格定义



### 查询表格

初始化：





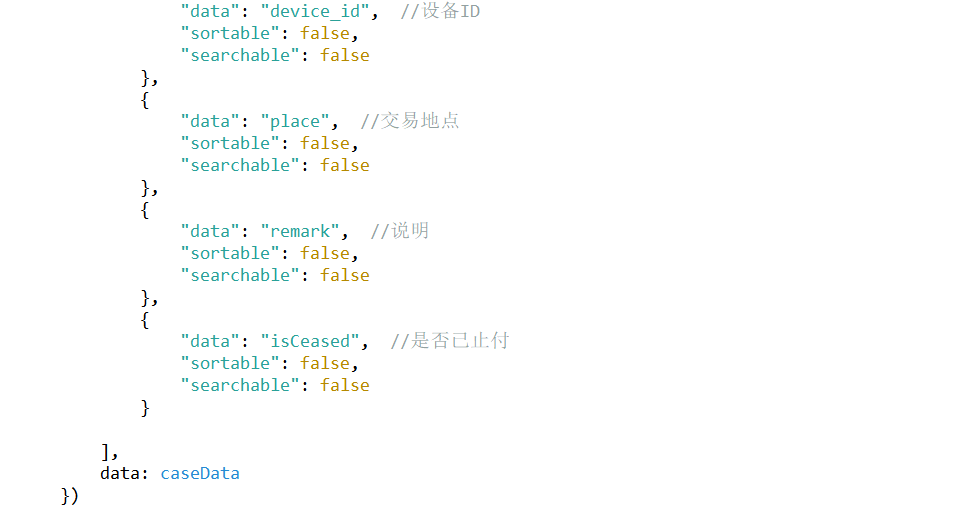


### 自定义表格

初始化：

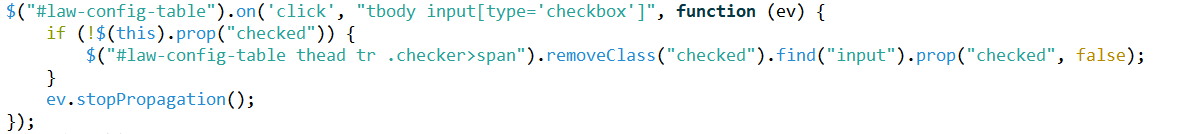




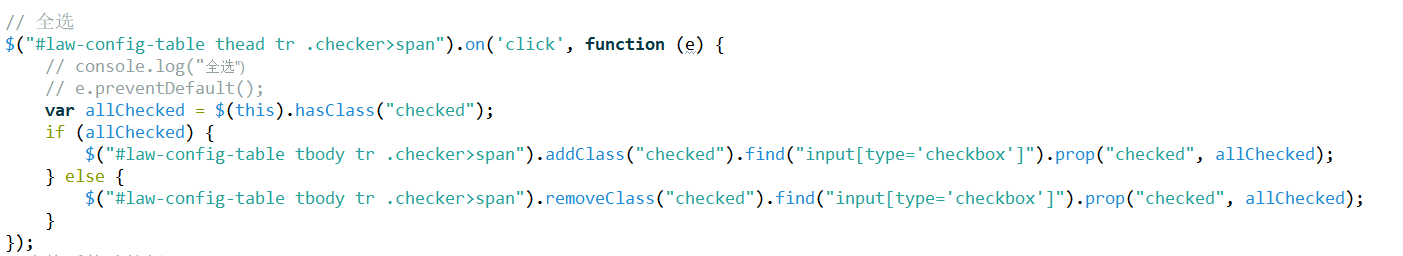


### 表格事件

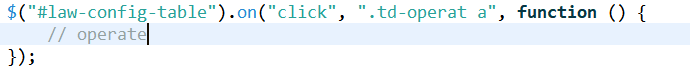
给表格第一列复选框添加点击效果：



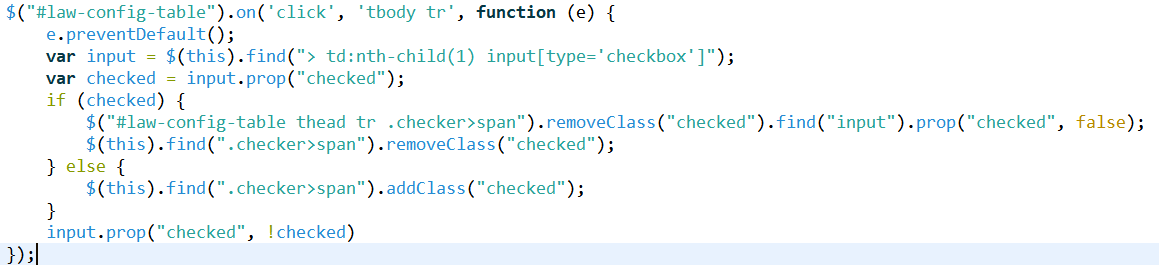
复选框全选：



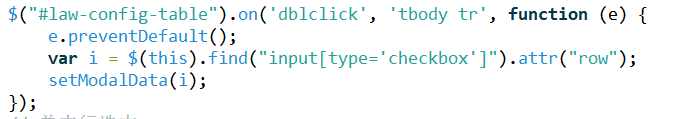
表格最后一列操作按钮事件：



单击行选中此行：

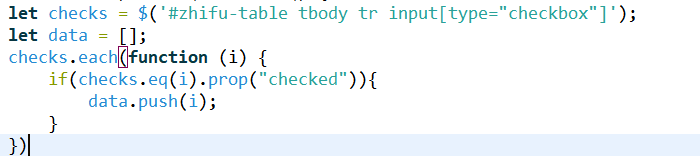


双击行事件：



### 表格其他操作

获取选中的行：



快速获取选中的行的数量：



自定义表格重新加载数据并刷新表格：



设置查询表格的额外参数：



重新查询并刷新表格：



根据行号获取该行数据：



# 其他

## Toast轻提示

定义：



使用：





效果：

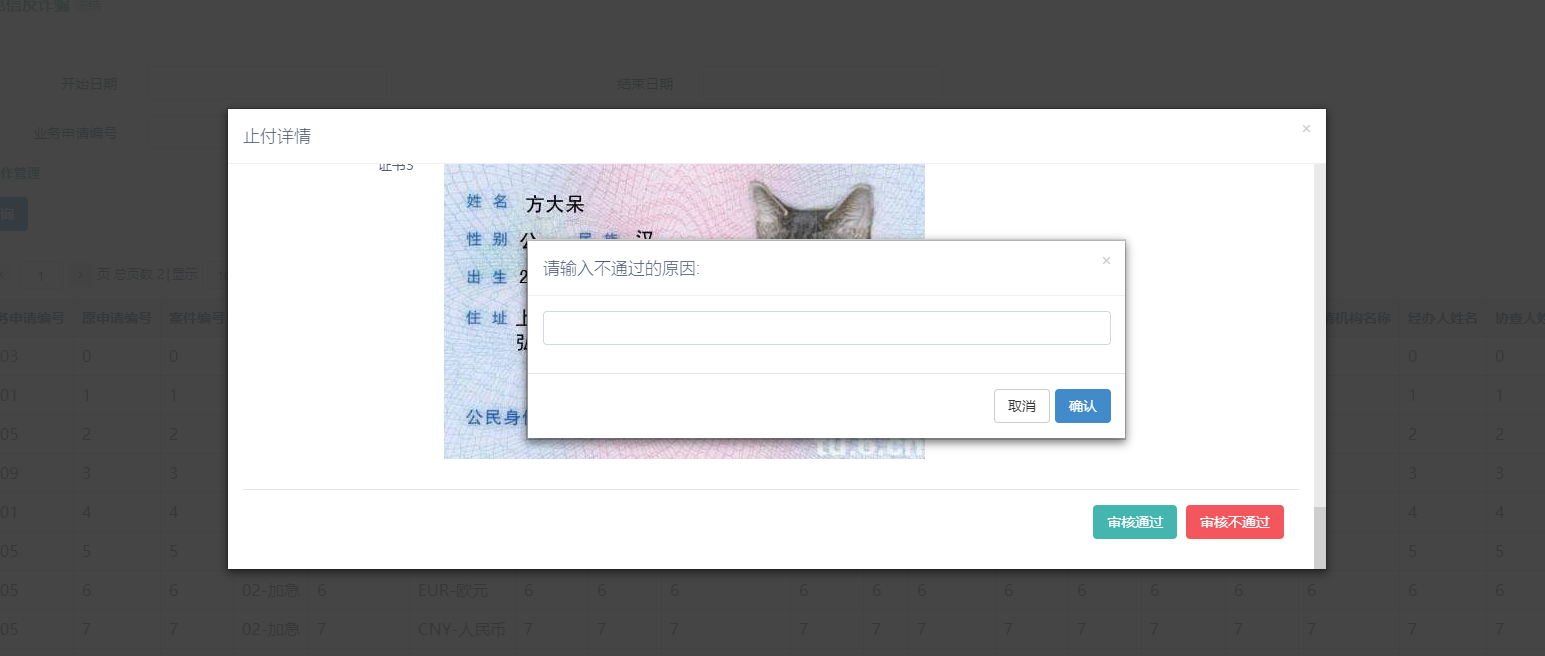


## Prompt对话框

使用：



效果：

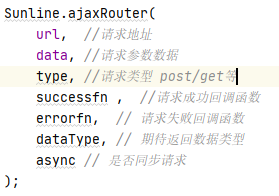


## Ajax请求

一般先通过ajaxPath方法获取请求地址：



再通过ajaxRouter发送请求：



## 读取xlsx表格

### 插件概述

由SheetJS出品的js-xlsx是一款非常方便的只需要纯JS即可读取和导出excel的工具库，功能强大，支持格式众多，支持xls、xlsx、ods(一种OpenOffice专有表格文件格式)等十几种格式。主要为xlsx表格。

官方github：<https://github.com/SheetJS/js-xlsx>

### 使用步骤

1.插件xlsx,存放的路径为/sump/src/main/webapp/assets/global/plugins/xlsx/xlsx.core.min.js

2.jsp页面导入插件:

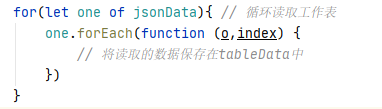


3.读取文件



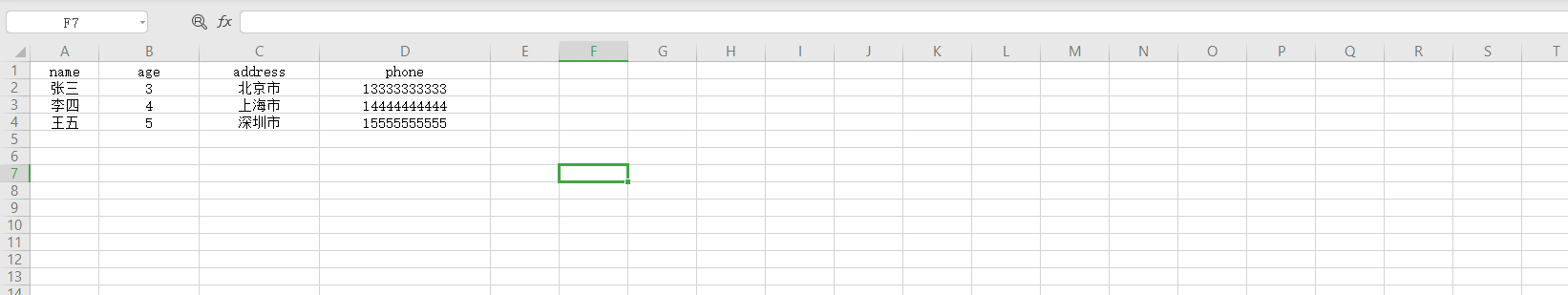
### 注意事项

通过以上方法读取xlsx文件，在

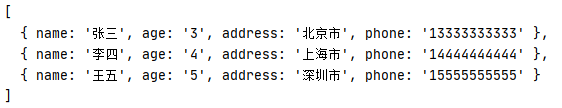


这个循环中,one为工作表解析后的数据，参数o为每行键值对，index为除表头的行数。例如：

工作表数据为：



读取后one数组为:



# 优化

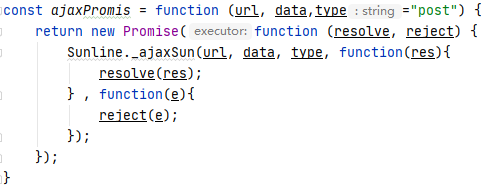
## Ajax请求优化

### 概述

由于ajaxRouter请求方法的参数较多，回调函数可能复杂多变，容易在写法上造成代码错乱冗余，因此，我改写封装了一个Promise方法，并可以配合async/await语法，可以极大改善这种情况。

### 代码示例

方法定义：



方法使用：

