物业管理系统V1.0源代码

北京杰迅鸿翔信息技术有限公司

<template>

<div class="client-window">

<p>修改客户信息<button item class="right" @click="clientwindowoff">x</button></p>

<div class="cli-item-body">

<div class="cli-item">公司名称<input type="text" class="cli-ipt Bipt right" disabled="disabled" v-model="cli\_window\_text.corporateName"></div>

<div class="cli-item">

<div class="cli-item-box left">合同号<input type="text" class="cli-ipt right" disabled="disabled" v-model="cli\_window\_text.contractNumber"></div>

<div >入住日期<input type="date" class="cli-ipt right" v-model="cli\_window\_text.housingDate"></div>

</div>

<div class="cli-item addClone" v-for="(item,index) in rooms" :key="index">

<div class="cli-item">

<div class="cli-item-box left">

楼宇名称<select class="cli-sele right" disabled="disabled">

<option>{{item.floorName}}</option>

</select>

</div>

<div>

楼层<select class="cli-sele right" disabled="disabled">

<option>{{item.storey}}</option>

</select>

</div>

</div>

<div class="cli-item-box left">

房间号<input type="text" class="cli-ipt right" disabled="disabled" :value=item.roomNumber>

</div>

<div class="">

面积<input type="text" class="cli-ipt right" disabled="disabled" :value=item.area>

</div>

</div>

<div class="cli-item">

<div class="cli-item-box left">

联系人<input type="text" class="cli-ipt right" v-model="cli\_window\_text.contactName">

</div>

<div>

电话<input type="text" class="cli-ipt right" v-model="cli\_window\_text.phoneNumber">

</div>

</div>

<div class="cli-item">

<div class = "cli-item-box left">

邮箱地址<input type="text" class="cli-ipt right" v-model="cli\_window\_text.mailbox">

</div>

<div>

注册状态<select class="cli-sele right" v-model="cli\_window\_text.registrationStatus">

<option value="YES">是</option>

<option value="NO">否</option>

</select>

</div>

</div>

<div class="cli-item">

<button @click="updateClientInfo">修改</button>

<button @click="clientwindowoff">取消</button>

</div>

</div>

</div>

</template>

<script>

export default {

name: 'ClientWindow',

props:{

cli\_window\_text:Object

},

data () {

return {

isAdd:false,

clientwindowisoff:false,

rooms:this.cli\_window\_text.rentRoomSet,

}

},

components: {

},

methods:{

//以下是网络请求方法

updateClientInfo() { //点击根据ID查找客户信息

let ClientChangeParams={

contactName:this.cli\_window\_text.contactName, //联系人

housingDate:this.cli\_window\_text.housingDate, //入住日期

mailbox:this.cli\_window\_text.mailbox, //邮箱

phoneNumber:this.cli\_window\_text.phoneNumber, //电话

registrationStatus:this.cli\_window\_text.registrationStatus,//注册状态

}

this.$ajax.jsonpost("/api/merchant/"+this.cli\_window\_text.id, ClientChangeParams).then((res)=>{

console.log(res);

this.$router.go(0)

}).catch(err=>{

console.log(err)

})

},

//以下是本页面的方法

clientwindowoff(){//点击关闭弹窗

this.$emit("turnoff", this.clientwindowisoff)

}

}

}

</script>

<style scoped>

@import "../../../../assets/css/reset.css";

.client-window{

background: #fff;

position: absolute;

top: 16%;

left: 30%;

border: 1px solid #000;

position: absolute;

z-index: 3;

background:white;

}

P{

background: #81B845;

color: white;

padding:2px 8px;

margin-bottom: 25px;

border: none;

}

p button{

width: 20px;

color: white;

background: none;

}

.cli-item-body{

padding: 0px 45px;

}

.cli-item{

margin-bottom: 18px;

}

.cli-item:last-of-type{

clear: both;

display: block;

text-align: center;

}

.cli-item:last-of-type button{

width: 65px;

height: 26px;

background: white;

margin: 15px 0 0 30px;

border: 1px solid rgb(201, 197, 197);

}

.cli-item:last-of-type button:first-of-type{

background: #81B845;

color: #fff;

}

.cli-ipt{

width: 123px;

height: 20px;

}

.cli-sele{

width: 124px;

height: 22px;

}

.Bipt{

width: 350px;

height: 20px;

margin-left: 17px;

}

.cli-item-box{

width: 200px;

margin-right: 35px;

}

</style>

<template>

<div class="detail\_window">

<p><span class="iconfont icon-close" @click="turnoff"></span></p>

<Table :columns="columns1" :data="data1" class="table"></Table>

</div>

</template>

<script>

export default {

name: 'DetailWindow',

props:{

DetailWindowShowList:Object,

},

data () {

return {

columns1:[

{

title:'合同号',

key:'contractNumber'

},

{

title:'公司名称',

key:'corporateName'

},

{

title:'经办人',

key:'operatorName'

},

{

title:'联系电话',

key:'phoneNumber'

},

{

title:'租金',

key:'rent'

},

{

title:'保证金',

key:'deposit'

},

{

title:'付款方式',

key:'payMoneyWayString'

},

{

title:'起租日期',

key:'startDate'

},

{

title:'结束日期',

key:'endDate'

},

{

title:'总金额',

key:''

},

{

title:'月租金',

key:''

},

{

title:'电费/月',

key:'electricCharge'

}

],

data1: [

this.DetailWindowShowList

],

DetailWindowOffBoo:false

}

},

components:{

},

methods:{

turnoff(){

this.$emit("DetailWindowOff",this.DetailWindowOffBoo)

}

}

}

</script>

<style scoped>

@import "../../../../assets/font/iconfont.css";

.ivu-table-wrapper{

height: auto;

padding: 0 0 20px 0;

}

.detail\_window{

border: 1px solid #000;

background: #fff;

width: 1500px;

position: absolute;

top: calc(50% - 131px );

left: calc(50%-750px);

}

.icon-close{

font-size: 23px;

float: right;

cursor: pointer;

}

.table{

clear: both;

}

</style>

<template>

<div class="box">

<Options class="options" :isJurisdiction="isJurisdiction">

<button slot="other" class="options-item" @click="ClientExport">导出Excel</button>

<div slot="other">

<span>显示条数：</span>

<input slot="other" type="text" style="width:40px;margin: 0 10px 0 0" v-model="clientfinddata.pageSize" @change="clifind">

</div>

<input type="text" slot="other" class="options-item" placeholder="公司名称关键词" v-model="clientfinddata.corporateName">

<button slot="other" class="options-item" @click="clifind">搜索</button>

</Options>

<div style="width:100%;height:640px;overflow: auto;">

<List :message = "msg"

:sonmenu="fathermenu"

@lisgcheangeshow="Clientshow"

:isDeleteShowBoo="isDeleteShowBoo"

@saveDownloadId="saveDownloadId"

ref="dataList"

:isJurisdiction="isJurisdiction"

></List>

</div>

<ClientWindow v-if="isAdd" @turnoff="turnoff" :cli\_window\_text="cli\_window\_text" ></ClientWindow>

<DetailWindow v-if="DetailWindowIsShow" :DetailWindowShowList="DetailWindow\_text" @DetailWindowOff="DetailWindowOff"></DetailWindow>

<RoomsWindow v-if="DetailRoomsWindowIsShow" :rentRoomSet='rentRoomSet' :DetailRoomscorporateName='DetailRoomscorporateName' @RoomsWindowOff= "RoomsWindowOff"></RoomsWindow>

<Page ref="pages" :total=totalPages size="small" id="cli-page" @on-change="changepage"/>

</div>

</template>

<script>

import Options from "../../components/options/Options"

import List from "../../components/list/List"

import ClientWindow from "./childs/window/ClientWindow"

import DetailWindow from "./childs/window/DetailWindow"

import RoomsWindow from "../../components/rooms-window/RoomsWindow"

export default {

name:"Client",

data(){

return{

isAdd:false,

msg:[],

fathermenu: [],

clientfinddata:{

corporateName:null,

pageNo:1,

pageSize:15

},

isAddtatle: "新增",

totalPages:10,

isDeleteShowBoo:false,

clientFindId:null,

cli\_window\_text:{},

rentRoomSet:[],

DetailWindowIsShow:false,

DetailWindow\_text:{},

DetailWindowFindcontractNumber:null,

DetailRoomsWindowIsShow:false,

DetailRoomscorporateName:'',

DownloadId:[],

isJurisdiction:false,

roleName:sessionStorage.getItem("roleName"),

}

},

created(){

this.clientInterList()

this.roleName = sessionStorage.getItem("roleName")

if(this.roleName == "系统管理员"||this.roleName == "物业人员"){

this.isJurisdiction = true

this.fathermenu = [

{

type: 'selection',

width: 50,

align: 'center'

},

{

title: '公司名称',

key: 'corporateName',

},

{

title: '房间号',

key: 'roomNumber',

width:60,

render:(h,params) =>{

return h('span',{

'style':{

'color':'#169BD5',

'cursor':'pointer'

},

on: {

click: () => { //点击合同号显示详情

this.DetailWindowFindcontractNumber = params.row.contractNumber

console.log(this.DetailWindowFindcontractNumber);

this.DetailWindowFindInter(this.DetailWindowFindcontractNumber)

setTimeout(()=>{

this.DetailRoomsWindowIsShow=true

},800)

}

}

},params.row.roomNumber)

}

},

{

title: '合同号',

key: 'contractNumber',

render:(h,params) =>{

return h('span',{

'style':{

'color':'#169BD5',

'cursor':'pointer'

},

on: {

click: () => { //点击合同号显示详情

this.DetailWindowFindcontractNumber = params.row.contractNumber

console.log(this.DetailWindowFindcontractNumber);

this.DetailWindowFindInter(this.DetailWindowFindcontractNumber)

setTimeout(()=>{

this.DetailWindowIsShow = true

},800)

}

}

},params.row.contractNumber)

}

},

{

title: '入住日期',

key: 'housingDate',

},

{

title: '联系人',

key: 'contactName',

},

{

title: '联系电话',

key: 'phoneNumber',

},

{

title: '邮箱',

key: 'mailbox',

width:180

},

{

title: '注册状态',

key: 'registrationStatus'

},

{

title: '操作',

slot: 'action',

width: 150,

align: 'center'

}

]

}else if(this.roleName == "财务人员"||this.roleName == "维修人员"){

this.isJurisdiction = false

this.fathermenu =[

{

type: 'selection',

width: 50,

align: 'center'

},

{

title: '公司名称',

key: 'corporateName',

},

{

title: '房间号',

key: 'roomNumber',

width:60,

render:(h,params) =>{

return h('span',{

'style':{

'color':'#169BD5',

'cursor':'pointer'

},

on: {

click: () => { //点击合同号显示详情

this.DetailWindowFindcontractNumber = params.row.contractNumber

console.log(this.DetailWindowFindcontractNumber);

this.DetailWindowFindInter(this.DetailWindowFindcontractNumber)

setTimeout(()=>{

this.DetailRoomsWindowIsShow=true

},400)

}

}

},params.row.roomNumber)

}

},

{

title: '合同号',

key: 'contractNumber',

render:(h,params) =>{

return h('span',{

'style':{

'color':'#169BD5',

'cursor':'pointer'

},

on: {

click: () => { //点击合同号显示详情

this.DetailWindowFindcontractNumber = params.row.contractNumber

console.log(this.DetailWindowFindcontractNumber);

this.DetailWindowFindInter(this.DetailWindowFindcontractNumber)

setTimeout(()=>{

this.DetailWindowIsShow = true

},400)

}

}

},params.row.contractNumber)

}

},

{

title: '入住日期',

key: 'housingDate',

},

{

title: '联系人',

key: 'contactName',

},

{

title: '联系电话',

key: 'phoneNumber',

},

{

title: '邮箱',

key: 'mailbox',

width:180

},

{

title: '注册状态',

key: 'registrationStatus'

}

]

}else {

this.isJurisdiction = false

this.fathermenu = []

}

},

components:{

Options,

ClientWindow,

List,

DetailWindow,

RoomsWindow

},

methods:{

//以下是网络请求的方法

clientInterList(){ //这是初始页面的请求

this.$ajax.get('/api/merchant/list',this.clientfinddata).then(res =>{

console.log(res);

this.msg = res.data.data.data

this.totalPages = res.data.data.totalPages\*10

}).catch(err =>{

console.log(err);

})

},

clientFindIdInternet(id){ //客户信息根据id查找填入弹窗

this.$ajax.get('/api/merchant/'+id).then(res =>{

this.cli\_window\_text = res.data.data

let registrationStatusName = this.cli\_window\_text.registrationStatus

this.cli\_window\_text.registrationStatusName = registrationStatusName

this.cli\_window\_text.registrationStatus = registrationStatusName === '否' ? "NO" : "YES"

}).catch(err =>{

console.log(err);

})

},

DetailWindowFindInter(contractNumber){ //客户详情根据合同号在合同的数据库中查找

this.$ajax.get('/api/agreement/contractNumber?contractNumber='+contractNumber).then(res =>{

console.log(res);

this.DetailWindow\_text = res.data.data

console.log(this.DetailWindow\_text);

this.rentRoomSet = res.data.data.rooms

console.log(this.rentRoomSet);

this.DetailRoomscorporateName = res.data.data.corporateName

console.log(this.DetailRoomscorporateName);

}).catch(err =>{

console.log(err);

})

},

ClientExportInter(arr){ //客户页点击导出的网络请求

this.$ajax.download('/api/merchant/export',{"ids":arr}).then((res)=>{

console.log( res.data)

const link = document.createElement('a')

let blob = new Blob([res.data],{type: 'application/vnd.ms-excel'});

link.style.display = 'none'

link.href = URL.createObjectURL(blob);

const filename = res.headers["content-disposition"].split(";")[1].split("filename=")[1];

link.setAttribute('download', decodeURI(filename))

document.body.appendChild(link)

link.click()

URL.revokeObjectURL(link.href)

document.body.removeChild(link)

setTimeout(() => { //下载后完刷新页面

history.go(0)

}, 3000);

}).catch(err=>{

console.log(err)

})

},

//以下是页面的方法

add(){ //点击显示弹窗

this.isAdd=true

},

turnoff(data){ //从弹窗发过来的请求关闭弹窗

this.isAdd = data;

},

clifind(){ //点击查询的模糊查询

this.clientfinddata.pageNo = 1

this.clientInterList()

this.changeCurrentPage()

},

changepage(index){ //分页器翻页请求

console.log(index);

this.clientfinddata.pageNo= index

console.log(this.clientfinddata);

this.clientInterList()

//分页之前把downloadid,保存到本地。

this.$refs.dataList.handleSelectRow()

let oldids = sessionStorage.getItem("downLoadIds")

let newids = oldids+","+this.DownloadId

sessionStorage.setItem("downLoadIds",newids)

},

Clientshow(id,MainWindowListisShow,index){ //点击List修改按钮,弹出模态框

if(this.$route.path == "/client"){

this.clientFindId = id

this.clientFindIdInternet(this.clientFindId)

setTimeout(() => {

this.isAdd = true

}, 600);

}

},

DetailWindowOff(data){ //合同号的详情弹窗点击关闭

this.DetailWindowIsShow = data

},

RoomsWindowOff(data){ //房间的详情弹窗点击关闭

this.DetailRoomsWindowIsShow = data

},

ClientExport(){ //客户页点击导出的方法

console.log(sessionStorage.getItem("downLoadIds")+","+this.DownloadId);

let idarr = (Array.from(new Set(((sessionStorage.getItem("downLoadIds")+","+this.DownloadId)).split(","||",,")))).slice(1)

console.log(idarr);

if(idarr.length!=0&&idarr[0]!=""){

this.ClientExportInter(idarr)

}else{

this.$Message["error"]({

background: true,

content:"请先选择打印数据"

});

}

sessionStorage.setItem("downLoadIds","")

},

changeCurrentPage(){ //分页器样式重置为第一页

this.$nextTick(function(){

this.$refs['pages'].currentPage = 1;

})

},

saveDownloadId(data){ //Listc传出的事件保存当前页选中的ID号

this.DownloadId = data

}

}

}

</script>

<style scoped>

.box{

background:#FFF;

}

.options{

background:#f3f3f3;

margin-top: 35px;

padding-bottom: 16px;

}

.options-item{

margin-right: 16px;

}

#cli-page{

background: #f3f3f3;

padding: 30px 0 0px 0;

}

</style>

<template>

<div class="fin-box">

<div class="SecondMenu">

<SecondMenu @showright="sonindex" :sonsecmenu="fathersecmenu" id="SecondMenu"></SecondMenu>

</div>

<div class="fin-right">

<component :is="whitch" @RoomsWindowIsShow="RoomsWindowIsShow" @PaymentchangePath="PaymentchangePath"></component>

</div>

</div>

</template>

<script>

import Options from "../../components/options/Options";

import SecondMenu from "../../components/second-menu/SecondMenu";

import List from "../../components/list/List";

import contract from "./child/contract";

import electric from "./child/electric";

import payment from "./child/payment";

export default {

name: "Finance",

data() {

return {

// fathersecmenu: [

// {

// name: "合同管理",

// path: "agreement",

// icon: "iconfont icon-hetongguanli",

// },

// {

// name: "缴费管理",

// path: "pay",

// icon: "iconfont icon-jiaofei",

// },

// {

// name: "电费填报",

// path: "electricCharge",

// icon: "iconfont icon-dianfei",

// },

// ],

fathersecmenu: this.fathersecmenuISshow(),

isAdd: false,

leftindex: 0,

msg: [

{

contractNumber: "zjxhjzhx219389",

roomNumber: "301",

corporateName: "杰讯鸿翔",

operatorName: "王二麻子",

phoneNumber: 12233334444,

area: 999,

contactName: "张三",

mailbox: "456789111@133.com",

rent: 30000,

deposit: 500000,

payMoneyWay: "年付",

createTime: "2009-1-1",

electricCharge: "500",

remarks: "这是没网的时候显示的数据",

},

],

fathermenu: [

{

type: "selection",

width: 50,

align: "center",

},

{

title: "合同号",

key: "contractNumber",

},

{

title: "房间号",

key: "roomNumber",

width: 60,

},

{

title: "公司名称",

key: "corporateName",

},

{

title: "经办人",

key: "operatorName",

},

{

title: "联系电话",

key: "phoneNumber",

},

{

title: "面积",

key: "area",

},

{

title: "租金",

key: "rent",

},

{

title: "保证金",

key: "deposit",

},

{

title: "付款方式",

key: "payMoneyWay",

},

{

title: "创建日期",

key: "createTime",

},

{

title: "电费",

key: "electricCharge",

},

{

title: "备注",

key: "remarks",

},

{

title: "操作",

slot: "action",

width: 150,

align: "center",

},

],

whitch: contract,

roleName:sessionStorage.getItem("roleName"),

leftPathSrting:'agreement',

};

},

components: {

Options,

SecondMenu,

List,

contract,

electric,

payment,

},

methods: {

fathersecmenuISshow(){

this.roleName = sessionStorage.getItem("roleName")

if(this.roleName=="系统管理员"||this.roleName=="物业人员"||this.roleName=="财务人员"){

return [

{

name: "合同管理",

path: "agreement",

icon: "iconfont icon-hetongguanli",

},

{

name: "缴费管理",

path: "pay",

icon: "iconfont icon-jiaofei",

},

{

name: "电费填报",

path: "electricCharge",

icon: "iconfont icon-dianfei",

},

]

}else if(this.roleName=="维修人员"){

return [

{

name: "电费填报",

path: "electricCharge",

icon: "iconfont icon-dianfei",

}

]

}else {

return []

}

},

add() {

this.isAdd = true;

},

turnoff(data) {

this.isAdd = data;

},

sonindex(index,item) {

this.leftindex = index;

this.leftPathSrting = item

},

RoomsWindowIsShow(data){

this.RoomsWindowIsShowBoo = data

},

PaymentchangePath(data){

this.leftPathSrting = data

}

},

watch: {

leftPathSrting(val) {

console.log(val);

switch (val) {

case "agreement":

this.whitch = contract;

break;

case "pay":

this.whitch = payment;

break;

case "electricCharge":

this.whitch = electric;

break;

}

},

},

};

</script>

<style>

.fin-box {

background: #f3f3f3;

}

#SecondMenu{

margin: -20px 25px 0 0 ;

}

</style>

<template>

<div class="right-box">

<Options class="options" :isJurisdiction="isJurisdiction">

<PaymentAddwindow slot="butt" @PaymentchangePath="PaymentchangePath"></PaymentAddwindow>

<input type="text" slot="other" placeholder="房间号" v-model="payMentfindParams.roomNumber">

<button slot="other" @click="find">搜索</button>

<button slot="other" @click="paymentExport">导出Excel</button>

<div slot="other">

<span>显示条数：</span>

<input slot="other" type="text" style="width:40px;margin: 0 10px 0 0" v-model="payMentfindParams.pageSize" @change="find">

</div>

</Options>

<div style="height:640px;overflow: auto;background:#FFF;">

<PaymentLis :message="msg" @Addshow="Addshow" ref="dataList"

@saveDownloadId="saveDownloadId"

@Listchange="Listchange"

></PaymentLis>

</div>

<DetailPaymentAdd ref="DetailPaymentAdd" :upDataId="upDataId"

></DetailPaymentAdd>

<ChangePaymentAdd ref="ChangePaymentAdd" :DetailPaymentAddParams='PaymentListUpitem'></ChangePaymentAdd>

<Page ref="pages" :total=totalPages size="small" id="mini" @on-change="changepage"/>

</div>

</template>

<script>

import Options from "../../../components/options/Options"

import PaymentLis from "./contract-child/PaymentList"

import PaymentAddwindow from "./contract-child/PaymentAddwindow"

import DetailPaymentAdd from "./contract-child/DetailPaymentAdd"

import ChangePaymentAdd from "./contract-child/ChangePaymentAdd"

export default {

name: 'payment',

data () {

return {

msg:[],

isDeleteShowBoo:true,

payMentfindParams:{

pageNo:'1',

pageSize:"10",

roomNumber:null

},

totalPages:10,

upDataId:null,

DownloadId:[],

changepath: "pay",

isJurisdiction:false,

roleName:sessionStorage.getItem("roleName"),

PaymentListUpitem:{}

}

},

created(){

this.payStartInter()

if(this.roleName == "系统管理员"||this.roleName == "财务人员"){

this.isJurisdiction = true

}

},

components:{

Options,

PaymentLis,

PaymentAddwindow,

DetailPaymentAdd,

ChangePaymentAdd

},

methods:{

//以下是请求网络的方法

payStartInter(){

this.$ajax.get('/api/pay/list',this.payMentfindParams).then(res =>{

console.log(res.data.data.data);

let arr = res.data.data.data

for(var i = 0;i<arr.length;i++){

this.filter(arr[i])

if(arr[i].children){

arr[i].children.forEach( item => {

item.childrenid = item.payId

this.filter(item)

});

}

}

console.log(arr);

this.msg = arr

this.totalPages = res.data.data.totalPages\*10

}).catch(err =>{

console.log(err);

})

},

PaymentExportInter(arr) {//财务-合同管理 导出的网络请求

this.$ajax

.download("/api/pay/export/pay", { 'ids': arr })

.then((res) => {

console.log(res.data);

const link = document.createElement("a");

let blob = new Blob([res.data], { type: "application/vnd.ms-excel" });

link.style.display = "none";

link.href = URL.createObjectURL(blob);

const filename = res.headers["content-disposition"]

.split(";")[1]

.split("filename=")[1];

link.setAttribute("download", decodeURI(filename));

document.body.appendChild(link);

link.click();

URL.revokeObjectURL(link.href);

document.body.removeChild(link);

})

.catch((err) => {

console.log(err);

});

},

//一下是本页面的方法

changepage(index){ //分页器翻页请求

this.payMentfindParams.pageNo= index

this.payStartInter()

// //分页之前把downloadid,保存到本地。

this.$refs.dataList.handleSelectRow()

let oldids = sessionStorage.getItem("downLoadIds")

let newids = oldids+","+this.DownloadId

sessionStorage.setItem("downLoadIds",newids)

},

find(){ //查找按钮事件

this.payMentfindParams.pageNo= 1

this.payStartInter()

},

Addshow(id){ //list例表发过来的新增事件

console.log(id);

this.upDataId = id

this.$refs.DetailPaymentAdd.open()

},

paymentExport(){

console.log(sessionStorage.getItem("downLoadIds")+","+this.DownloadId);

let idarr = (Array.from(new Set(((sessionStorage.getItem("downLoadIds")+","+this.DownloadId)).split(","||",,")))).slice(1)

console.log(idarr);

if(idarr.length!=0&&idarr[0]!=""){

this.PaymentExportInter(idarr)

setTimeout(() => { //下载后完刷新页面

history.go(0)

}, 3000);

}else{

this.$Message["error"]({

background: true,

content:"请先选择打印数据"

});

}

sessionStorage.setItem("downLoadIds","")

},

saveDownloadId(data){ //Listc传出的事件保存当前页选中的ID号

this.DownloadId = data

},

PaymentchangePath(){ //window传出来的重新定位到缴费管理的页面

this.$emit("PaymentchangePath",this.changepath)

},

filter(newarr){

if( newarr.electricCharge&&newarr.unitPrice&&newarr.rentMoney){

newarr.electricCharge = newarr.electricCharge/100

newarr.unitPrice = newarr.unitPrice/100

newarr.rentMoney = newarr.rentMoney/100

}

},

Listchange(data){ //列表点击修改传递的事件

this.PaymentListUpitem = data

this.$refs.ChangePaymentAdd.open()

}

}

}

</script>

<style scoped>

.right-box{

margin: 20px 0 0 0;

}

.options input{

margin: 0 18px 10px 0px;

}

#mini{

background: #f3f3f3;

padding: 30px 0 0 0;

}

</style>

<template>

<div class="right-box">

<Options class="options" :isJurisdiction="isJurisdiction">

<ElectricWindow slot="butt" @updatalist="updatalist">新增</ElectricWindow>

<button slot="other" @click="ElectricExport">导出EXcel</button>

<div slot="other">

<span>显示条数：</span>

<input slot="other" type="text" style="width:40px;margin: 0 10px 0 0" v-model="ElectricFindparams.pageSize" @change="ElectricFindInter">

</div>

</Options>

<div style="height:640px;overflow: auto;background:#FFF;margin:15px 0 30px 0 ">

<List :message="msg" :sonmenu="fathermenu" @lisgcheangeshow="homeshow" :isDeleteShowBoo="isDeleteShowBoo"

@messageIndexData="messageIndexData"

@saveDownloadId="saveDownloadId"

ref="dataList"

:isJurisdiction="isJurisdiction"

></List>

</div>

<ChangeElectricWindow ref="ChangeElectricWindow" :ElectricAddParams="ChangeElectricWindowData" :upDataId="upDataId"></ChangeElectricWindow>

<Page ref="pages" :total=totalPages size="small" id="mini" @on-change="changepage"/>

</div>

</template>

<script>

import Options from '../../../components/options/Options';

import List from '../../../components/list/List';

import ElectricWindow from './contract-child/ElectricWindow';

import ChangeElectricWindow from './contract-child/ChangeElectricWindow';

export default {

name: 'electric',

data () {

return {

ElectricFindparams:{

pageNo:1,

pageSize:16

},

fathermenu:[

{

type: 'selection',

width: 50,

align: 'center',

},

{

title: '楼宇名称',

width: 70,

key: 'floorName',

},

{

title: '楼层',

width: 70,

key: 'storey'

},

{

title: '房间号',

width: 70,

key: 'roomNumber',

},

{

title: '公司名称',

key: 'corporateName',

},

{

title: '面积',

key: 'area',

width: 70,

},

{

title:'上期电表数',

key:'lastPeriodElectricity'

},

{

title:'本期电表数',

key:'currentElectricity'

},

{

title:'电表倍率',

key:'electricityOdds'

},

{

title:'单价',

key:'unitPrice'

},

{

title:'金额',

key:'sumMoney',

},

{

title: '备注',

key: 'remarks',

},

{

title: '操作',

slot: 'action',

width: 100,

align: 'center'

}

],

msg:[],

isDeleteShowBoo:true,

totalPages:10,

ChangeElectricWindowData:{},

upDataId:null,

DownloadId:[],

isJurisdiction:false,

roleName:sessionStorage.getItem("roleName"),

}

},

created(){

this.ElectricFindInter()

sessionStorage.setItem("downLoadIds", "");

this.roleName = sessionStorage.getItem("roleName")

if(this.roleName == "系统管理员"||this.roleName == "财务人员"||this.roleName == "维修人员"){

this.isJurisdiction = true

this.fathermenu = [

{

type: 'selection',

width: 50,

align: 'center',

},

{

title: '楼宇名称',

width: 70,

key: 'floorName',

},

{

title: '楼层',

width: 70,

key: 'storey'

},

{

title: '房间号',

width: 70,

key: 'roomNumber',

},

{

title: '公司名称',

key: 'corporateName',

},

{

title: '面积',

key: 'area',

width: 70,

},

{

title:'上期电表数',

key:'lastPeriodElectricity'

},

{

title:'本期电表数',

key:'currentElectricity'

},

{

title:'电表倍率',

key:'electricityOdds'

},

{

title:'单价',

key:'unitPrice'

},

{

title:'金额',

key:'sumMoney',

},

{

title: '备注',

key: 'remarks',

},

{

title: '操作',

slot: 'action',

width: 100,

align: 'center'

}

]

}else if(this.roleName == "物业人员"){

this.isJurisdiction = false

this.fathermenu = [

{

type: 'selection',

width: 50,

align: 'center',

},

{

title: '楼宇名称',

width: 70,

key: 'floorName',

},

{

title: '楼层',

width: 70,

key: 'storey'

},

{

title: '房间号',

width: 70,

key: 'roomNumber',

},

{

title: '公司名称',

key: 'corporateName',

},

{

title: '面积',

key: 'area',

width: 70,

},

{

title:'上期电表数',

key:'lastPeriodElectricity'

},

{

title:'本期电表数',

key:'currentElectricity'

},

{

title:'电表倍率',

key:'electricityOdds'

},

{

title:'单价',

key:'unitPrice'

},

{

title:'金额',

key:'sumMoney',

},

{

title: '备注',

key: 'remarks',

}

]

}else {

this.isJurisdiction = false

this.fathermenu = []

}

},

components:{

Options,

List,

ElectricWindow,

ChangeElectricWindow

},

methods:{

//以下是网络请求的方法

ElectricFindInter(){

this.$ajax.get('/api/electriccharge/list',this.ElectricFindparams).then(res =>{

console.log(res.data.data.data);

this.msg = res.data.data.data.map(item=>{

let obj ={

area:item.area,

corporateName:item.corporateName,

createTime:item.createTime,

currentElectricity:item.currentElectricity/100,

electricityOdds:item.electricityOdds/100,

floorName:item.floorName,

id:item.id,

lastPeriodElectricity:item.lastPeriodElectricity/100,

remarks:item.remarks,

roomNumber:item.roomNumber,

storey:item.storey,

sumMoney:item.sumMoney/10000,

unitPrice:item.unitPrice,

}

return obj

})

this.totalPages = res.data.data.totalPages\*10

}).catch(err =>{

console.log(err);

})

},

ElectricExportInter(arr){ //电费填报点击导出的网络请求

this.$ajax.download('/api/electriccharge/export/electriccharge',{'ids':arr}).then((res)=>{

console.log( res.data)

const link = document.createElement('a')

let blob = new Blob([res.data],{type: 'application/vnd.ms-excel'});

link.style.display = 'none'

link.href = URL.createObjectURL(blob);

const filename = res.headers["content-disposition"].split(";")[1].split("filename=")[1];

link.setAttribute('download', decodeURI(filename))

document.body.appendChild(link)

link.click()

URL.revokeObjectURL(link.href)

document.body.removeChild(link)

setTimeout(() => { //下载后完刷新页面

history.go(0)

}, 3000);

}).catch(err=>{

console.log(err)

})

},

//以下是本页面的方法事件

homeshow(id,boo,index){

console.log(id);

this.$refs.ChangeElectricWindow.open()

this.upDataId = id

},

messageIndexData(data){

this.ChangeElectricWindowData = data

},

ElectricExport(){

console.log(sessionStorage.getItem("downLoadIds")+","+this.DownloadId);

let idarr = (Array.from(new Set(((sessionStorage.getItem("downLoadIds")+","+this.DownloadId)).split(","||",,")))).slice(1)

console.log(idarr);

if(idarr.length!=0&&idarr[0]!=""){

this.ElectricExportInter(idarr)

}else{

this.$Message["error"]({

background: true,

content:"请先选择打印数据"

});

}

sessionStorage.setItem("downLoadIds","")

},

saveDownloadId(data){ //Listc传出的事件保存当前页选中的ID号

this.DownloadId = data

},

changepage(index){

this.ElectricFindparams.pageNo= index

this.ElectricFindInter()

//分页之前把downloadid,保存到本地。

this.$refs.dataList.handleSelectRow()

let oldids = sessionStorage.getItem("downLoadIds")

let newids = oldids+","+this.DownloadId

sessionStorage.setItem("downLoadIds",newids)

},

updatalist(){

this.ElectricFindInter()

}

},

}

</script>

<style scoped>

.right-box{

margin: 20px 0 0 0;

}

</style>

<template>

<div class="right-box">

<Options class="options" :isJurisdiction="isJurisdiction">

<ContractWindow slot="butt">新增</ContractWindow>

<input type="text" slot="other" placeholder="关键词" v-model="ContractFindParams.corporateName" />

<button slot="other" @click="find">搜索</button>

<button slot="other" @click="contractExport">导出Excel</button>

<div slot="other">

<span>显示条数：</span>

<input slot="other" type="text" style="width:40px;margin: 0 10px 0 0" v-model="ContractFindParams.pageSize" @change="find">

</div>

</Options>

<div style="height:640px;overflow: auto;background:#FFF;margin:15px 0 0 0 ">

<List

:sonmenu="fathermenu"

:message="msg"

@lisgcheangeshow="contractshow"

:isDeleteShowBoo="isDeleteShowBoo"

ref="dataList"

@saveDownloadId="saveDownloadId"

:isJurisdiction="isJurisdiction"

></List>

</div>

<RoomsWindow

v-if="RoomsWindowIsShowBoo"

@RoomsWindowOff="RoomsWindowOff"

:rentRoomSet="rentRoomSet"

:DetailRoomscorporateName="DetailRoomscorporateName"

></RoomsWindow>

<DetailChange ref="DetailChange" :Detailist="DetailChangemsg" :upDateId="upDateId" :upDateIndex="upDateIndex" @ContractlistItemUpdated="ContractlistItemUpdated"></DetailChange>

<Page ref="pages" :total=totalPages size="small" id="mini" @on-change="changepage"/>

</div>

</template>

<script>

import Options from "../../../components/options/Options";

import List from "../../../components/list/List.vue";

import ContractWindow from "./contract-child/ContractWindow";

import RoomsWindow from "../../../components/rooms-window/RoomsWindow";

import DetailChange from "./contract-child/DetailChange";

export default {

name: "contract",

data() {

return {

isDeleteShowBoo: false, //控制List删除按钮的显示

fathermenu: [],

msg: [],

ContractFindParams: {

corporateName: null,

pageNo: "1",

pageSize: "15",

},

totalPages:10 ,

rentRoomSet: [],

DetailRoomscorporateName: "",

RoomsWindowIsShowBoo: false,

isshow: false,

DetailChangemsg: {},

upDateId: null,

upDateIndex:null,

DownloadId:[],

isJurisdiction:false,

roleName:sessionStorage.getItem("roleName"),

};

},

created() {

this.ContractStartInter();

sessionStorage.setItem("downLoadIds", "");

this.roleName = sessionStorage.getItem("roleName")

if(this.roleName == "系统管理员"||this.roleName == "财务人员"){

this.isJurisdiction = true

this.fathermenu = [

{

type: "selection",

width: 50,

align: "center",

},

{

title: "合同号",

key: "contractNumber",

},

{

title: "房间号",

key: "roomNumber",

render: (h, params) => {

return h(

"span",

{

style: {

color: "#169BD5",

cursor: "pointer",

},

on: {

click: () => {

//点击合同号显示详情

console.log(params.row);

this.rentRoomSet = params.row.rooms;

this.DetailRoomscorporateName = params.row.corporateName;

setTimeout(() => {

this.RoomsWindowIsShowBoo = true;

}, 400);

},

},

},

params.row.roomNumber

);

},

width: 60,

},

{

title: "公司名称",

width: 130,

key: "corporateName",

},

{

title: "经办人",

width: 60,

key: "operatorName",

},

{

title: "联系电话",

key: "phoneNumber",

},

{

title: "总面积/m²",

width: 70,

key: "sumArea",

},

{

title: "租金/元",

width: 60,

key: "rent",

},

{

title: "保证金/元",

width: 60,

key: "deposit",

},

{

title: "付款方式",

width: 70,

key: "payMoneyWayString",

},

{

title: "开始日期",

key: "startDate",

},

{

title: "结束日期",

key: "endDate",

},

{

title: "电费/月(元)",

width: 70,

key: "electricCharge",

},

{

title: "备注",

width: 150,

key: "remarks",

},

{

title: "操作",

slot: "action",

width: 50,

align: "center",

}

]

}else if(this.roleName == "物业人员"){

this.isJurisdiction = false

this.fathermenu = [

{

type: "selection",

width: 50,

align: "center",

},

{

title: "合同号",

key: "contractNumber",

},

{

title: "房间号",

key: "roomNumber",

render: (h, params) => {

return h(

"span",

{

style: {

color: "#169BD5",

cursor: "pointer",

},

on: {

click: () => {

//点击合同号显示详情

console.log(params.row);

this.rentRoomSet = params.row.rooms;

this.DetailRoomscorporateName = params.row.corporateName;

setTimeout(() => {

this.RoomsWindowIsShowBoo = true;

}, 400);

},

},

},

params.row.roomNumber

);

},

width: 60,

},

{

title: "公司名称",

width: 130,

key: "corporateName",

},

{

title: "经办人",

width: 60,

key: "operatorName",

},

{

title: "联系电话",

key: "phoneNumber",

},

{

title: "总面积/m²",

width: 70,

key: "sumArea",

},

{

title: "租金/元",

width: 60,

key: "rent",

},

{

title: "保证金/元",

width: 60,

key: "deposit",

},

{

title: "付款方式",

width: 70,

key: "payMoneyWayString",

},

{

title: "开始日期",

key: "startDate",

},

{

title: "结束日期",

key: "endDate",

},

{

title: "电费/月(元)",

width: 70,

key: "electricCharge",

},

{

title: "备注",

width: 150,

key: "remarks",

}

]

}else {

this.isJurisdiction = false

this.fathermenu = []

}

},

components: {

Options,

List,

ContractWindow,

RoomsWindow,

DetailChange,

},

methods: {

//以下是网络请求的方法

ContractStartInter() {

//页面一加载的初始请求 页数加页码

this.$ajax

.get("/api/agreement/list", this.ContractFindParams)

.then((res) => {

console.log(res.data.data.data);

this.msg = res.data.data.data.map(item =>{

let obj = {

contractNumber:item.contractNumber,

corporateName:item.corporateName,

deposit:item.deposit/100,

electricCharge:item.electricCharge/100,

endDate:item.endDate,

id:item.id,

mailbox:item.mailbox,

operatorName:item.operatorName,

payMoneyWay:item.payMoneyWay,

payMoneyWayString:item.payMoneyWayString,

phoneNumber:item.phoneNumber,

remarks:item.remarks,

rent:item.rent/100,

roomNumber:item.roomNumber,

rooms:item.rooms,

startDate:item.startDate,

sumArea:item.sumArea,

}

return obj

});

this.totalPages = res.data.data.totalPages\*10

})

.catch((err) => {

console.log(err);

});

},

ContractExportInter(arr) {//财务-合同管理 导出的网络请求

this.$ajax

.download("/api/agreement/export/agreement", { 'ids': arr })

.then((res) => {

console.log(res.data);

const link = document.createElement("a");

let blob = new Blob([res.data], { type: "application/vnd.ms-excel" });

link.style.display = "none";

link.href = URL.createObjectURL(blob);

const filename = res.headers["content-disposition"]

.split(";")[1]

.split("filename=")[1];

link.setAttribute("download", decodeURI(filename));

document.body.appendChild(link);

link.click();

URL.revokeObjectURL(link.href);

document.body.removeChild(link);

})

.catch((err) => {

console.log(err);

});

},

//以下是本页面的方法

contractshow(id, b, index) {

//点击合同管理的列表的 修改 应该进行的事件

this.$refs.DetailChange.open();

this.DetailChangemsg = this.msg[index];

this.upDateId = id;

this.upDateIndex = index;

},

RoomsWindowOff(data) {//点击关闭rooms的弹窗事件

this.RoomsWindowIsShowBoo = data;

},

find() {

//点击搜索进行的事件

this.ContractFindParams.pageNo = 1;

this.ContractStartInter();

},

changepage(index){ //分页器翻页事件

this.ContractFindParams.pageNo= index

this.ContractStartInter()

//分页之前把downloadid,保存到本地。

this.$refs.dataList.handleSelectRow()

let oldids = sessionStorage.getItem("downLoadIds")

let newids = oldids+","+this.DownloadId

sessionStorage.setItem("downLoadIds",newids)

},

saveDownloadId(data){ //Listc传出的事件保存当前页选中的ID号

this.DownloadId = data

},

contractExport() { //点击导出的事件

console.log(sessionStorage.getItem("downLoadIds")+","+this.DownloadId);

let idarr = (Array.from(new Set(((sessionStorage.getItem("downLoadIds")+","+this.DownloadId)).split(","||",,")))).slice(1)

console.log(idarr);

if(idarr.length!=0&&idarr[0]!=""){

this.ContractExportInter(idarr)

setTimeout(() => { //下载后完刷新页面

history.go(0)

}, 3000);

}else{

this.$Message["error"]({

background: true,

content:"请先选择打印数据"

});

}

sessionStorage.setItem("downLoadIds","")

},

ContractlistItemUpdated(newItem,index){ //使用List组件的更新方法

this.$refs.dataList.updateListItem(index, newItem)

}

},

};

</script>

<style scoped>

.right-box {

margin: 20px 0 0 0;

}

.options input {

margin: 0 18px 0 8px;

}

#mini{

background: #f3f3f3;

padding: 30px 0 0 0;

}

</style>

<template>

<div class="box">

<Options class="options" :isJurisdiction="isJurisdiction">

<button slot="butt" class="options-item " @click="add" >+ 新增</button>

<button slot="other" class="options-item" @click="HomeExport">导出Excel</button>

<div slot="other">

<span>显示条数：</span>

<input slot="other" type="text" style="width:40px;margin: 0 10px 0 0" v-model="findparams.pageSize" @change="find">

</div>

<div slot="other" class="options-item">

<span>位置：</span><select class="homesele">

<option v-for="item in address" :key="item">{{item}}</option>

</select>

</div>

<div slot="other" class="options-item">

<span>状态：</span><select class="homesele">

<option selected>全部</option>

<option>空置</option>

<option>已租</option>

</select>

</div>

<div slot="other" class="options-item">

<span>排序方式：</span><select class="homesele date">

<option value="" selected>无</option>

<option value="DESC">面积从大到小</option>

<option value="ASC" >面积从小到大</option>

</select>

</div>

<button slot="other" class="options-item" @click="find">搜索</button>

</Options>

<HomeWindow v-if="isAdd" @turnoff="turnoff" :sontitle="isAddtatle" v-on:listItemUpdated="updateListItemMethod" :itemIndex="itemIndex" :isShowId="listShowId" :textareavalue = "textareavalue">

</HomeWindow>

<div style="width:100%;height:640px;overflow: auto;">

<List ref="dataList" :message="msg" :sonmenu="fathermenu"

@lisgcheangeshow="homeshow"

:isDeleteShowBoo="isDeleteShowBoo"

@saveDownloadId="saveDownloadId"

:isJurisdiction="isJurisdiction"

></List>

</div>

<Page ref="pages" :total=totalPages size="small" id="mini" @on-change="changepage"/>

</div>

</template>

<script>

import qs from "qs"

import Options from "../../components/options/Options"

import HomeWindow from "./childs/window/HomeWindow"

import List from "../../components/list/List"

import LoginVue from '../login/Login.vue'

export default {

name:"Home",

data(){

return{

isAdd:false,

isAddtatle:"新增",

isAddshowdata:{},

textareavalue:"",

address:["全部","A座","A-1","A-2","A-3","A-4","A-5","A-6","A-7","A-8",

"B座","B-1","B-2","B-3","B-4","B-5","B-6","B-7","B-8",

"C座","C— -1","C-1","C-2","C-3","C-4","C-5","C-6"

],

findparams:{ //这是请求物业列表的信息，默认什么都不传

floorName:null,

pageNo:'1',

pageSize:'15',

romeType:null,

storey:null,

orderString:null

},

msg:[ ], // 这是列表页的数据

fathermenu:[], // 这是列表页的表头数据

totalPages:10 , //这个是列表页的页数,

listShowId:null,

itemIndex: null,

updateListItem: null,

isDeleteShowBoo:true, //控制List删除按钮的显示

ExportIdObj:{'ids':[1,2,3]},

currentPage:1,

DownloadId:[],

roleName:sessionStorage.getItem("roleName"),

isJurisdiction:false

}

},

created(){

this.HouseNone()

this.roleName = sessionStorage.getItem("roleName")

console.log(this.roleName);

if(this.roleName == "系统管理员"||this.roleName == "物业人员"){

this.isJurisdiction = true

this.fathermenu = [

{

type: 'selection',

width: 50,

align: 'center',

},

{

title: '楼宇名称',

width: 70,

key: 'floorName',

},

{

title: '楼层',

width: 70,

key: 'storey'

},

{

title: '房间号',

width: 70,

key: 'roomNumber',

},

{

title: '面积',

key: 'area',

width: 70

},

{

title: '朝向',

key:"orientations"

},

{

title: '状态',

key: 'romeType',

render:(h,params) =>{

if(params.row.romeType ==="已租"){

return h('span',{

'style':{

'color':'#339900'

}

},params.row.romeType)

}else{

return h('span',{

'style':{

'color':'#FEA004'

}

}, params.row.romeType)

}

}

},

{

title: '备注',

key: 'remarks',

width:300

},

{

title: '操作',

slot: 'action',

width: 150,

align: 'center'

}

]

}else if(this.roleName == "财务人员"||this.roleName == "维修人员"||this.roleName == "普通商户"){

this.isJurisdiction = false

this.fathermenu = [

{

type: 'selection',

width: 50,

align: 'center',

},

{

title: '楼宇名称',

width: 70,

key: 'floorName',

},

{

title: '楼层',

width: 70,

key: 'storey'

},

{

title: '房间号',

width: 70,

key: 'roomNumber',

},

{

title: '面积',

key: 'area',

width: 70

},

{

title: '朝向',

key:"orientations"

},

{

title: '状态',

key: 'romeType',

render:(h,params) =>{

if(params.row.romeType ==="已租"){

return h('span',{

'style':{

'color':'#339900'

}

},params.row.romeType)

}else{

return h('span',{

'style':{

'color':'#FEA004'

}

}, params.row.romeType)

}

}

},

{

title: '备注',

key: 'remarks',

width:300

}

]

}

},

methods:{

//以下是请求数据的方法

HouseNone(){ //默认根据时间搜索的请求

this.$ajax.get('/api/property/list',this.findparams).then((res)=>{

this.msg =res.data.data.data

console.log(this.msg);

this.totalPages = res.data.data.totalPages\*10

}).catch(err=>{

console.log(err)

})

},

DateSortInter(){ //根据面积排序的请求

this.$ajax.get('/api/property/list/byarea',this.findparams).then((res)=>{

this.msg =res.data.data.data

console.log(this.msg);

this.totalPages = res.data.data.totalPages\*10

}).catch(err=>{

console.log(err)

})

},

homefindInter(){ //点击查找字段的接口

var seles = document.getElementsByClassName("windowsele")

var ipts = document.getElementsByClassName("windowipt")

var tex = document.getElementsByTagName("textarea")

this.$ajax.get('/api/property/'+this.listShowId).then((res)=>{

console.log(res);

this.isAddshowdata = res.data.data

seles[0].value = res.data.data.floorName + "座"

seles[1].value = res.data.data.storey

seles[2].value = res.data.data.romeTypeString

ipts[0].value = res.data.data.roomNumber

ipts[1].value = res.data.data.area

ipts[2].value = res.data.data.orientations

this.textareavalue = res.data.data.remarks

}).catch(err=>{

console.log(err)

})

},

HomeExportInter(arr){ //首页点击导出的网络请求

this.$ajax.download('/api/property/export',{'ids':arr}).then((res)=>{

console.log( res)

const link = document.createElement('a')

let blob = new Blob([res.data],{type: 'application/vnd.ms-excel'});

link.style.display = 'none'

link.href = URL.createObjectURL(blob);

const filename = res.headers["content-disposition"].split(";")[1].split("filename=")[1];

link.setAttribute('download', decodeURI(filename))

document.body.appendChild(link)

link.click()

URL.revokeObjectURL(link.href)

document.body.removeChild(link)

setTimeout(() => { //下载后完刷新页面

history.go(0)

}, 3000);

}).catch(err=>{

console.log(err)

})

},

//以下是本页面的方法

add(){ //点击option菜单的邢增

this.isAdd=true

this.isAddtatle = "新增"

this.textareavalue = ""

},

turnoff(data){ //点击关闭新增&修改弹窗

this.isAdd = data;

},

find(){ //点击搜索事件

this.findparams.floorName = null

this.findparams.romeType = null

this.findparams.storey = null

this.findparams.orderString = null

var selects = document.getElementsByClassName("homesele")

if(selects[0].value.length ==2){

if(selects[0].value=="全部"){

this.findparams.floorName = null

}else{

console.log(selects[0].value);

this.findparams.floorName = selects[0].value

}

}else if(selects[0].value.length ==3){

this.findparams.floorName = selects[0].value.substr(0,1)+'座'

this.findparams.storey = selects[0].value.slice(-1,3)

}else if(selects[0].value.length ==5){

this.findparams.floorName = selects[0].value.substr(0,1)+'座'

this.findparams.storey = -1

}

if(selects[1].value=="全部"){

this.findparams.romeType = null

// }else if(selects[1].value=="已租"){

// this.findparams.romeType = '1'

// }else if(selects[1].value=="空置"){

// this.findparams.romeType = '2'

// }

} else {

this.findparams.romeType = this.GLOBAL.getDictMapKeyByValue(this.GLOBAL.dict\_roomType, selects[1].value)

}

console.log(this.findparams);

if(selects[2].value==""){

this.findparams.orderString = null

this.findparams.pageNo = "1"

this.HouseNone()

}else{

this.findparams.orderString = selects[2].value

this.findparams.pageNo = "1"

console.log(this.findparams);

this.DateSortInter()

}

this.changeCurrentPage()

},

changeCurrentPage(){ //分页器样式重置为第一页，搜索之后调用

this.$nextTick(function(){

this.$refs['pages'].currentPage = 1;

})

},

HomeExport(){ //点击导出Excel事件

console.log(sessionStorage.getItem("downLoadIds")+","+this.DownloadId);

let idarr = (Array.from(new Set(((sessionStorage.getItem("downLoadIds")+","+this.DownloadId)).split(","||",,")))).slice(1)

console.log(idarr);

if(idarr.length!=0&&idarr[0]!=""){

this.HomeExportInter(idarr)

}else{

this.$Message["error"]({

background: true,

content:"请先选择打印数据"

});

}

sessionStorage.setItem("downLoadIds","")

},

changepage(index){ //分页器翻页请求

var selects = document.getElementsByClassName("homesele")

this.findparams.pageNo= index

if(selects[2].value==""){

this.findparams.orderString = null

this.HouseNone()

}else{

this.findparams.orderString = selects[2].value

console.log(this.findparams);

this.DateSortInter()

}

//分页之前把downloadid,保存到本地。

this.$refs.dataList.handleSelectRow()

let oldids = sessionStorage.getItem("downLoadIds")

let newids = oldids+","+this.DownloadId

sessionStorage.setItem("downLoadIds",newids)

},

homeshow(id, MainWindowListisShow, index){ //点击List修改按钮,弹出模态框

if(this.$route.path == "/home"){ //判断当前活跃路由情况下调用本页面方法

this.isAddtatle = "修改"

this.isAdd = true

this.listShowId = id

this.itemIndex = index

this.homefindInter()

}

},

updateListItemMethod(newItem, index) {

this.updateListItem = newItem

this.itemIndex = index

let roomTypeName = this.updateListItem.romeType;

this.updateListItem.romeTypeName = roomTypeName;

this.updateListItem.romeType = this.GLOBAL.getDictMapValueByName(this.GLOBAL.dict\_roomType, roomTypeName)

this.$refs.dataList.updateListItem(this.itemIndex, this.updateListItem)

console.log(newItem,index);

},

saveDownloadId(data){ //Listc传出的事件保存当前页选中的ID号

this.DownloadId = data

}

},

components:{

Options,

HomeWindow,

List

}

}

</script>

<style scoped>

.box{

background:#FFF;

}

.box .options{

background:#f3f3f3;

margin-top: 35px;

padding-bottom: 16px;

}

.options-item{

margin-right: 16px;

}

#mini{

background: #f3f3f3;

padding: 30px 0 0 0;

}

.date{

width: 120px;

}

</style>

<template>

<div class="login">

<div class="box">

<div class="left">

<img src="../../assets/img/first\_img.png" alt="" class="first-img">

</div>

<div class="right">

<img src="../../assets/img/logo.png" alt="" class="logo">

<h2>物业部</h2>

<div class="inputs">

<input type="text" placeholder="用户名" id="loginName">

<span class="span1">必须为1-12位字母+下划线或者数字</span>

<i class="iconfont icon-wode"></i>

</div>

<div class="inputs">

<input type="password" placeholder="密码不能为空" id="password">

<span id="response" v-if="isPassword">用户名或密码错误</span>

<i class="iconfont icon-mima"></i>

</div>

<Button type="success" long @click="login">登录</Button>

<h5>北京超市发连锁股份有限公司</h5>

</div>

</div>

</div>

</template>

<script>

import qs from "qs"

import Utils from '@/utils'

var body = document.body

export default {

name: 'Login',

data () {

return {

params:{'loginName':'','password':''},

isPassword:false

}

},

mounted(){

//绑定事件

window.addEventListener('keydown',this.keyDown);

},

destroyed(){

window.removeEventListener('keydown',this.keyDown,false);

},

methods:{

//下面是网络请求方法

LoginInternet(){

this.$ajax.post('/login',this.params).then((res)=>{

if(res.data.code == 1 ){

// console.log(res.data.data);

//登录成功后保存 token

sessionStorage.setItem("token", res.headers.authorization);

//登陆后保存用户昵称

sessionStorage.setItem("userName",res.data.data.loginName);

//登录后保存用户角色

sessionStorage.setItem("roleName",res.data.data.roleName);

//登录后保存路由映射表

Utils.setItem("routers",res.data.data.routers);

//登录后重置导航标签的下标，初始为0

sessionStorage.setItem("tab", 0)

this.loadDicts();

this.$router.push("/home")

}else{

this.isPassword = true

}

}).catch(err=>{

console.log(err)

})

},

//下面是页面的方法

//验证登录信息

login(){

var loginName = document.querySelector("#loginName")

var password = document.querySelector("#password")

var spans = document.querySelectorAll("span")

var reg1 = /^[A-Za-z]\w{1,12}$/

var reg2 = /\S/

var boo1 = reg1.test(loginName.value);

var boo2 = reg2.test(password.value);

if(boo1){

spans[0].style.opacity=0

}else{

spans[0].style.opacity=1

loginName.value = ""

password.value = ""

}

if(boo1&&boo2){

this.params.loginName = loginName.value

this.params.password = password.value

this.LoginInternet()

}

},

keyDown(e){

//如果是回车则执行登录方法

if(e.keyCode == 13){

this.login();

}

},

loadDicts() {

this.getDict('/dict/gender', this.GLOBAL.dict\_gender)

this.getDict('/dict/dealingSituation', this.GLOBAL.dict\_deal)

this.getDict('/dict/payMoneyWay', this.GLOBAL.dict\_pay)

this.getDict('/dict/registrationStatusType', this.GLOBAL.dict\_reg)

this.getDict('/dict/roomType', this.GLOBAL.dict\_roomType)

},

getDict(url, key) {

if(!Utils.existsKey(key)) {

this.$ajax.get(url).then((resp) => {

if(resp.data.code == 1 ) {

Utils.setItem(key, resp.data.data)

}

})

}

}

}

}

</script>

<style scoped>

@import '../../assets/font/iconfont.css';

.login{

width: 100%;

height: 100vh;

display: flex;

align-items:center;

}

.box{

width: 62.5%;

height: 67.4%;

margin: auto;

background:white;

}

.left{

float: left;

width: 50%;

height: 100%;

}

.right{

float:right;

width: 50%;

height: 100%;

}

.first-img{

width: 100%;

height: 100%;

}

.logo{

width: 64%;

height: 6.2%;

margin: 12% 18% 3% 18%;

}

h2{

text-align:center;

font-size: 28px;

letter-spacing:12px;

margin-bottom: 25%;

color: #757575;

}

input{

display: block;

width: 54%;

height: 35px;

margin: 0 20%;

border: 1px solid #ccc;

border-radius: 3px;

padding-left: 6%;

font-size: 18px;

}

input::-webkit-input-placeholder{

color: #ccc;

}

.inputs{

position: relative;

}

i{

position:absolute ;

top: 7px;

left: 21.5%;

font-weight: bold;

color: #B0B0B0;

}

button{

width: 54%;

height: 41px;

margin: 8% 20% 0;

background: #81b845;

color: white;

font-weight: 550;

font-size: 18px;

text-align: center;

border: none;

border-radius: 4px;

}

button:hover{

background: #689f38

}

h5{

font-size: 16px;

margin: 20% auto;

text-align: center;

letter-spacing:7px;

color: #848484;

}

.span1{

color: coral;

font-size: 15px;

margin: 0 0 0 20%;

opacity: 0;

}

#response{

color: coral;

font-size: 16px;

margin: 0 0 0 36%;

position: absolute;

top: 55px;

}

</style>

<template>

<div class="box">

<div>

<Options :isJurisdiction="isJurisdiction">

<button slot="butt" class="options-item" @click="add">+ 新增报修</button>

<div slot="other">

<span>显示条数：</span>

<input slot="other" type="text" style="width:40px;margin: 0 10px 0 0" v-model="Maintainparams.pageSize" @change="MaintainfindInternet">

</div>

</Options>

<div style="width:100%;height:640px;overflow: auto;">

<List ref="dataList" :message = "msg" :sonmenu="fathermenu" @lisgcheangeshow="Mainshow" :isJurisdiction="isJurisdiction"></List>

</div>

</div>

<Page :total=totalPages size="small" id="cli-page" @on-change="changepage"/>

<MainWindow v-if="isAdd" :sontitle="isAddtatle" v-on:listItemUpdated="updateListItemMethod" :itemIndex="itemIndex" @turnoff="turnoff" :MainWindowListisShow="MainWindowListisShow" :windowShowIndexObj="windowShowIndexObj" :showId="showId"></MainWindow>

</div>

</template>

<script>

import Options from "../../components/options/Options"

import List from "../../components/list/List"

import MainWindow from "./child/MainWindow"

export default {

name: 'Maintain',

data () {

return {

msg:[ //传给List的 且下面是没有网络时候的数据

{

reairName:"灯管",

floorName:"A座",

storey:"1层",

roomNumber:"102",

reairTime:"2020-1-1",

repairItemName:"张三报修",

detailedDescription:"灯管不亮",

dealingInformation:"换了灯管，回复正常",

dealingSituationString:"已处理",

id:0

}

],//传给List的表头

fathermenu:[],

Maintainparams:{ //查询参数

pageNo:'1',

pageSize:'15',

},

totalPages:10,

isAdd:false,

isAddtatle:"新增",

MainWindowListisShow:false,

showId:null,

windowShowIndexObj:null,

itemIndex: null,

updateListItem: null,

isJurisdiction:false,

roleName:sessionStorage.getItem("roleName"),

}

},

components:{

Options,

List,

MainWindow

},

created(){

this.MaintainfindInternet()

this.roleName = sessionStorage.getItem("roleName")

if(this.roleName == "系统管理员"||this.roleName == "财务人员"||this.roleName == "维修人员"){

this.isJurisdiction = true

this.fathermenu = [

{

title: '报修项目',

key: 'reairName'

},

{

title: '楼宇名称',

key: 'floorName'

},

{

title: '楼层',

key: 'storey'

},

{

title: '房间号',

key: 'roomNumber'

},

{

title: '报修时间',

key: 'reairTime'

},

{

title: '报修人',

key: 'repairItemName'

},

{

title: '详细说明',

key: 'detailedDescription'

},

{

title: '处理情况',

key: 'dealingSituationString',

render:(h,params) =>{

if(params.row.dealingSituationString ==="已处理"){

return h('span',{

'style':{

'color':'#339900'

}

},params.row.dealingSituationString)

}else{

return h('span',{

'style':{

'color':'#ff6600'

}

}, params.row.dealingSituationString)

}

}

},

{

title: '处理详情',

key: 'dealingInformation',

width:200

},

{

title: '操作',

slot: 'action',

width: 150,

align: 'center'

}

]

}else if(this.roleName == "物业人员"||this.roleName == "普通商户"){

this.isJurisdiction = true

this.fathermenu = [

{

title: '报修项目',

key: 'reairName'

},

{

title: '楼宇名称',

key: 'floorName'

},

{

title: '楼层',

key: 'storey'

},

{

title: '房间号',

key: 'roomNumber'

},

{

title: '报修时间',

key: 'reairTime'

},

{

title: '报修人',

key: 'repairItemName'

},

{

title: '详细说明',

key: 'detailedDescription'

},

{

title: '处理情况',

key: 'dealingSituationString',

render:(h,params) =>{

if(params.row.dealingSituationString ==="已处理"){

return h('span',{

'style':{

'color':'#339900'

}

},params.row.dealingSituationString)

}else{

return h('span',{

'style':{

'color':'#ff6600'

}

}, params.row.dealingSituationString)

}

}

},

{

title: '处理详情',

key: 'dealingInformation',

width:200

}

]

}

},

methods:{

//以下是网络请求的方法

MaintainfindInternet(){

this.$ajax.get('/api/repair/list',this.Maintainparams).then((res)=>{

console.log(res);

this.msg =res.data.data.data

this.totalPages = res.data.data.totalPages\*10

}).catch(err=>{

console.log(err)

})

},

// 以下是本页面的方法

add(){

this.isAdd=true

this.isAddtatle = "新增"

this.MainWindowListisShow=false

},

turnoff(data){

this.isAdd = data;

},

Mainshow(id,MainWindowListisShow,index){

this.isAdd = true

this.isAddtatle = "修改"

this.showId = id

this.itemIndex = index

this.MainWindowListisShow=MainWindowListisShow

this.windowShowIndexObj = this.msg[index]

},

changepage(index){ //分页器翻页请求

this.Maintainparams.pageNo= index

this.MaintainfindInternet()

},

updateListItemMethod(newItem, index) {

this.updateListItem = newItem

this.itemIndex = index

this.$refs.dataList.updateListItem(this.itemIndex, this.updateListItem)

}

}

}

</script>

<style scoped>

.box{

background:#FFF;

}

.options{

background:#f3f3f3;

margin-top: 35px;

padding-bottom: 16px;

}

#cli-page{

background: #f3f3f3;

padding: 30px 0 0 0;

}

</style>

<template>

<div class="shop-box">

<div class="shop\_box\_box">

<SecondMenu @showright="sonindex" :sonsecmenu ="fathersecmenu" id="SecondMenu"></SecondMenu>

<div class="shop-right">

<!-- <component :is="whitch" @outIndex="outIndex"></component> -->

<component :is="whitch" @ShopPutchangePath="ShopPutchangePath" @ShopOutchangePath="ShopOutchangePath"></component>

</div>

</div>

</div>

</template>

<script>

import SecondMenu from "../../components/second-menu/SecondMenu"

import OutStore from './child/OutStore.vue'

import PutStore from './child/PutStore'

import Store from './child/Store.vue'

import OutHistory from './child/OutHistory.vue'

import PutHistory from './child/PutHistory.vue'

import Utils from "../../utils/index.js"

export default {

name: 'Shop',

data () {

return {

fathersecmenu:this.fathersecmenuISshow(),

leftindex:0,

leftPathSrting:'store',

whitch:Store,

roleName:sessionStorage.getItem("roleName")

}

},

components: {

SecondMenu,

OutStore,

PutStore,

Store,

OutHistory,

PutHistory

},

methods:{

sonindex(index,item){

this.leftindex = index

this.leftPathSrting = item

},

ShopPutchangePath(data){

setTimeout(()=>{

this.leftPathSrting = data

},1000)

},

ShopOutchangePath(data){

console.log(data);

// setTimeout(()=>{

this.leftPathSrting = data

// },1000)

},

fathersecmenuISshow(){

this.roleName = sessionStorage.getItem("roleName")

if(this.roleName=="系统管理员"||this.roleName=="物业人员"||this.roleName=="维修人员"){

return [

{

name:'库存',

icon:"iconfont icon-ziyuan",

path:'store'

},

{

name:'入库',

icon:"iconfont icon-ruku",

path:'inSpecification'

},

{

name:'出库',

icon:"iconfont icon-chuku",

path:'outSpecificationt'

},

{

name:'出库历史',

icon:"iconfont icon-navicon-ckd",

path:'OutHistory'

},

{

name:'入库历史',

icon:"iconfont icon-navicon-rkd",

path:'PutHistory'

}

]

}else if(this.roleName=="普通商户"){

console.log("无权限");

}else if(this.roleName=="财务人员"){

return [

{

name:'库存',

icon:"iconfont icon-ziyuan",

path:'store'

}

]

}else{

return [{

name:'库存',

icon:"iconfont icon-ziyuan",

path:'store'

}]

}

}

},

watch:{

leftPathSrting(val){

if(this.roleName=="系统管理员"||this.roleName=="物业人员"||this.roleName =='维修人员'){

switch(val){

case "store" : this.whitch = Store;

break;

case "inSpecification" : this.whitch = PutStore;

break;

case "outSpecificationt" : this.whitch = OutStore;

break;

case "OutHistory" : this.whitch = OutHistory;

break;

case "PutHistory" : this.whitch = PutHistory;

break;

}

}else if(this.roleName=="普通商户"){

console.log("无权限");

}else if(this.roleName=="财务人员"){

switch(val){

case "store" : this.whitch = Store;

break;

}

}

}

}

}

</script>

<style scoped>

.shop-box{

width: 100%;

background:#f3f3f3;

}

/deep/ul{

text-align: left;

}

#SecondMenu{

margin: -20px 25px 0 0 ;

}

</style>

package com.jiexun.estate.electriccharge.controller;

import com.jiexun.estate.admin.config.SwaggerConfig;

import com.jiexun.estate.electriccharge.domain.ElectricCharge;

import com.jiexun.estate.electriccharge.service.ElectricChargeService;

import com.jiexun.estate.electriccharge.vo.ElectricChargeExportVO;

import com.jiexun.estate.electriccharge.vo.ElectricChargeUpdateVO;

import com.jiexun.estate.electriccharge.vo.ElectricChargeVO;

import com.jiexun.estate.property.util.EasyExcelUtil;

import com.jiexun.estate.property.util.FileNameUtil;

import com.jiexun.estate.property.util.ResponseEntityUtil;

import com.kzxd.support.commons.JsonResponse;

import com.kzxd.support.page.DataTablePage;

import com.kzxd.support.page.DataTableParams;

import io.swagger.annotations.Api;

import io.swagger.annotations.ApiImplicitParam;

import io.swagger.annotations.ApiImplicitParams;

import io.swagger.annotations.ApiOperation;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.data.domain.Page;

import org.springframework.http.ResponseEntity;

import org.springframework.web.bind.annotation.\*;

import javax.validation.Valid;

import java.util.ArrayList;

import java.util.List;

import java.util.Map;

/\*\*

\* @Author wzy

\* @Date 2020/7/22

\*/

@Api(tags = {SwaggerConfig.Tag\_Money})

@RestController

@RequestMapping(value = "/api/electriccharge")

public class ElectricChargeController {

@Autowired

ElectricChargeService electricchargeService;

@ApiOperation(value = "分页列出电费")

@ApiImplicitParams(value = {

@ApiImplicitParam(name = "pageNo", value = "页数，从1开始", required = true),

@ApiImplicitParam(name = "pageSize", value = "每页数目", required = true)

})

@GetMapping(value = "list")

public JsonResponse<DataTablePage<ElectricChargeVO>> listElectriccharge(DataTableParams params){

Page<ElectricCharge> allElectriccharge = electricchargeService.listElectriccharge(params.createPageable());

return JsonResponse.success(params.createPage(allElectriccharge, ElectricChargeVO::make));

}

@ApiOperation(value = "新增电费")

@ApiImplicitParams(value = {

@ApiImplicitParam(name = "electricchargeVO", value = "新增电费json", required = true,

dataType = "Electriccharge", paramType = "body")

})

@PostMapping(value = "/add")

public JsonResponse insertElelctrccharge(@Valid @RequestBody ElectricChargeVO electricchargeVO){

Integer currentElectricity = electricchargeVO.getCurrentElectricity();

Integer lastPeriodElectricity = electricchargeVO.getLastPeriodElectricity();

if(Integer.valueOf(currentElectricity)>Integer.valueOf(lastPeriodElectricity)){

electricchargeService.insertElectriccharge(electricchargeVO);

return JsonResponse.success(null);

}

return JsonResponse.error("currentElectricity>lastPeriodElectricity");

}

@ApiOperation(value = "删除电费")

@ApiImplicitParams(value = {

@ApiImplicitParam(name = "id", value = "电费ID", required = true, paramType = "path")

})

@DeleteMapping(value = "{id}")

public JsonResponse deleteElelctrccharge(@PathVariable Long id){

electricchargeService.deleteElectriccharge(id);

return JsonResponse.success(null);

}

@ApiOperation(value = "修改电费")

@ApiImplicitParams(value = {

@ApiImplicitParam(name = "electricChargeUpdateVO", value = "修改电费json", required = true,

dataType = "ElectricChargeUpdate", paramType = "body"),

@ApiImplicitParam(name = "id", value = "电费ID", required = true, paramType = "path")

})

@PostMapping(value = "{id}")

public JsonResponse updateElelctrccharge(@PathVariable Long id,@Valid @RequestBody ElectricChargeUpdateVO electricChargeUpdateVO){

electricchargeService.updateElectriccharge(electricChargeUpdateVO,id);

return JsonResponse.success(null);

}

@ApiOperation(value = "批量删除电费")

//@ApiImplicitParams(value = {@ApiImplicitParam(name = "ids", value = "电费id Array[]", required = true)})

@DeleteMapping("delete")

public JsonResponse<Void> deleteElectricchargeByIds(@RequestBody Map<String,List<Long>> map) {

electricchargeService.deleteElectricchargeByIds(map.get("ids"));

return JsonResponse.success(null);

}

@ApiOperation(value = "导出电费信息",notes = "传参格式：{'ids':[1,2,3,...]}")

@PostMapping(value = "/export/electriccharge")

public ResponseEntity<byte[]> exportElectriccharge(@RequestBody Map<String,List<Long>> map) {

List<ElectricCharge> electricChargeList = electricchargeService.exportElectricChargeRignt(map.get("ids"));

List<ElectricChargeExportVO> excelData = new ArrayList<ElectricChargeExportVO>();

for(ElectricCharge electricCharge : electricChargeList){

excelData.add(ElectricChargeExportVO.make(electricCharge));

}

//List<ElectricChargeExportVO> excelData = electricchargeService.exportElectriccharge(ids);

byte[] byteArray = EasyExcelUtil.writeDataToByteArray(excelData, ElectricChargeExportVO.class);

String fileName = FileNameUtil.generateFileName("电费信息");

return ResponseEntityUtil.downloadFile(byteArray, fileName);

}

@ApiOperation(value = "根据ID查找电费信息")

@ApiImplicitParams(value = {

@ApiImplicitParam(name = "id", value = "电费ID", required = true, paramType = "path")

})

@GetMapping(value = "{id}")

public JsonResponse<ElectricChargeVO> getElectricChargeById(@PathVariable Long id){

ElectricChargeVO selectElectricChargeVO = electricchargeService.getElectricChargeById(id).map(ElectricChargeVO::make).orElse(null);

return JsonResponse.success(selectElectricChargeVO);

}

}

package com.jiexun.estate.electriccharge.domain;

import com.kzxd.support.jpa.BaseEntity;

import lombok.AllArgsConstructor;

import lombok.Data;

import lombok.NoArgsConstructor;

import javax.persistence.\*;

/\*\*

\* @Author yangke

\* @Date 2020/7/15 22:47

\*/

@Data

@Entity

@NoArgsConstructor

@AllArgsConstructor

@Table(name="usm\_electric\_charge")

public class ElectricCharge extends BaseEntity {

/\*\*

\*楼宇名称

\*/

@Column(name="floor\_name",length = 15)

private String floorName;

/\*\*

\*楼层

\*/

@Column(name="storey")

private Integer storey;

/\*\*

\*房间号

\*/

@Column(name="room\_number",length = 10)

private String roomNumber;

/\*\*

\*公司名称

\*/

@Column(name = "corporate\_name")

private String corporateName;

/\*\*

\*面积

\*/

@Column(name="area",length = 10)

private String area;

/\*\*

\*上期电表数

\*/

@Column(name="last\_period\_electricity",length = 10)

private Integer lastPeriodElectricity;

/\*\*

\*本期电表数

\*/

@Column(name="current\_electricity",length = 10)

private Integer currentElectricity;

/\*\*

\*电表赔率

\*/

@Column(name="electricity\_odds")

private Integer electricityOdds;

/\*\*

\*单价

\*/

@Column(name ="unit\_price" )

private Integer unitPrice;

/\*\*

\*总金额

\*/

@Column(name = "sum\_money")

private Integer sumMoney;

/\*\*

\*备注

\*/

@Column(name = "remarks" ,length = 100)

private String remarks;

}

package com.jiexun.estate.electriccharge.repository;

import com.jiexun.estate.electriccharge.domain.ElectricCharge;

import com.kzxd.support.jpa.BaseRepository;

import org.springframework.data.domain.Page;

import org.springframework.data.domain.Pageable;

/\*\*

\* @Author wzy

\* @Date 2020/7/21

\*/

public interface ElectricChargeRepository extends BaseRepository<ElectricCharge> {

Page<ElectricCharge> findByOrderByIdDesc(Pageable pageable);

}

package com.jiexun.estate.electriccharge.service.impl;

import com.jiexun.estate.electriccharge.domain.ElectricCharge;

import com.jiexun.estate.electriccharge.repository.ElectricChargeRepository;

import com.jiexun.estate.electriccharge.service.ElectricChargeService;

import com.jiexun.estate.electriccharge.vo.ElectricChargeUpdateVO;

import com.jiexun.estate.electriccharge.vo.ElectricChargeVO;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.data.domain.Page;

import org.springframework.data.domain.Pageable;

import org.springframework.stereotype.Service;

import java.util.ArrayList;

import java.util.List;

import java.util.Optional;

/\*\*

\* @Author wzy

\* @Date 2020/7/22

\*/

@Service

public class ElectricChargeServiceImpl implements ElectricChargeService {

@Autowired

ElectricChargeRepository electricChargeRepository;

@Override

public Page<ElectricCharge> listElectriccharge(Pageable pageable) {

return electricChargeRepository.findByOrderByIdDesc(pageable);

}

@Override

public Optional<ElectricCharge> getElectricchargeById(Long Id) {

return electricChargeRepository.findById(Id);

}

/\*\*

\* 新增电费

\* @param electricchargeVO

\*/

@Override

public void insertElectriccharge(ElectricChargeVO electricchargeVO) {

ElectricCharge electriccharge = electricchargeVO.newElectricchargeEntity();

electricChargeRepository.save(electriccharge);

}

/\*\*

\* 删除电费

\* @param id

\*/

@Override

public void deleteElectriccharge(Long id) {

electricChargeRepository.deleteById(id);

}

/\*\*

\* 修改电费

\* @param electricChargeUpdateVO,Id

\*/

@Override

public void updateElectriccharge(ElectricChargeUpdateVO electricChargeUpdateVO, Long Id) {

getElectricchargeById(Id).ifPresent(electriccharge -> {

Integer sumMoney = (Integer.valueOf(electricChargeUpdateVO.getCurrentElectricity())-Integer.valueOf(electricChargeUpdateVO.getLastPeriodElectricity()))

\*electricChargeUpdateVO.getElectricityOdds()\*electricChargeUpdateVO.getUnitPrice();

electriccharge.setSumMoney(sumMoney);

electriccharge.setRemarks(electricChargeUpdateVO.getRemarks());

electriccharge.setUnitPrice(electricChargeUpdateVO.getUnitPrice());

electriccharge.setElectricityOdds(electricChargeUpdateVO.getElectricityOdds());

electriccharge.setCurrentElectricity(electricChargeUpdateVO.getCurrentElectricity());

electriccharge.setLastPeriodElectricity(electricChargeUpdateVO.getLastPeriodElectricity());

electricChargeRepository.save(electriccharge);

});

}

/\*\*

\* 批量删除

\* @param ids

\*/

@Override

public void deleteElectricchargeByIds(List<Long> ids) {

List<ElectricCharge> ElectricCharges = electricChargeRepository.findAllById(ids);

if (null != ElectricCharges && !ElectricCharges.isEmpty()) {

electricChargeRepository.deleteAll(ElectricCharges);

}

}

/\*\*

\* 根据ID查合同信息

\* @param id

\* @return

\*/

@Override

public Optional<ElectricCharge> getElectricChargeById(Long id) {

return electricChargeRepository.findById(id);

}

/\*\*

\* 根据导出电费信息

\* @return

\*/

@Override

public List<ElectricCharge> exportElectricChargeRignt(List<Long> ids) {

if(ids.size()>0){

List<ElectricCharge> electricChargeList = electricChargeRepository.findAllById(ids);

/\*List<ElectricChargeExportVO> voList = new ArrayList<>(electricChargeList.size());

for(ElectricCharge electriccharge: electricChargeList){

ElectricChargeExportVO electricchargeExportVO = ElectricChargeExportVO.make(electriccharge);

voList.add(electricchargeExportVO);

}\*/

return electricChargeList;

}else{

return new ArrayList<ElectricCharge>();

}

}

}

package com.jiexun.estate.electriccharge.vo;

import com.alibaba.excel.annotation.ExcelProperty;

import com.jiexun.estate.electriccharge.domain.ElectricCharge;

import io.swagger.annotations.ApiModelProperty;

import lombok.Data;

import java.io.Serializable;

/\*\*

\* @Author wzy

\* @Date 2020/7/24

\*/

@Data

public class ElectricChargeExportVO implements Serializable {

@ApiModelProperty("楼宇名称")

@ExcelProperty("楼宇名称")

private String floorName;

@ApiModelProperty("楼层")

@ExcelProperty("楼层")

private Integer storey;

@ApiModelProperty("房间号")

@ExcelProperty("房间号")

private String roomNumber;

@ApiModelProperty("公司名称")

@ExcelProperty("公司名称")

private String corporateName;

@ApiModelProperty("面积")

@ExcelProperty("面积")

private String area;

@ApiModelProperty("上期电表数")

@ExcelProperty("上期电表数")

private Double lastPeriodElectricity;

@ApiModelProperty("本期电表数")

@ExcelProperty("本期电表数")

private Double currentElectricity;

@ApiModelProperty("电表倍率")

@ExcelProperty("电表倍率")

private Double electricityOdds;

@ApiModelProperty("单价")

@ExcelProperty("单价")

private Double unitPrice;

@ApiModelProperty("总金额")

@ExcelProperty("总金额")

private Double sumMoney;

@ApiModelProperty("备注")

@ExcelProperty("备注")

private String remarks;

public static ElectricChargeExportVO make(ElectricCharge electriccharge){

ElectricChargeExportVO vo = new ElectricChargeExportVO();

vo.setFloorName(electriccharge.getFloorName());

vo.setStorey(electriccharge.getStorey());

vo.setRoomNumber(electriccharge.getRoomNumber());

vo.setCorporateName(electriccharge.getCorporateName());

vo.setArea(electriccharge.getArea());

vo.setLastPeriodElectricity(electriccharge.getLastPeriodElectricity()\*0.01);

vo.setCurrentElectricity(electriccharge.getCurrentElectricity()\*0.01);

vo.setElectricityOdds(electriccharge.getElectricityOdds()\*0.01);

vo.setUnitPrice(electriccharge.getUnitPrice()\*0.01);

vo.setSumMoney(electriccharge.getSumMoney()\*0.0001);

vo.setRemarks(electriccharge.getRemarks());

return vo;

}

}