<template>

<div id="placeholder">

</div>

</template>

<script>

import './api.js'

export default {

name: 'ExcelWindow',

components: {},

props: {

documentType: {

type: String,

default: 'cell'

},

fileUrl: {

type: String,

default: ''

},

mode: {

type: String,

default: 'edit' //view

},

callbackUrl: {

type: String,

default: ''

},

title:{

type:String,

default:''

},

iframeOrigin:{

type:String,

default:''

},

token:{

type:String,

default:''

},

user:{

type:Object,

default:()=>{

return {

name:'RSH',

id:"01"

}

}

},

excelKey:{

type:String,

default:''

}

},

data() {

return {

docEditor:''

};

},

created() {},

mounted() {

console.log(' this.config', this.config)

var docEditor = this.docEditor = new DocsAPI.DocEditor("placeholder", this.config)

// setTimeout(()=>{

// console.log('刷新')

// docEditor.setReferenceData({

// "url": this.config.document.url

// })

// },8000)

// docEditor.serviceCommand

// console.log('this.docEditor.callCommand',docEditor.serviceCommand)

// var connector = docEditor.createConnector()

// console.log('docEditor',docEditor)

// docEditor.processSaveResult(()=>{

// console.log('argu',arguments)

// })

},

watch: {

mode(val){

if(val == 'edit'){

// this.docEditor.destroyEditor()

// console.log('this.config',this.config)

this.docEditor = new DocsAPI.DocEditor("placeholder", this.config);

}

},

fileUrl(val){

console.log('fileUrl-this.config',this.config)

this.docEditor = new DocsAPI.DocEditor("placeholder", this.config);

},

// config(val) { // 重命名文件，生成新实例不生效

// console.log('config=val',val)

// // console.log('config=val',this.config)

// this.docEditor.destroyEditor();

// this.docEditor = new DocsAPI.DocEditor("placeholder", this.config)

// // if (val) {

// // setTimeout(x=>{

// // console.log('time')

// // console.log('this.config',this.config)

// // this.docEditor = new DocsAPI.DocEditor("placeholder", this.config)

// // },1000)

// // }

// }

},

computed: {

config() {

return {

iframeOrigin:this.iframeOrigin,

documentType: this.documentType,

token: this.token,

type: "desktop",

height: "100%",

// height: "95%",

width: "100%",

document: {

key: this.excelKey,

title: this.title,

url: this.fileUrl,

info:{

"owner": "RSH",

}

},

permissions: {

"chat": false,

"comment": false,

"edit": true,

"protect":false,

},

editorConfig: {

"lang": "zh",

"location": "",

"mode": this.mode,

"customization": {

// "anonymous": {

// "request": false,

// "label": "Guest"

// },

// 自动保存可以关闭，常规ctrl+s更好用

// "autosave": false,

"autosave": true,

"comments": true,

"compactHeader": true,

"compatibleFeatures": false,

"forcesave": true,

// "forcesave": false,

"hideRulers": false,

"compactToolbar": false,

"toolbarNoTabs": true,

"help": false,

// "integrationMode": "embed",

"logo": {

"image": "https://cloud.runshihua.com/rsh/static/img/guanwang/logo\_white.png",

"imageDark": "https://cloud.runshihua.com/rsh/static/img/guanwang/logo\_white.png",

},

"macros": false,

"macrosMode": "警告",

"mentionShare": false,

"mobileForceView": true,

"plugins": false,

"toolbarHideFileName": false,

"toolbarNoTabs": false,

"showReviewChanges":false,

// "uiTheme": "theme-dark",

"unit": "厘米",

"zoom": 100,

"hideRightMenu": true,

"features": {

"spellcheck": {

"mode": false

}

}

},

"callbackUrl": this.callbackUrl,//"http://192.168.1.113:9628/information/center/webRelease/savess?id=40d065ab-13db-4683-8c30-e54efdad4ed5&sid=40d065ab-13db-4683-8c30-e54efdad4ed5",

"user": {

"name": this.user.name,

"id": this.user.id,

}

},

"events": {

"onDocumentReady": this.onDocumentReady,

"onDocumentStateChange":this.onDocumentStateChange,

"onRequestSaveAs":this.onRequestSaveAs,

"onRequestSelectDocument": () => { }

},

}

}

},

methods: {

init(){

console.log('init')

console.log(' this.config2', this.config)

this.docEditor = new DocsAPI.DocEditor("placeholder", this.config)

},

destroy(){

console.log('destroy')

this.docEditor.destroyEditor();

},

onDocumentReady(){

// console.log('ar--',arguments)

console.log('文档加载完成')

console.log('this.docEditor',this.docEditor)

// this.docEditor.serviceCommand('refresh',{})

// window.connector = this.docEditor.createConnector();

// console.log(111)

// console.log('window.connector',window.connector)

// setTimeout(()=>{

// this.docEditor.destroyEditor()

// },5000)

},

onDocumentStateChange(data){

this.$emit('onDocumentStateChange',data)

},

onRequestSaveAs(){

console.log('onRequestSaveAs--',arguments)

},

generateRandomId(length) {

var result = '';

var characters = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789';

var charactersLength = characters.length;

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

result += characters.charAt(Math.floor(Math.random() \* charactersLength));

}

return result;

},

},

};

</script>

<style lang="scss" scoped>

@import './excel.css';

</style>