<template xmlns:background-color="http://www.w3.org/1999/xhtml">  
 <a-row :gutter="10">  
 <a-col :md="12" :sm="24">  
 <a-card :bordered="false">  
  
 <!-- 按钮操作区域 -->  
 <a-row style="margin: 0 0 0 14px" class="table-operator">  
 <a-button @click="handleAdd(1)" type="primary">添加部门</a-button>  
 <a-button @click="handleAdd(2)" type="primary">添加下级</a-button>  
 <a-button type="primary" icon="download" @click="handleExportXls('部门信息')">导出</a-button>  
 <a-upload name="file" :showUploadList="false" :multiple="false" :headers="tokenHeader" :action="importExcelUrl" @change="handleImportExcel">  
 <a-button type="primary" icon="import">导入</a-button>  
 </a-upload>  
 <j-third-app-button biz-type="depart" :selected-row-keys="selectedRowKeys" syncToApp @sync-finally="onSyncFinally"/>  
 <a-button title="删除多条数据" @click="batchDel" type="default">批量删除</a-button>  
 </a-row>  
 <div style="background: #fff;padding-left:16px;height: 100%; margin-top: 5px">  
 <a-alert type="info" :showIcon="true">  
 <div slot="message">  
 当前选择：<span v-if="this.currSelected.title">{{ getCurrSelectedTitle() }}</span>  
 <a v-if="this.currSelected.title" style="margin-left: 10px" @click="onClearSelected">取消选择</a>  
 </div>  
 </a-alert>  
 <a-input-search @search="onSearch" style="width:100%;margin-top: 10px" placeholder="请输入部门名称"/>  
 <!-- 树-->  
 <div>  
 <a-empty v-if="departTree.length===0" description="暂无部门" style="margin-top: 8px;"/>  
 <template v-else>  
 <a-dropdown :trigger="[this.dropTrigger]" @visibleChange="dropStatus">  
 <span style="user-select: none">  
 <a-tree  
 v-if="treeLoading"  
 checkable  
 multiple  
 @select="onSelect"  
 @check="onCheck"  
 @rightClick="rightHandle"  
 :selectedKeys="selectedKeys"  
 :checkedKeys="checkedKeys"  
 :treeData="departTree"  
 :checkStrictly="checkStrictly"  
 :expandedKeys.sync="iExpandedKeys"  
 :load-data="loadSubTree"  
 @expand="onExpand"/>  
 </span>  
 <a-menu slot="overlay">  
 <a-menu-item @click="handleAdd(3)" key="1">添加</a-menu-item>  
 <a-menu-item @click="handleDelete" key="2">删除</a-menu-item>  
 <a-menu-item @click="closeDrop" key="3">取消</a-menu-item>  
 </a-menu>  
 </a-dropdown>  
 </template>  
 </div>  
 </div>  
 </a-card>  
 <!---- author:os\_chengtgen -- date:20190827 -- for:切换父子勾选模式 =======------>  
 <div class="drawer-bootom-button">  
 <a-dropdown :trigger="['click']" placement="topCenter">  
 <a-menu slot="overlay">  
 <a-menu-item key="1" @click="switchCheckStrictly(1)">父子关联</a-menu-item>  
 <a-menu-item key="2" @click="switchCheckStrictly(2)">取消关联</a-menu-item>  
 <a-menu-item key="3" @click="checkALL">全部勾选</a-menu-item>  
 <a-menu-item key="4" @click="cancelCheckALL">取消全选</a-menu-item>  
 <a-menu-item key="5" @click="expandAll">展开所有</a-menu-item>  
 <a-menu-item key="6" @click="closeAll">合并所有</a-menu-item>  
 </a-menu>  
 <a-button>  
 树操作 <a-icon type="up" />  
 </a-button>  
 </a-dropdown>  
 </div>  
 <!---- author:os\_chengtgen -- date:20190827 -- for:切换父子勾选模式 =======------>  
 </a-col>  
 <a-col :md="12" :sm="24">  
 <a-tabs defaultActiveKey="1">  
 <a-tab-pane tab="基本信息" key="1" >  
 <a-card :bordered="false" v-if="selectedKeys.length>0">  
 <a-form-model ref="form" :model="model" :rules="validatorRules">  
 <a-form-model-item :labelCol="labelCol" :wrapperCol="wrapperCol" label="机构名称" prop="departName" >  
 <a-input placeholder="请输入机构/部门名称" v-model="model.departName"/>  
 </a-form-model-item>  
 <a-form-model-item :labelCol="labelCol" :wrapperCol="wrapperCol" label="上级部门" prop="parentId">  
 <a-tree-select style="width:100%" :dropdownStyle="{maxHeight:'200px',overflow:'auto'}" :treeData="treeData" :disabled="disable" v-model="model.parentId" placeholder="无">  
 </a-tree-select>  
 </a-form-model-item>  
 <a-form-model-item :labelCol="labelCol" :wrapperCol="wrapperCol" label="机构编码" prop="orgCode">  
 <a-input disabled placeholder="请输入机构编码" v-model="model.orgCode"/>  
 </a-form-model-item>  
 <a-form-model-item :labelCol="labelCol" :wrapperCol="wrapperCol" label="机构类型" prop="orgCategory">  
 <template v-if="orgCategoryDisabled">  
 <a-radio-group v-model="model.orgCategory" placeholder="请选择机构类型">  
 <a-radio value="1">  
 公司  
 </a-radio>  
 </a-radio-group>  
 </template>  
 <template v-else>  
 <a-radio-group v-model="model.orgCategory" placeholder="请选择机构类型">  
 <a-radio value="2">  
 部门  
 </a-radio>  
 <a-radio value="3">  
 岗位  
 </a-radio>  
 </a-radio-group>  
 </template>  
 </a-form-model-item>  
 <a-form-model-item :labelCol="labelCol" :wrapperCol="wrapperCol" label="排序" prop="departOrder">  
 <a-input-number v-model="model.departOrder"/>  
 </a-form-model-item>  
 <a-form-model-item :labelCol="labelCol" :wrapperCol="wrapperCol" label="手机号" prop="mobile">  
 <a-input placeholder="请输入手机号" v-model="model.mobile"/>  
 </a-form-model-item>  
 <a-form-model-item :labelCol="labelCol" :wrapperCol="wrapperCol" label="地址" prop="address">  
 <a-input placeholder="请输入地址" v-model="model.address"/>  
 </a-form-model-item>  
 <a-form-model-item :labelCol="labelCol" :wrapperCol="wrapperCol" label="备注" prop="memo">  
 <a-textarea placeholder="请输入备注" v-model="model.memo"/>  
 </a-form-model-item>  
 </a-form-model>  
 <div class="anty-form-btn">  
 <a-button @click="emptyCurrForm" type="default" htmlType="button" icon="sync">重置</a-button>  
 <a-button @click="submitCurrForm" type="primary" htmlType="button" icon="form">保存</a-button>  
 </div>  
 </a-card>  
 <a-card v-else >  
 <a-empty>  
 <span slot="description"> 请先选择一个部门! </span>  
 </a-empty>  
 </a-card>  
 </a-tab-pane>  
 <a-tab-pane tab="部门权限" key="2" forceRender>  
 <depart-auth-modal ref="departAuth"/>  
 </a-tab-pane>  
 </a-tabs>  
   
 </a-col>  
 <depart-modal ref="departModal" @ok="loadTree"></depart-modal>  
 </a-row>  
</template>  
<script>  
 import DepartModal from './modules/DepartModal'  
 import { deleteByDepartId, queryDepartTreeSync, searchByKeywords } from '@/api/api'  
 import { deleteAction, httpAction } from '@/api/manage'  
 import { JeecgListMixin } from '@/mixins/JeecgListMixin'  
 import DepartAuthModal from './modules/DepartAuthModal'  
 import { cloneObject } from '@/utils/util'  
 import JThirdAppButton from '@comp/jeecgbiz/thirdApp/JThirdAppButton'  
 // 表头  
 const columns = [  
 {  
 title: '机构名称',  
 dataIndex: 'departName'  
 },  
 {  
 title: '机构类型',  
 align: 'center',  
 dataIndex: 'orgType'  
 },  
 {  
 title: '机构编码',  
 dataIndex: 'orgCode',  
 },  
 {  
 title: '手机号',  
 dataIndex: 'mobile'  
 },  
 {  
 title: '传真',  
 dataIndex: 'fax'  
 },  
 {  
 title: '地址',  
 dataIndex: 'address'  
 },  
 {  
 title: '排序',  
 align: 'center',  
 dataIndex: 'departOrder'  
 },  
 {  
 title: '操作',  
 align: 'center',  
 dataIndex: 'action',  
 scopedSlots: {customRender: 'action'}  
 }  
 ]  
 export default {  
 name: 'DepartList',  
 mixins: [JeecgListMixin],  
 components: {  
 JThirdAppButton,  
 DepartAuthModal,  
 DepartModal  
 },  
 data() {  
 return {  
 iExpandedKeys: [],  
 treeLoading: true,  
 autoExpandParent: false,  
 currFlowId: '',  
 currFlowName: '',  
 disable: true,  
 treeData: [],  
 visible: false,  
 departTree: [],  
 departTreeAll: [],  
 loadedKeys: [],  
 allIds: [],  
 rightClickSelectedKey: '',  
 rightClickSelectedOrgCode: '',  
 hiding: true,  
 model: {},  
 dropTrigger: '',  
 depart: {},  
 columns: columns,  
 disableSubmit: false,  
 checkedKeys: [],  
 selectedKeys: [],  
 autoIncr: 1,  
 currSelected: {},  
  
 allTreeKeys:[],  
 loadTreeKeys:[],  
 checkStrictly: true,  
 labelCol: {  
 xs: {span: 24},  
 sm: {span: 5}  
 },  
 wrapperCol: {  
 xs: {span: 24},  
 sm: {span: 16}  
 },  
 graphDatasource: {  
 nodes: [],  
 edges: []  
 },  
 validatorRules: {  
 departName: [{required: true, message: '请输入机构/部门名称!'}],  
 orgCode: [{required: true, message: '请输入机构编码!'}],  
 orgCategory: [{required: true, message: '请输入机构类型!'}],  
 mobile: [{validator: this.validateMobile}]  
 },  
 url: {  
 delete: '/sys/sysDepart/delete',  
 edit: '/sys/sysDepart/edit',  
 deleteBatch: '/sys/sysDepart/deleteBatch',  
 exportXlsUrl: "sys/sysDepart/exportXls",  
 importExcelUrl: "sys/sysDepart/importExcel",  
 },  
 orgCategoryDisabled:false,  
 }  
 },  
 computed: {  
 importExcelUrl: function () {  
 return `${window.\_CONFIG['domianURL']}/${this.url.importExcelUrl}`;  
 }  
 },  
 methods: {  
 loadData() {  
 this.refresh();  
 },  
 loadTree() {  
 var that = this  
 //上级部门树信息  
 that.treeData = []  
 //保存的树信息，搜索是需要用到  
 that.departTreeAll = []  
 //所有的树信息  
 that.departTree = []  
 //所有的树节点key信息  
 that.allIds = []  
   
 that.iExpandedKeys = []  
 //update-begin---author:wangshuai ---date:20220105 for：[JTC-364]sqlserver 部门导入导入失败，部门树数据丢失------------  
 //部门树v-if用到了loading,和上传loading冲突了，换一个名称  
 that.treeLoading = false  
 //update-end---author:wangshuai ---date:20220105 for：[JTC-364]sqlserver 部门导入导入失败，部门树数据丢失------------  
 queryDepartTreeSync().then((res) => {  
 if (res.success) {  
 this.allTreeKeys = [];  
 for (let i = 0; i < res.result.length; i++) {  
 let temp = res.result[i]  
 that.treeData.push(temp)  
 that.departTreeAll.push(temp)  
 that.departTree.push(temp)  
 that.allIds.push(temp.key)  
 that.allTreeKeys.push(temp.key)  
 if(that.loadTreeKeys.indexOf(temp.key)>=0){  
 that.iExpandedKeys.push(temp.key)  
 }  
 }  
 that.$nextTick(()=>{  
 //部门树v-if用到了loading,和上传loading冲突了，换一个名称  
 that.treeLoading = true  
 })  
 }  
 })  
 },  
 loadSubTree(treeNode) {  
 var that = this;  
 return new Promise(resolve => {  
 queryDepartTreeSync({pid:treeNode.dataRef.id}).then((res) => {  
 if (res.success) {  
 //判断chidlren是否为空，并修改isLeaf属性值  
 if(res.result.length == 0){  
 treeNode.dataRef['isLeaf']=true  
 return;  
 }else{  
 treeNode.dataRef['children']= res.result;  
 }  
 for (let i = 0; i < res.result.length; i++) {  
 let temp = res.result[i]  
 that.allIds.push(temp.key)  
 if(that.loadTreeKeys.indexOf(temp.key)>0){  
 that.iExpandedKeys.push(temp.key)  
 }  
 }  
 }  
 })  
 resolve();  
 });  
 //保存全部部门信息，方便后面搜索使用  
 that.departTreeAll=that.departTree  
 },  
 refresh() {  
 //部门树v-if用到了loading,和上传loading冲突了，换一个名称  
 this.treeLoading = true  
 this.loadTree()  
 },  
 // 右键操作方法  
 rightHandle(node) {  
 this.dropTrigger = 'contextmenu'  
 console.log(node.node.eventKey)  
 this.rightClickSelectedKey = node.node.eventKey  
 this.rightClickSelectedOrgCode = node.node.dataRef.orgCode  
 },  
 onExpand(expandedKeys) {  
 console.log('onExpand', expandedKeys)  
 this.iExpandedKeys = expandedKeys  
 this.autoExpandParent = false  
 this.allTreeKeys=expandedKeys  
 this.loadTreeKeys=expandedKeys  
 },  
 backFlowList() {  
 this.$router.back(-1)  
 },  
 // 右键点击下拉框改变事件  
 dropStatus(visible) {  
 if (visible == false) {  
 this.dropTrigger = ''  
 }  
 },  
 // 右键店家下拉关闭下拉框  
 closeDrop() {  
 this.dropTrigger = ''  
 },  
 addRootNode() {  
 this.$refs.nodeModal.add(this.currFlowId, '')  
 },  
 batchDel: function () {  
 console.log(this.checkedKeys)  
 if (this.checkedKeys.length <= 0) {  
 this.$message.warning('请选择一条记录！')  
 } else {  
 var ids = ''  
 for (var a = 0; a < this.checkedKeys.length; a++) {  
 ids += this.checkedKeys[a]+ ','  
 }  
 var that = this  
 this.$confirm({  
 title: '确认删除',  
 content: '确定要删除所选中的 ' + this.checkedKeys.length + ' 条数据，以及子节点数据吗?',  
 onOk: function () {  
 deleteAction(that.url.deleteBatch, {ids: ids}).then((res) => {  
 if (res.success) {  
 that.$message.success(res.message)  
 that.loadTree()  
 that.onClearSelected()  
 } else {  
 that.$message.warning(res.message)  
 }  
 })  
 }  
 })  
 }  
 },  
 onSearch(value) {  
 let that = this  
 if (value) {  
 searchByKeywords({keyWord: value}).then((res) => {  
 if (res.success) {  
 that.departTree = []  
 for (let i = 0; i < res.result.length; i++) {  
 let temp = res.result[i]  
 that.departTree.push(temp)  
 }  
 } else {  
 that.$message.warning(res.message)  
 }  
 })  
 } else {  
 that.departTree=that.departTreeAll  
 }  
   
 },  
 nodeModalOk() {  
 this.loadTree()  
 },  
 nodeModalClose() {  
 },  
 hide() {  
 this.visible = false  
 },  
 onCheck(checkedKeys, info) {  
 console.log('onCheck', checkedKeys, info)  
 this.hiding = false  
 //this.checkedKeys = checkedKeys.checked  
 // <!---- author:os\_chengtgen -- date:20190827 -- for:切换父子勾选模式 =======------>  
 if(this.checkStrictly){  
 this.checkedKeys = checkedKeys.checked;  
 }else{  
 this.checkedKeys = checkedKeys  
 }  
 // <!---- author:os\_chengtgen -- date:20190827 -- for:切换父子勾选模式 =======------>  
 },  
 onSelect(selectedKeys, e) {  
 console.log('selected', selectedKeys, e)  
 this.hiding = false  
 let record = e.node.dataRef  
 console.log('onSelect-record', record)  
 this.currSelected = Object.assign({}, record)  
 this.model = this.currSelected  
 this.selectedKeys = [record.key]  
 this.model.parentId = record.parentId  
 this.setValuesToForm(record)  
 this.$refs.departAuth.show(record.id);  
   
 },  
 // 触发onSelect事件时,为部门树右侧的form表单赋值  
 setValuesToForm(record) {  
 if(record.orgCategory == '1'){  
 this.orgCategoryDisabled = true;  
 }else{  
 this.orgCategoryDisabled = false;  
 }  
 this.$nextTick(() => {  
 this.model = cloneObject(record)  
 })  
 },  
 getCurrSelectedTitle() {  
 return !this.currSelected.title ? '' : this.currSelected.title  
 },  
 onClearSelected() {  
 this.hiding = true  
 this.checkedKeys = []  
 this.currSelected = {}  
 this.model = cloneObject(this.currSelected)  
 this.selectedKeys = []  
 this.$refs.departAuth.departId = ''  
 },  
 handleNodeTypeChange(val) {  
 this.currSelected.nodeType = val  
 },  
 notifyTriggerTypeChange(value) {  
 this.currSelected.notifyTriggerType = value  
 },  
 receiptTriggerTypeChange(value) {  
 this.currSelected.receiptTriggerType = value  
 },  
 submitCurrForm() {  
 this.$refs.form.validate((ok, err) => {  
 if (ok) {  
 if (!this.currSelected.id) {  
 this.$message.warning('请点击选择要修改部门!')  
 return  
 }  
   
 let formData = Object.assign(this.currSelected, this.model)  
 console.log('Received values of form: ', formData)  
 httpAction(this.url.edit, formData, 'put').then((res) => {  
 if (res.success) {  
 this.$message.success('保存成功!')  
 this.loadTree()  
 } else {  
 this.$message.error(res.message)  
 }  
 })  
 }  
 })  
 },  
 emptyCurrForm() {  
 this.model = this.currSelected  
 },  
 nodeSettingFormSubmit() {  
 this.$refs.form.validate((ok, err) => {  
 if (ok) {  
 console.log('Received values of form: ', this.model)  
 }  
 })  
 },  
 openSelect() {  
 this.$refs.sysDirectiveModal.show()  
 },  
 handleAdd(num) {  
 if (num == 1) {  
 this.$refs.departModal.add()  
 this.$refs.departModal.title = '新增'  
 } else if (num == 2) {  
 let key = this.currSelected.key  
 if (!key) {  
 this.$message.warning('请先点击选中上级部门！')  
 return false  
 }  
 this.$refs.departModal.add(this.selectedKeys[0])  
 this.$refs.departModal.title = '新增'  
 } else {  
 this.$refs.departModal.add(this.rightClickSelectedKey)  
 this.$refs.departModal.title = '新增'  
 }  
 },  
 handleDelete() {  
 var that = this  
 this.$confirm({  
 title: '确认删除',  
 content: '确定要删除此部门以及子节点数据吗?',  
 onOk: function () {  
 deleteByDepartId({id: that.rightClickSelectedKey}).then((resp) => {  
 if (resp.success) {  
 //删除成功后，去除已选中中的数据  
 that.checkedKeys.splice(that.checkedKeys.findIndex(key => key === that.rightClickSelectedKey), 1);  
 that.$message.success('删除成功!')  
 that.loadTree()  
 //删除后同步清空右侧基本信息内容  
 let orgCode = that.model.orgCode  
 if(orgCode && orgCode === that.rightClickSelectedOrgCode){  
 that.onClearSelected()  
 }  
 } else {  
 that.$message.warning('删除失败!')  
 }  
 })  
 }  
 })  
 },  
 selectDirectiveOk(record) {  
 console.log('选中指令数据', record)  
 this.nodeSettingForm.setFieldsValue({directiveCode: record.directiveCode})  
 this.currSelected.sysCode = record.sysCode  
 },  
 getFlowGraphData(node) {  
 this.graphDatasource.nodes.push({  
 id: node.id,  
 text: node.flowNodeName  
 })  
 if (node.children.length > 0) {  
 for (let a = 0; a < node.children.length; a++) {  
 let temp = node.children[a]  
 this.graphDatasource.edges.push({  
 source: node.id,  
 target: temp.id  
 })  
 this.getFlowGraphData(temp)  
 }  
 }  
 },  
 // <!---- author:os\_chengtgen -- date:20190827 -- for:切换父子勾选模式 =======------>  
 expandAll () {  
 this.iExpandedKeys = this.allTreeKeys  
 //this.loadTree()  
 },  
 closeAll () {  
 this.iExpandedKeys = []  
 },  
 checkALL () {  
 this.checkStriccheckStrictlytly = false  
 //this.checkedKeys = this.allTreeKeys  
 this.checkedKeys = this.allIds  
 },  
 cancelCheckALL () {  
 //this.checkedKeys = this.defaultCheckedKeys  
 this.checkedKeys = []  
 },  
 switchCheckStrictly (v) {  
 if(v==1){  
 this.checkStrictly = false  
 }else if(v==2){  
 this.checkStrictly = true  
 }  
 },  
 getAllKeys(node) {  
 // console.log('node',node);  
 this.allTreeKeys.push(node.key)  
 if (node.children && node.children.length > 0) {  
 for (let a = 0; a < node.children.length; a++) {  
 this.getAllKeys(node.children[a])  
 }  
 }  
 },  
 // <!---- author:os\_chengtgen -- date:20190827 -- for:切换父子勾选模式 =======------>  
   
 // 验证手机号  
 validateMobile(rule,value,callback){  
 if (!value || new RegExp(/^1([38][0-9]|4[579]|5[0-3,5-9]|6[6]|7[0135678]|9[89])\d{8}$/).test(value)){  
 callback();  
 }else{  
 callback("您的手机号码格式不正确!");  
 }  
   
 },  
 onSyncFinally({isToLocal}) {  
 // 同步到本地时刷新下数据  
 if (isToLocal) {  
 this.loadData()  
 }  
 },  
 },  
 created() {  
 this.currFlowId = this.$route.params.id  
 this.currFlowName = this.$route.params.name  
 // this.loadTree()  
 },  
  
 }  
</script>  
<style scoped>  
 @import '~@assets/less/common.less';  
</style>  
<style scoped>  
 .ant-card-body .table-operator {  
 margin: 15px;  
 }  
  
 .anty-form-btn {  
 width: 100%;  
 text-align: center;  
 }  
  
 .anty-form-btn button {  
 margin: 0 5px;  
 }  
  
 .anty-node-layout .ant-layout-header {  
 padding-right: 0  
 }  
  
 .header {  
 padding: 0 8px;  
 }  
  
 .header button {  
 margin: 0 3px  
 }  
  
 .ant-modal-cust-warp {  
 height: 100%  
 }  
  
 .ant-modal-cust-warp .ant-modal-body {  
 height: calc(100% - 110px) !important;  
 overflow-y: auto  
 }  
  
 .ant-modal-cust-warp .ant-modal-content {  
 height: 90% !important;  
 overflow-y: hidden  
 }  
  
 #app .desktop {  
 height: auto !important;  
 }  
  
 /\*\* Button按钮间距 \*/  
 .drawer-bootom-button {  
 /\*position: absolute;\*/  
 bottom: 0;  
 width: 100%;  
 border-top: 1px solid #e8e8e8;  
 padding: 10px 16px;  
 text-align: left;  
 left: 0;  
 background: #fff;  
 border-radius: 0 0 2px 2px;  
 }  
</style>