<template>

    <div class="login">

        <div class="main">

            <div class="left">

                <h1>知识产权从业人员信息化管理系统</h1>

            </div>

            <div class="right">

                <el-form ref="ruleFormRef" :model="ruleForm" :rules="rules" label-width="80px" class="demo-ruleForm">

                    <el-form-item label="用户名" prop="username" style="margin-top: 70px;">

                        <el-input v-model="ruleForm.username" />

                    </el-form-item>

                    <el-form-item label="密码" prop="password" style="margin-top: 30px;">

                        <el-input v-model="ruleForm.password" type="password" />

                    </el-form-item>

                    <el-button @click="handleLogin" type="primary"

                        style="margin-top: 30px;width: 380px;margin-left: 80px;">

                        登录</el-button>

                </el-form>

            </div>

        </div>

    </div>

    <Particles id="tsparticles" class="login-partic" :options="options" />

</template>

<script lang="ts">

export default {

    name: "Login"

}

</script>

<script lang="ts" setup>

import { ref, reactive } from 'vue'

import { useRouter } from 'vue-router'

import { ElMessage, FormInstance, FormRules } from 'element-plus'

import axios from '../http'

const options = reactive({})

const router = useRouter()

const ruleForm = reactive({

    username: '',

    password: ''

})

const rules = reactive<FormRules>({

    username: [

        {

            trigger: 'blur',

            required: true,

            validator: (rule: any, value: any, callback: Function) => {

                var passwordreg = /^\w+$/

                if (!passwordreg.test(value)) {

                    callback(

                        new Error(

                            '用户名必须英文字母、数字、下划线组成!'

                        )

                    )

                } else {

                    callback()

                }

            }

        },

        { min: 5, max: 8, message: '请输入5-8位用户名', trigger: 'blur' },

    ],

    password: [

        {

            trigger: 'blur',

            required: true,

            validator: (rule: any, value: any, callback: Function) => {

                var passwordreg = /^\w+$/

                if (!passwordreg.test(value)) {

                    callback(

                        new Error(

                            '用户名必须英文字母、数字、下划线组成!'

                        )

                    )

                } else {

                    callback()

                }

            }

        },

        { min: 6, max: 10, message: '请输入6-10位用户密码', trigger: 'blur' },

    ],

})

const ruleFormRef = ref(null)

const handleLogin = () => {

    (ruleFormRef.value as any).validate((valid: any, fields: any) => {

        if (valid) {

            axios.post('/login.json', { ...ruleForm }).then((res: any) => {

                ElMessage({

                    type: res.data.status === 1 ? 'success' : 'error',

                    message: res.data.message

                })

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

                    router.push({ name: 'home' })

                    sessionStorage.setItem('userinfo', JSON.stringify(ruleForm))

                }

            })

        }

    })

}

</script>

<style lang="less" scoped>

.login {

    height: 100%;

    background-image: url(@/assets/img/bg.png);

    background-repeat: no-repeat;

    background-size: 100% 100%;

    overflow: hidden;

    .main {

        margin-top: 150px;

        display: flex;

        justify-content: space-around;

        align-items: center;

        .right {

            width: 500px;

            height: 350px;

            background: #fff;

            border-radius: 10px;

            padding-right: 40px;

            box-shadow: 2px 2px 10px #06C;

        }

    }

    .footer {

        margin: 0 auto;

        margin-top: 100px;

        text-align: center;

    }

}

</style>

<template>

    <div class="charts">

        <div class="container " ref="container1"></div>

        <div class="container " ref="container2"></div>

    </div>

</template>

<script lang="ts">

export default {

    name: ""

}

</script>

<script lang="ts" setup>

import { ref, reactive, onMounted } from 'vue'

import \* as echarts from 'echarts';

import axios from '../../http';

const container1 = ref<HTMLElement | null>(null)

const container2 = ref<HTMLElement | null>(null)

const initChart = () => {

    axios.get('/charts.json', { params: {} }).then(res => {

        const myChart1 = echarts.init(container1.value as HTMLElement)

        myChart1.setOption(res.data.chart1);

        const myChart2 = echarts.init(container2.value as HTMLElement)

        myChart2.setOption(res.data.chart2)

    })

}

onMounted(() => {

    initChart()

})

</script>

<style lang="less" scoped>

.charts {

    width: 100%;

    display: flex;

    align-items: center;

    justify-content: space-around;

}

.container {

    width: 400px;

    height: 400px;

}

</style>

<template>

    <div class="info">

        <el-breadcrumb :separator-icon="ArrowRight">

            <el-breadcrumb-item :to="{ path: '/' }">主页</el-breadcrumb-item>

            <el-breadcrumb-item>基本信息</el-breadcrumb-item>

        </el-breadcrumb>

        <div class="form-wrap">

            <div class="btn-group">

                <el-button @click="add" type="success" style="width: 100px;">添加信息</el-button>

                <el-button style="width: 100px;">导入</el-button>

                <el-button style="width: 100px;">导出</el-button>

            </div>

            <el-table :data="tableData" style="width: 100%" :row-class-name="tableRowClassName">

                <el-table-column prop="id" label="编号" width="55px" />

                <el-table-column prop="name" label="姓名" width="80px" />

                <el-table-column prop="gender" label="性别" width="80px">

                    <template #default="{ row, column, $index }">

                        <span>

                            {{row.gender===1?'男':'女'}}

                        </span>

                    </template>

                </el-table-column>

                <el-table-column prop="educational" label="学历" width="120px">

                    <template #default="{ row, column, $index }">

                        <span>

                            {{row.educational===0?'初中及以下':row.educational===1?'高中':row.educational===2?'统招专科':row.educational===3?'统招本科':row.educational===4?'硕士及以上':"其他"}}

                        </span>

                    </template>

                </el-table-column>

                <el-table-column prop="unit" label="单位名称" />

                <el-table-column prop="phone" label="联系电话" />

                <el-table-column label="操作" width="120px">

                    <template #default="{ row, column, $index }">

                        <el-button link type="primary" size="small" @click="edit(row)">编辑</el-button>

                        <el-button link type="primary" size="small" @click="del(row)">删除</el-button>

                    </template>

                </el-table-column>

            </el-table>

            <el-pagination v-model:currentPage="currentPage" v-model:page-size="pageSize" :page-sizes="[7, 14, 21, 28]"

                background layout="total, sizes, prev, pager, next, jumper" :total="totalPage"

                @size-change="handleSizeChange" @current-change="handleCurrentChange" />

            <el-dialog v-model="dialogshow" width="770px">

                <el-form :model="form" ref="formRef" label-width="100px" :rules="rules">

                    <div class="form">

                        <div class="company">

                            <div class="title">企业信息</div>

                            <el-form-item label="单位名称" prop="unit">

                                <el-input v-model="form.unit" placeholder="请填写单位名称" />

                            </el-form-item>

                            <el-form-item label="单位地址" prop="address">

                                <el-input v-model="form.address" placeholder="请填写单位地址" />

                            </el-form-item>

                            <el-form-item label="联系人" prop="linkman">

                                <el-input v-model="form.linkman" placeholder="请填写联系人" />

                            </el-form-item>

                            <el-form-item label="联系电话" prop="phone">

                                <el-input v-model="form.phone" placeholder="请填写联系电话" />

                            </el-form-item>

                        </div>

                        <div class="personal">

                            <div class="title">个人信息</div>

                            <el-form-item label="姓名" prop="name">

                                <el-input v-model="form.name" placeholder="请填写人员姓名" />

                            </el-form-item>

                            <el-form-item label="性别" prop="gender">

                                <el-radio-group v-model="form.gender" placeholder="请选择人员性别">

                                    <el-radio :label="1">男</el-radio>

                                    <el-radio :label="0">女</el-radio>

                                </el-radio-group>

                            </el-form-item>

                            <el-form-item label="学历" prop="educational" style="width:100%">

                                <el-select v-model="form.educational" placeholder="请选择人员学历">

                                    <el-option v-for="(item,index) in educational" :label="item.message"

                                        :value="item.value" />

                                </el-select>

                            </el-form-item>

                            <el-form-item label="专业" prop="major">

                                <el-input v-model="form.major" placeholder="请填写人员专业" />

                            </el-form-item>

                            <el-form-item label="个人简介" prop="remark">

                                <el-input v-model="form.remark" type="textarea" />

                            </el-form-item>

                        </div>

                    </div>

                </el-form>

                <template #footer>

                    <span class="dialog-footer">

                        <el-button type="primary" @click="onSubmit" style="width:100px">确认</el-button>

                        <el-button @click="onCancel" style="width:100px">取消</el-button>

                    </span>

                </template>

            </el-dialog>

        </div>

    </div>

</template>

<script lang="ts">

export default {

    name: ""

}

</script>

<script setup lang="ts">

import { ref, reactive, nextTick } from 'vue'

import { ArrowRight } from '@element-plus/icons-vue'

import { ElMessage, FormRules, ElMessageBox } from 'element-plus'

import axios from '../../http'

const formRef = ref(null)

const tableData = ref([])

const currentPage = ref(1)

const totalPage = ref(1)

const pageSize = ref(7)

const handleSizeChange = (evt: any) => {

    pageSize.value = evt

    getInfo()

}

const handleCurrentChange = (evt: any) => {

    currentPage.value = evt

    getInfo()

}

const getInfo = () => {

    axios.get('/getinfo.json', { params: { page: currentPage.value, pageSize: pageSize.value } }).then(res => {

        totalPage.value = res.data.total

        tableData.value = res.data.result

    })

}

getInfo()

const dialogshow = ref(false)

const isEdit = ref(false)

const add = () => {

    dialogshow.value = true;

    form.value = JSON.parse(JSON.stringify(defaultForm))

    isEdit.value = false

}

const edit = (row: any) => {

    dialogshow.value = true;

    form.value = JSON.parse(JSON.stringify(row));

    isEdit.value = true

    // setTimeout(() => {

    //     (formRef.value as any).resetFields()

    // }, 100)

}

const del = (row: any) => {

    ElMessageBox.confirm(`确定删除【${row.name}】吗？`, {

        confirmButtonText: '确认',

        cancelButtonText: '取消',

    }).then(() => {

        axios.get('/deleteinfo.json', { params: { id: row.id } }).then((res: any) => {

            getInfo()

            ElMessage({

                type: 'success',

                message: res.data.message,

            })

        })

    })

}

const defaultForm = {

    name: '',

    educational: 3,

    major: '',

    unit: '',

    remark: '',

    address: '',

    linkman: '',

    phone: '',

    gender: 1

}

const form = ref({

    name: '',

    educational: '',

    major: '',

    unit: '',

    remark: '',

    address: '',

    linkman: '',

    phone: '',

    gender: ""

})

const onSubmit = () => {

    (formRef.value as any).validate((valid: any, fields: any) => {

        if (valid) {

            let url

            if (isEdit.value) {

                url = '/editinfo.json'

            } else {

                url = '/addinfo.json'

            }

            axios.post(url, {

                ...JSON.parse(JSON.stringify(form.value))

            }).then((res: any) => {

                getInfo()

                ElMessage({

                    type: 'success',

                    message: res.data.message

                })

                form.value = JSON.parse(JSON.stringify(defaultForm))

                setTimeout(() => {

                    (formRef.value as any).resetFields()

                    dialogshow.value = false

                }, 100)

            })

        }

    })

}

const onCancel = () => {

    dialogshow.value = false

    setTimeout(() => {

        form.value = JSON.parse(JSON.stringify(defaultForm))

    }, 100)

}

const educational = reactive([

    { value: 0, message: '初中及以下' },

    { value: 1, message: '高中' },

    { value: 2, message: '统招专科' },

    { value: 3, message: '统招本科' },

    { value: 4, message: '硕士及以上' },

    { value: 5, message: '其他' },

])

const rules = reactive<FormRules>({

    name: [{ required: true, message: '请填写人员姓名', trigger: 'blur' }],

    educational: [{ required: true, message: '请填写人员姓名', trigger: 'change' }],

    major: [{ required: true, message: '请填写人员专业', trigger: 'blur' }],

    unit: [{ required: true, message: '请填写单位名称', trigger: 'blur' }],

    address: [{ required: true, message: '请填写单位地址', trigger: 'blur' }],

    linkman: [{ required: true, message: '请填写联系人', trigger: 'blur' }],

    phone: [{ required: true, message: '请填写联系电话', trigger: 'blur' }],

    gender: [{ required: true, message: '请选择人员性别', trigger: 'change' }],

})

const tableRowClassName = ({

    row,

    rowIndex,

}: {

    row: any

    rowIndex: any

}) => {

    if ((rowIndex + 1) % 2 === 0) {

        // return 'warning-row'

        return 'success-row'

    }

    // else if (rowIndex % 3 === 0) {

    //     return 'success-row'

    // }

    return ''

}

</script>

<style lang="less" scoped>

.info {

    width: 100%;

    height: 100%;

    // overflow-y: auto;

    display: flex;

    flex-direction: column;

    .form-wrap {

        display: flex;

        flex-direction: column;

        justify-content: space-around;

        padding: 0 50px;

        .btn-group {

            margin-bottom: 20px;

        }

        .el-table {

            height: 360px;

            overflow-y: auto;

        }

        .el-pagination {

            align-self: flex-end;

            padding: 10px 0;

        }

        .el-form {

            padding: 20px;

            width: 750px;

            .form {

                display: flex;

                .title {

                    text-align: center;

                    font-weight: bold;

                }

                >div {

                    flex: 1;

                }

            }

            .el-form-item {

                margin-top: 30px;

                .el-select {

                    width: 100%;

                }

            }

        }

    }

}

</style>

<template>

    <div class="record">

        <el-breadcrumb :separator-icon="ArrowRight">

            <el-breadcrumb-item :to="{ path: '/' }">主页</el-breadcrumb-item>

            <el-breadcrumb-item>工作履历</el-breadcrumb-item>

        </el-breadcrumb>

        <div class="form-wrap">

            <div class="btn-group">

                <el-button @click="add" type="success" style="width: 100px;">添加经历</el-button>

                <el-button style="width: 100px;">导入</el-button>

                <el-button style="width: 100px;">导出</el-button>

            </div>

            <el-table :data="tableData" style="width: 100%" :row-class-name="tableRowClassName">

                <el-table-column prop="id" label="编号" width="55px" />

                <el-table-column prop="userid" label="对应人员" width="80px">

                    <template #default="{ row, column, $index }">

                        <span v-if="userData.find((item:any)=>item.id===row.userid)">

                            {{(userData.find((item:any)=>item.id===row.userid) as any).name}}

                        </span>

                    </template>

                </el-table-column>

                <el-table-column prop="time" label="工作时间">

                    <template #default="{ row, column, $index }">

                        <span>

                            {{row.time[0] +'至'+row.time[1]}}

                        </span>

                    </template>

                </el-table-column>

                <el-table-column prop="company" label="公司名称" />

                <el-table-column prop="position" label="职位名称" />

                <el-table-column label="操作" width="120px">

                    <template #default="{ row, column, $index }">

                        <el-button link type="primary" size="small" @click="edit(row)">编辑</el-button>

                        <el-button link type="primary" size="small" @click="del(row)">删除</el-button>

                    </template>

                </el-table-column>

            </el-table>

            <el-pagination v-model:currentPage="currentPage" v-model:page-size="pageSize" :page-sizes="[7, 14, 21, 28]"

                background layout="total, sizes, prev, pager, next, jumper" :total="totalPage"

                @size-change="handleSizeChange" @current-change="handleCurrentChange" />

            <el-dialog v-model="dialogshow" width="580px">

                <el-form ref="formRef" :model="form" label-width="100px">

                    <el-form-item :label="'对应人员'" prop="userid" :rules="{

                      required: true,

                      message: '请填写对应人员',

                      trigger: 'blur',

                    }">

                        <el-select v-model="form.userid" placeholder="请选择对应人员" :disabled="isEdit">

                            <el-option v-for="(item,index) in userData" :label="item.name" :value="item.id" />

                        </el-select>

                    </el-form-item>

                    <el-form-item :label="'工作时间'" prop="time" :rules="{

                      required: true,

                      message: '请填写工作时间',

                      trigger: 'blur',

                    }">

                        <el-date-picker v-model="form.time" format="YYYY/MM/DD" value-format="YYYY-MM-DD"

                            type="daterange" start-placeholder="开始日期" end-placeholder="结束日期"

                            :default-value="[new Date(2020, 9, 1), new Date(2022, 10, 1)]" />

                    </el-form-item>

                    <!-- <el-form-item :label="'公司名称'" :prop="'record.'+index+'.company'" :rules="{ -->

                    <el-form-item :label="'公司名称'" prop="company" :rules="{

                      required: true,

                      message: '请填写公司名称',

                      trigger: 'blur',

                    }">

                        <el-input v-model="form.company" placeholder="请填写公司名称" />

                    </el-form-item>

                    <el-form-item :label="'职位名称'" prop="position" :rules="{

                      required: true,

                      message: '请填写职位名称',

                      trigger: 'blur',

                    }">

                        <el-input v-model="form.position" placeholder="请填写职位名称" />

                    </el-form-item>

                </el-form>

                <template #footer>

                    <span class="dialog-footer">

                        <el-button type="primary" @click="onSubmit" style="width:100px">确认</el-button>

                        <el-button @click="onCancel" style="width:100px">取消</el-button>

                    </span>

                </template>

            </el-dialog>

        </div>

    </div>

</template>

<script lang="ts">

export default {

    name: ""

}

</script>

<script setup lang="ts">

import { ref, reactive } from 'vue'

import { ArrowRight, Delete } from '@element-plus/icons-vue'

import { ElMessage, ElMessageBox } from 'element-plus'

import axios from '../../http'

interface IPubilc {

    [k: string]: any

}

const userData = ref<IPubilc[]>([])

axios.get('/getinfo.json', { params: { page: 1, pageSize: 10000 } }).then(res => {

    userData.value = res.data.result

})

const currentPage = ref(1)

const pageSize = ref(7)

const totalPage = ref(1)

const handleSizeChange = (evt: any) => {

    pageSize.value = evt

    getData()

}

const handleCurrentChange = (evt: any) => {

    currentPage.value = evt

    getData()

}

const tableData = ref([])

const getData = () => {

    axios.get('/getrecord.json', { params: { page: currentPage.value, pageSize: pageSize.value } }).then(res => {

        tableData.value = res.data.result

        totalPage.value = res.data.total

    })

}

getData()

const dialogshow = ref(false)

const edit = (row: any) => {

    dialogshow.value = true;

    form.value = JSON.parse(JSON.stringify(row))

    isEdit.value = true

}

const del = (row: any) => {

    ElMessageBox.confirm(`确定删除【当前经历】吗？`, {

        confirmButtonText: '确认',

        cancelButtonText: '取消',

    }).then(() => {

        axios.get('/deleterecord.json', { params: { id: row.id, userid: row.userid } }).then((res: any) => {

            getData()

            ElMessage({

                type: 'success',

                message: res.data.message,

            })

        })

    })

}

const add = () => {

    dialogshow.value = true;

    form.value = JSON.parse(JSON.stringify(defaultForm))

    isEdit.value = false

}

const defaultForm = {

    time: '',

    company: '',

    position: '',

    userid: 2

}

const form = ref({

    time: '',

    company: '',

    position: '',

    userid: 2

})

const isEdit = ref(false)

const onSubmit = () => {

    (formRef.value as any).validate((valid: any, fields: any) => {

        if (valid) {

            let url

            if (isEdit.value) {

                url = '/editrecord.json'

            } else {

                url = '/addrecord.json'

            }

            axios.post(url, {

                ...JSON.parse(JSON.stringify(form.value))

            }).then((res: any) => {

                getData()

                ElMessage({

                    type: 'success',

                    message: res.data.message

                })

                form.value = JSON.parse(JSON.stringify(defaultForm))

                dialogshow.value = false

                setTimeout(() => {

                    (formRef.value as any).resetFields()

                }, 100)

            })

        }

    })

}

const formRef = ref(null)

const onCancel = () => {

    dialogshow.value = false

    setTimeout(() => {

        form.value = JSON.parse(JSON.stringify(defaultForm))

    }, 100)

}

const tableRowClassName = ({

    row,

    rowIndex,

}: {

    row: any

    rowIndex: any

}) => {

    if ((rowIndex + 1) % 2 === 0) {        // return 'warning-row'

        return 'success-row'

    }

    // else if (rowIndex % 3 === 0) {

    //     return 'success-row'

    // }

    return ''

}

</script>

<style lang="less" scoped>

.record {

    height: 100%;

    width: 100%;

    display: flex;

    flex-direction: column;

    overflow-y: auto;

    .form-wrap {

        width: 100%;

        display: flex;

        flex-direction: column;

        justify-content: space-around;

        padding: 0 50px;

        .btn-group {

            margin-bottom: 20px;

        }

        .el-table {

            height: 360px;

            overflow-y: auto;

        }

        .el-pagination {

            align-self: flex-end;

            padding: 10px 0;

        }

        .el-form {

            width: 550px;

            .el-form-item {

                margin-bottom: 30px;

            }

            .record-item {

                box-shadow: 2px 2px 10px #06C;

                border: 1px solid #ccc;

                padding-right: 20px;

                border-radius: 10px;

                margin-bottom: 20px;

                display: flex;

                .title {

                    font-weight: bold;

                    display: flex;

                    align-items: center;

                    justify-content: space-between;

                    padding-left: 120px;

                    position: relative;

                    padding-top: 10px;

                    .mt-2 {

                        position: absolute;

                        right: -20px;

                        top: 0;

                        border-radius: 10px;

                    }

                }

                .box {

                    width: 100%;

                }

            }

        }

    }

}

</style>

<template>

    <div class="train">

        <el-breadcrumb :separator-icon="ArrowRight">

            <el-breadcrumb-item :to="{ path: '/' }">主页</el-breadcrumb-item>

            <el-breadcrumb-item>继续教育</el-breadcrumb-item>

        </el-breadcrumb>

        <div class="form-wrap">

            <div class="btn-group">

                <el-button @click="add" type="success" style="width: 100px;">添加经历</el-button>

                <el-button style="width: 100px;">导入</el-button>

                <el-button style="width: 100px;">导出</el-button>

            </div>

            <el-table :data="tableData" style="width: 100%" :row-class-name="tableRowClassName">

                <el-table-column prop="id" label="编号" width="55px" />

                <el-table-column prop="userid" label="对应人员" width="80px">

                    <template #default="{ row, column, $index }">

                        <span v-if="userData.find((item: any) => item.id === row.userid)">

                            {{ (userData.find((item: any) => item.id === row.userid) as any).name }}

                        </span>

                    </template>

                </el-table-column>

                <el-table-column prop="time" label="培训时间">

                    <template #default="{ row, column, $index }">

                        <span>

                            {{ row.time[0] + '至' + row.time[1] }}

                        </span>

                    </template>

                </el-table-column>

                <el-table-column prop="company" label="机构名称" />

                <el-table-column prop="position" label="培训内容" />

                <el-table-column label="操作" width="120px">

                    <template #default="{ row, column, $index }">

                        <el-button link type="primary" size="small" @click="edit(row)">编辑</el-button>

                        <el-button link type="primary" size="small" @click="del(row)">删除</el-button>

                    </template>

                </el-table-column>

            </el-table>

            <el-pagination v-model:currentPage="currentPage" v-model:page-size="pageSize" :page-sizes="[7, 14, 21, 28]"

                background layout="total, sizes, prev, pager, next, jumper" :total="totalPage"

                @size-change="handleSizeChange" @current-change="handleCurrentChange" />

            <el-dialog v-model="dialogshow" width="580px">

                <el-form ref="formRef" :model="form" label-width="100px">

                    <el-form-item :label="'对应人员'" prop="userid" :rules="{

                        required: true,

                        message: '请填写对应人员',

                        trigger: 'blur',

                    }">

                        <el-select v-model="form.userid" :disabled="isEdit" placeholder="请选择对应人员">

                            <el-option v-for="(item, index) in userData" :label="item.name" :value="item.id" />

                        </el-select>

                    </el-form-item>

                    <el-form-item :label="'培训时间'" prop="time" :rules="{

                        required: true,

                        message: '请填写培训时间',

                        trigger: 'blur',

                    }">

                        <el-date-picker v-model="form.time" format="YYYY/MM/DD" value-format="YYYY-MM-DD"

                            type="daterange" start-placeholder="开始日期" end-placeholder="结束日期"

                            :default-value="[new Date(2020, 9, 1), new Date(2022, 10, 1)]" />

                    </el-form-item>

                    <el-form-item :label="'机构名称'" prop="company" :rules="{

                        required: true,

                        message: '请填写机构名称',

                        trigger: 'blur',

                    }">

                        <el-input v-model="form.company" placeholder="请填写机构名称" />

                    </el-form-item>

                    <el-form-item :label="'培训内容'" prop="position" :rules="{

                        required: true,

                        message: '请填写培训内容',

                        trigger: 'blur',

                    }">

                        <el-input v-model="form.position" placeholder="请填写培训内容" />

                    </el-form-item>

                </el-form>

                <template #footer>

                    <span class="dialog-footer">

                        <el-button type="primary" @click="onSubmit" style="width:100px">确认</el-button>

                        <el-button @click="onCancel" style="width:100px">取消</el-button>

                    </span>

                </template>

            </el-dialog>

        </div>

    </div>

</template>

<script lang="ts">

export default {

    name: ""

}

</script>

<script setup lang="ts">

import { ref, reactive } from 'vue'

import { ArrowRight, Delete } from '@element-plus/icons-vue'

import { ElMessage, ElMessageBox } from 'element-plus'

import axios from '../../http'

interface IPubilc {

    [k: string]: any

}

const userData = ref<IPubilc[]>([])

axios.get('/getinfo.json', { params: { page: 1, pageSize: 10000 } }).then(res => {

    userData.value = res.data.result

})

const currentPage = ref(1)

const totalPage = ref(1)

const pageSize = ref(7)

const handleSizeChange = (evt: any) => {

    pageSize.value = evt

    getData()

}

const handleCurrentChange = (evt: any) => {

    currentPage.value = evt

    getData()

}

const tableData = ref([])

const getData = () => {

    axios.get('/gettrain.json', { params: { page: currentPage.value, pageSize: pageSize.value } }).then(res => {

        tableData.value = res.data.result

        totalPage.value = res.data.total

    })

}

getData()

const dialogshow = ref(false)

const isEdit = ref(false)

const edit = (row: any) => {

    dialogshow.value = true;

    form.value = JSON.parse(JSON.stringify(row))

    isEdit.value = true

}

const del = (row: any) => {

    ElMessageBox.confirm(`确定删除【当前经历】吗？`, {

        confirmButtonText: '确认',

        cancelButtonText: '取消',

    }).then(() => {

        axios.get('/deletetrain.json', { params: { id: row.id, userid: row.userid } }).then((res: any) => {

            getData()

            ElMessage({

                type: 'success',

                message: res.data.message,

            })

        })

    })

}

const add = () => {

    dialogshow.value = true;

    form.value = JSON.parse(JSON.stringify(defaultForm))

    isEdit.value = false

}

const defaultForm = {

    userid: 2,

    time: '',

    company: '',

    position: ''

}

const form = ref({

    userid: 2,

    time: '',

    company: '',

    position: ''

})

const formRef = ref(null)

const onCancel = () => {

    dialogshow.value = false

    setTimeout(() => {

        form.value = JSON.parse(JSON.stringify(defaultForm))

    }, 100)

}

const onSubmit = () => {

    (formRef.value as any).validate((valid: any, fields: any) => {

        if (valid) {

            let url

            if (isEdit.value) {

                url = '/edittrain.json'

            } else {

                url = '/addtrain.json'

            }

            axios.post(url, {

                ...JSON.parse(JSON.stringify(form.value))

            }).then((res: any) => {

                getData()

                ElMessage({

                    type: 'success',

                    message: res.data.message

                })

                form.value = JSON.parse(JSON.stringify(defaultForm))

                dialogshow.value = false

                setTimeout(() => {

                    (formRef.value as any).resetFields()

                }, 100)

            })

        }

    })

}

const tableRowClassName = ({

    row,

    rowIndex,

}: {

    row: any

    rowIndex: any

}) => {

    if ((rowIndex + 1) % 2 === 0) {

        // return 'warning-row'

        return 'success-row'

    }

    // else if (rowIndex % 3 === 0) {

    //     return 'success-row'

    // }

    return ''

}

</script>

<style lang="less" scoped>

.train {

    height: 100%;

    width: 100%;

    display: flex;

    flex-direction: column;

    overflow-y: auto;

    .form-wrap {

        width: 100%;

        display: flex;

        flex-direction: column;

        justify-content: space-around;

        padding: 0 50px;

        .btn-group {

            margin-bottom: 20px;

        }

        .el-table {

            height: 360px;

            overflow-y: auto;

        }

        .el-pagination {

            align-self: flex-end;

            padding: 10px 0;

        }

        .el-form {

            width: 550px;

            .el-form-item {

                margin-bottom: 30px;

            }

            .record-item {

                box-shadow: 2px 2px 10px #06C;

                border: 1px solid #ccc;

                padding-right: 20px;

                border-radius: 10px;

                margin-bottom: 20px;

                display: flex;

                .title {

                    font-weight: bold;

                    display: flex;

                    align-items: center;

                    justify-content: space-between;

                    padding-left: 120px;

                    position: relative;

                    padding-top: 10px;

                    .mt-2 {

                        position: absolute;

                        right: -20px;

                        top: 0;

                        border-radius: 10px;

                    }

                }

                .box {

                    width: 100%;

                }

            }

        }

    }

}

</style>

<template>

    <div class="query">

        <el-breadcrumb :separator-icon="ArrowRight">

            <el-breadcrumb-item :to="{ path: '/' }">主页</el-breadcrumb-item>

            <el-breadcrumb-item>信息检索</el-breadcrumb-item>

        </el-breadcrumb>

        <div class="search">

            <el-form :model="form" ref="formRef" label-width="80px">

                <el-form-item label="关键词">

                    <el-input v-model="form.name" style="width: 200px;" />

                </el-form-item>

                <el-form-item label="工作年限">

                    <el-input-number v-model="form.years" :min="1" :max="30" />

                </el-form-item>

                <el-form-item label="业务类型" prop="type">

                    <el-select v-model="form.type" placeholder="请选择所属业务类型" style="width:100%">

                        <el-option v-for="(item,index) in type" :label="item.message" :value="item.value" />

                    </el-select>

                </el-form-item>

                <div class="btn">

                    <el-button type="primary" @click="onSubmit">查询</el-button>

                    <el-button @click="onReset">重置</el-button>

                </div>

            </el-form>

        </div>

        <div class="table">

            <el-table :data="tableData" style="width: 100%" :row-class-name="tableRowClassName">

                <el-table-column prop="id" label="编号" width="55px" />

                <el-table-column prop="name" label="姓名" width="80px" />

                <el-table-column prop="gender" label="性别" width="55px">

                    <template #default="{ row, column, $index }">

                        <span>

                            {{row.gender===1?'男':'女'}}

                        </span>

                    </template>

                </el-table-column>

                <el-table-column prop="educational" label="学历" width="120px">

                    <template #default="{ row, column, $index }">

                        <span>

                            {{row.educational===0?'初中及以下':row.educational===1?'高中':row.educational===2?'统招专科':row.educational===3?'统招本科':row.educational===4?'硕士及以上':"其他"}}

                        </span>

                    </template>

                </el-table-column>

                <el-table-column prop="unit" label="单位名称" />

                <el-table-column prop="phone" label="联系电话" />

                <el-table-column label="操作" width="120px">

                    <template #default="{ row, column, $index }">

                        <el-button link type="primary" size="small" @click="check(row)">查看详情</el-button>

                    </template>

                </el-table-column>

            </el-table>

            <el-pagination v-model:currentPage="currentPage" v-model:page-size="pageSize" :page-sizes="[7, 14, 21, 28]"

                background layout="total, sizes, prev, pager, next, jumper" :total="totalPage"

                @size-change="handleSizeChange" @current-change="handleCurrentChange" />

            <el-dialog v-model="dialogshow" width="580px" destroy-on-close>

                <div class="dialog-wrap" ref="dialog">

                    <div class="self">

                        <div class="c-title">个人信息</div>

                        <div class="con">

                            <div class="con-item">

                                <span>姓名：</span>

                                <div>{{info.name}}</div>

                            </div>

                            <div class="con-item">

                                <span>性别：</span>

                                <div>{{info.gender===1?"男":'女'}}</div>

                            </div>

                            <div class="con-item">

                                <span>学历：</span>

                                <div>

                                    {{info.educational===0?'初中及以下':info.educational===1?'高中':info.educational===2?'统招专科':info.educational===3?'统招本科':info.educational===4?'硕士及以上':"其他"}}

                                </div>

                            </div>

                            <div class="con-item">

                                <span>专业：</span>

                                <div>{{info.major}}</div>

                            </div>

                            <div class="con-item" v-if="info.remark">

                                <span>个人简介：</span>

                                <div>{{info.remark}}</div>

                            </div>

                        </div>

                    </div>

                    <br>

                    <hr>

                    <br>

                    <div class="compan">

                        <div class="c-title">企业信息</div>

                        <div class="con">

                            <div class="con-item">

                                <span>单位名称：</span>

                                <div>{{info.unit}}</div>

                            </div>

                            <div class="con-item">

                                <span>单位地址：</span>

                                <div>{{info.address}}</div>

                            </div>

                            <div class="con-item">

                                <span>联系人：</span>

                                <div>{{info.linkman}}</div>

                            </div>

                            <div class="con-item">

                                <span>联系电话：</span>

                                <div>{{info.phone}}</div>

                            </div>

                        </div>

                    </div>

                    <br>

                    <hr>

                    <br>

                    <div class="exp" v-if="info.record.length!==0">

                        <div class="c-title">工作履历</div>

                        <div class="con" v-for="(item,index) in info.record" :key="index">

                            <div class="con-item">

                                <span>工作时间：</span>

                                <div>{{item.time[0]+'至'+item.time[1]}}</div>

                            </div>

                            <div class="con-item">

                                <span>公司名称：</span>

                                <div>{{item.company}}</div>

                            </div>

                            <div class="con-item">

                                <span>职位名称：</span>

                                <div>{{item.position}}</div>

                            </div>

                            <br>

                        </div>

                    </div>

                    <br v-if="info.record.length!==0">

                    <hr v-if="info.record.length!==0">

                    <br v-if="info.record.length!==0">

                    <div class="train" v-if="info.train.length!==0">

                        <div class="c-title">继续教育</div>

                        <div class="con" v-for="(item,index) in info.train" :key="index">

                            <div class="con-item">

                                <span>培训时间：</span>

                                <div>{{item.time[0]+'至'+item.time[1]}}</div>

                            </div>

                            <div class="con-item">

                                <span>机构名称：</span>

                                <div>{{item.company}}</div>

                            </div>

                            <div class="con-item">

                                <span>培训内容：</span>

                                <div>{{item.position}}</div>

                            </div>

                            <br>

                        </div>

                    </div>

                    <br v-if="info.train.length!==0">

                    <hr v-if="info.train.length!==0">

                    <br v-if="info.train.length!==0">

                    <div class="case" v-if="info.case.length!==0">

                        <div class="c-title">经典案例</div>

                        <div class="con" v-for="(item,index) in info.case" :key="index">

                            <div class="con-item">

                                <span>案例名称：</span>

                                <div>{{item.name}}</div>

                            </div>

                            <div class="con-item">

                                <span>案例年份：</span>

                                <div>{{item.year}}</div>

                            </div>

                            <div class="con-item">

                                <span>所属业务类型：</span>

                                <div>{{info.type===0?'专利':info.type===1?'商标':'版权'}}</div>

                            </div>

                            <div class="con-item">

                                <span>案例基本情况：</span>

                                <div>{{item.intro}}</div>

                            </div>

                            <div class="con-item" v-if="item.error">

                                <span>问题争议点：</span>

                                <div>{{item.error}}</div>

                            </div>

                            <div class="con-item" v-if="item.deal">

                                <span>方案解决：</span>

                                <div>{{item.deal}}</div>

                            </div>

                            <br>

                        </div>

                    </div>

                </div>

            </el-dialog>

        </div>

    </div>

</template>

<script lang="ts">

export default {

    name: ""

}

</script>

<script setup lang="ts">

import { ref, reactive, nextTick } from 'vue'

import { ArrowRight } from '@element-plus/icons-vue'

import axios from '../../http'

interface IPubilc {

    [k: string]: any

}

const currentPage = ref(1)

const pageSize = ref(7)

const totalPage = ref(1)

const tableData = ref([])

const handleSizeChange = (evt: any) => {

    pageSize.value = evt

    getInfo()

}

const handleCurrentChange = (evt: any) => {

    currentPage.value = evt

    getInfo()

}

const getInfo = () => {

    axios.get('/getinfo.json', { params: { page: currentPage.value, pageSize: pageSize.value } }).then(res => {

        totalPage.value = res.data.total

        tableData.value = res.data.result

    })

}

getInfo()

const dialogshow = ref(false)

const info = ref<IPubilc>({})

const dialog = ref(null)

const check = (row: any) => {

    setTimeout(() => {

        info.value = JSON.parse(JSON.stringify(row))

        dialogshow.value = true

    }, 80)

}

const form = ref({

    name: '',

    type: '',

    years: 2

})

const type = ref([

    { value: 0, message: '专利' },

    { value: 1, message: '商标' },

    { value: 2, message: '版权' },

])

const onSubmit = () => { }

const onReset = () => {

    form.value = {

        name: '',

        type: '',

        years: 2

    }

}

const tableRowClassName = ({

    row,

    rowIndex,

}: {

    row: any

    rowIndex: any

}) => {

    if ((rowIndex + 1) % 2 === 0) {

        // return 'warning-row'

        return 'success-row'

    }

    // else if (rowIndex % 3 === 0) {

    //     return 'success-row'

    // }

    return ''

}

</script>

<style lang="less" scoped>

.query {

    width: 100%;

    .search {

        padding: 0 50px;

        .el-form {

            display: flex;

            justify-content: space-between;

        }

    }

    .table {

        display: flex;

        flex-direction: column;

        padding: 0 50px;

        margin-top: 30px;

        .el-table {

            height: 360px;

            overflow-y: auto;

        }

        .el-pagination {

            align-self: flex-end;

            padding: 10px 0;

        }

    }

}

.dialog-wrap {

    padding: 20px;

    padding-left: 30px;

    max-height: 450px;

    overflow-y: auto;

    .c-title {

        font-weight: bold;

        font-size: 17px;

        margin: 10px 0;

    }

    .con {

        .con-item {

            display: flex;

            span {

                font-weight: bold;

                color: gray;

            }

            div {

                width: 400px;

            }

        }

    }

}

</style>