<!DOCTYPE html><html lang="zh-CN"><head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <title>智博图书管理系统</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>智博图书管理系统</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item active"><span>图书信息管理</span></li></a> <a ><li class="rz-nav-item"><span>借阅管理</span></li></a> <a ><li class="rz-nav-item"><span>归还管理</span></li></a> <a ><li class="rz-nav-item"><span>用户管理</span></li></a> <a ><li class="rz-nav-item"><span>统计报表生成</span></li></a> <a ><li class="rz-nav-item"><span>图书查询</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>图书信息管理</h2> <form class="add-book-form"> <div class="form-group"> <label for="bookTitle">书名</label> <input type="text" id="bookTitle" placeholder="请输入书名" required> </div> <div class="form-group"> <label for="author">作者</label> <input type="text" id="author" placeholder="请输入作者" required> </div> <div class="form-group"> <label for="isbn">ISBN</label> <input type="text" id="isbn" placeholder="请输入ISBN" required> </div> <div class="form-group"> <label for="category">类别</label> <input type="text" id="category" placeholder="请输入类别" required> </div> <div class="form-group"> <label for="stock">库存量</label> <input type="number" id="stock" placeholder="请输入库存量" required> </div> <div class="form-group"> <button type="submit">添加书籍</button> </div> </form> <table> <thead> <tr> <th>书名</th> <th>作者</th> <th>ISBN</th> <th>类别</th> <th>库存量</th> <th>操作</th> </tr> </thead> <tbody> <tr> <td>示例书名</td> <td>示例作者</td> <td>1234567890</td> <td>示例类别</td> <td>10</td> <td><span class="btn-delete">删除</span></td> </tr> <!-- 其他书籍信息 --> </tbody> </table> </div> </main> </div></body></html>

<!DOCTYPE html><html lang="zh-CN"><head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <title>智博图书管理系统</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>智博图书管理系统</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>图书信息管理</span></li></a> <a ><li class="rz-nav-item active"><span>借阅管理</span></li></a> <a ><li class="rz-nav-item"><span>归还管理</span></li></a> <a ><li class="rz-nav-item"><span>用户管理</span></li></a> <a ><li class="rz-nav-item"><span>统计报表生成</span></li></a> <a ><li class="rz-nav-item"><span>图书查询</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>借阅管理</h2> <button class="button">新增借阅申请</button> <button class="button">审核借阅申请</button> <table> <thead> <tr> <th>书籍名称</th> <th>借阅用户</th> <th>借阅状态</th> <th>借出日期</th> <th>归还日期</th> <th>操作</th> </tr> </thead> <tbody> <tr> <td>计算机科学导论</td> <td>张三</td> <td>已借出</td> <td>2023-10-01</td> <td>2023-10-15</td> <td> <button class="button">确认归还</button> <button class="button">删除</button> </td> </tr> <tr> <td>数据结构</td> <td>李四</td> <td>待审核</td> <td>-</td> <td>-</td> <td> <button class="button">审核</button> <button class="button">删除</button> </td> </tr> <!-- 更多借阅记录 --> </tbody> </table> </div> </main> </div></body></html>

<!DOCTYPE html><html lang="zh-CN"><head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <title>智博图书管理系统</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>智博图书管理系统</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>图书信息管理</span></li></a> <a ><li class="rz-nav-item"><span>借阅管理</span></li></a> <a ><li class="rz-nav-item active"><span>归还管理</span></li></a> <a ><li class="rz-nav-item"><span>用户管理</span></li></a> <a ><li class="rz-nav-item"><span>统计报表生成</span></li></a> <a ><li class="rz-nav-item"><span>图书查询</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>归还管理</h2> <table> <thead> <tr> <th>书籍名称</th> <th>借阅人</th> <th>借阅日期</th> <th>应还日期</th> <th>操作</th> </tr> </thead> <tbody> <tr> <td>书籍 A</td> <td>张三</td> <td>2023-10-01</td> <td>2023-10-15</td> <td><button class="btn">归还</button></td> </tr> <tr> <td>书籍 B</td> <td>李四</td> <td>2023-10-05</td> <td>2023-10-20</td> <td><button class="btn">归还</button></td> </tr> <!-- 更多书籍记录 --> </tbody> </table> <button class="btn">确认所有归还</button> </div> </main> </div></body></html>

<!DOCTYPE html><html lang="zh-CN"><head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <title>智博图书管理系统</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>智博图书管理系统</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>图书信息管理</span></li></a> <a ><li class="rz-nav-item"><span>借阅管理</span></li></a> <a ><li class="rz-nav-item"><span>归还管理</span></li></a> <a ><li class="rz-nav-item active"><span>用户管理</span></li></a> <a ><li class="rz-nav-item"><span>统计报表生成</span></li></a> <a ><li class="rz-nav-item"><span>图书查询</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>用户管理</h2> <div class="user-actions"> <button>添加用户</button> <button>编辑用户</button> <button>删除用户</button> </div> <table> <thead> <tr> <th>用户名</th> <th>权限</th> <th>借阅记录</th> <th>操作</th> </tr> </thead> <tbody> <tr> <td>张三</td> <td>管理员</td> <td>5</td> <td><button>查看</button></td> </tr> <tr> <td>李四</td> <td>用户</td> <td>3</td> <td><button>查看</button></td> </tr> <tr> <td>王五</td> <td>用户</td> <td>1</td> <td><button>查看</button></td> </tr> </tbody> </table> </div> </main> </div></body></html>

<!DOCTYPE html><html lang="zh-CN"><head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <title>智博图书管理系统</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>智博图书管理系统</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>图书信息管理</span></li></a> <a ><li class="rz-nav-item"><span>借阅管理</span></li></a> <a ><li class="rz-nav-item"><span>归还管理</span></li></a> <a ><li class="rz-nav-item"><span>用户管理</span></li></a> <a ><li class="rz-nav-item active"><span>统计报表生成</span></li></a> <a ><li class="rz-nav-item"><span>图书查询</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>统计报表生成</h2> <form> <div class="form-group"> <label for="timeRange">选择时间范围:</label> <select id="timeRange"> <option value="lastWeek">过去一周</option> <option value="lastMonth">过去一个月</option> <option value="lastYear">过去一年</option> <option value="custom">自定义</option> </select> </div> <div class="form-group"> <label for="bookCategory">选择图书类别:</label> <select id="bookCategory"> <option value="all">所有类别</option> <option value="fiction">小说</option> <option value="nonFiction">非小说</option> <option value="science">科学</option> <option value="history">历史</option> </select> </div> <button type="submit" class="btn">生成报表</button> </form> <div class="report-output"> <h3>报表输出:</h3> <p>请通过上面的选择生成统计报表。</p> <!-- 生成的报表内容将会在这里显示 --> </div> </div> </main> </div></body></html>

<!DOCTYPE html><html lang="zh-CN"><head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <title>智博图书管理系统</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>智博图书管理系统</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>图书信息管理</span></li></a> <a ><li class="rz-nav-item"><span>借阅管理</span></li></a> <a ><li class="rz-nav-item"><span>归还管理</span></li></a> <a ><li class="rz-nav-item"><span>用户管理</span></li></a> <a ><li class="rz-nav-item"><span>统计报表生成</span></li></a> <a ><li class="rz-nav-item active"><span>图书查询</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>图书查询</h2> <div class="search-bar"> <input type="text" placeholder="请输入书名、作者或ISBN进行搜索" style="width: 100%; padding: 10px; border: 1px solid #ddd; border-radius: 4px;"> </div> <ul class="book-list"> <li class="book-item">书名: 《红楼梦》, 作者: 曹雪芹, ISBN: 978-7-5399-3005-8, 库存: 5</li> <li class="book-item">书名: 《西游记》, 作者: 吴承恩, ISBN: 978-7-5399-9005-4, 库存: 2</li> <li class="book-item">书名: 《三国演义》, 作者: 罗贯中, ISBN: 978-7-5399-1005-7, 库存: 0</li> </ul> </div> </main> </div></body></html>

/\*\* \* @description books \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/books")public class BooksController { @Resource private BooksService booksService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(Books books){ return booksService.insert(books); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return booksService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(Books books){ return booksService.update(books); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return booksService.load(id); } /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ @RequestMapping("/pageList") public Map<String, Object> pageList(@RequestParam(required = false, defaultValue = "0") int offset, @RequestParam(required = false, defaultValue = "10") int pagesize) { return booksService.pageList(offset, pagesize); }}/\*\* \* @description books \* @author 大大的盆 \*/public interface BooksService { /\*\* \* 新增 \*/ public Object insert(Books books); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(Books books); /\*\* \* 根据主键 id 查询 \*/ public Books load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description books \* @author 大大的盆 \*/@Servicepublic class BooksServiceImpl implements BooksService { @Resource private BooksMapper booksMapper; @Override public Object insert(Books books) { // valid if (books == null) { return ReturnT.error("必要参数缺失"); } booksMapper.insert(books); return ReturnT.success(); } @Override public Object delete(int id) { int ret = booksMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(Books books) { int ret = booksMapper.update(books); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Books load(int id) { return booksMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<Books> pageList = booksMapper.pageList(offset, pagesize); int totalCount = booksMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description books \* @author 大大的盆 \*/@Mapper@Repositorypublic interface BooksMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(Books books); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(Books books); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ Books load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<Books> pageList(int offset,int pagesize); /\*\* \* 查询 分页查询 count \* @author 大大的盆 \*\*/ int pageListCount(int offset,int pagesize);}<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd"><mapper namespace="HBJZGCXY.com.dao.BooksMapper"> <resultMap id="BaseResultMap" type="HBJZGCXY.com.entity.Books" > <result column="title" property="title" /> <result column="author" property="author" /> <result column="total\_copies" property="totalCopies" /> <result column="available\_copies" property="availableCopies" /> <result column="create\_time" property="createTime" /> <result column="create\_user" property="createUser" /> <result column="update\_time" property="updateTime" /> <result column="update\_user" property="updateUser" /> </resultMap> <sql id="Base\_Column\_List"> title, author, total\_copies, available\_copies, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="HBJZGCXY.com.entity.Books"> INSERT INTO books <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != title and '' != title"> title, </if> <if test="null != author and '' != author"> author, </if> <if test="null != totalCopies and '' != totalCopies"> total\_copies, </if> <if test="null != availableCopies and '' != availableCopies"> available\_copies, </if> <if test="null != createTime and '' != createTime"> create\_time, </if> <if test="null != createUser and '' != createUser"> create\_user, </if> <if test="null != updateTime and '' != updateTime"> update\_time, </if> <if test="null != updateUser and '' != updateUser"> update\_user </if> </trim> <trim prefix="values (" suffix=")" suffixOverrides=","> <if test="null != title and '' != title"> #{title}, </if> <if test="null != author and '' != author"> #{author}, </if> <if test="null != totalCopies and '' != totalCopies"> #{totalCopies}, </if> <if test="null != availableCopies and '' != availableCopies"> #{availableCopies}, </if> <if test="null != createTime and '' != createTime"> #{createTime}, </if> <if test="null != createUser and '' != createUser"> #{createUser}, </if> <if test="null != updateTime and '' != updateTime"> #{updateTime}, </if> <if test="null != updateUser and '' != updateUser"> #{updateUser} </if> </trim> </insert> <delete id="delete" > DELETE FROM books WHERE id = #{id} </delete> <update id="update" parameterType="HBJZGCXY.com.entity.Books"> UPDATE books <set> <if test="null != title and '' != title">title = #{title},</if> <if test="null != author and '' != author">author = #{author},</if> <if test="null != totalCopies and '' != totalCopies">total\_copies = #{totalCopies},</if> <if test="null != availableCopies and '' != availableCopies">available\_copies = #{availableCopies},</if> <if test="null != createTime and '' != createTime">create\_time = #{createTime},</if> <if test="null != createUser and '' != createUser">create\_user = #{createUser},</if> <if test="null != updateTime and '' != updateTime">update\_time = #{updateTime},</if> <if test="null != updateUser and '' != updateUser">update\_user = #{updateUser}</if> </set> WHERE id = #{id} </update> <select id="load" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM books WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM books LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM books </select></mapper>/\*\* \* @description books \* @author 大大的盆 \*/public class Books implements Serializable { private static final long serialVersionUID = 1L; private String title; private String author; private String totalCopies; private String availableCopies; private String createTime; private String createUser; private String updateTime; private String updateUser; public Books() { } public String getTitle() { return title; } public void setTitle(String title) { this.title = title; } public String getAuthor() { return author; } public void setAuthor(String author) { this.author = author; } public String getTotalCopies() { return totalCopies; } public void setTotalCopies(String totalCopies) { this.totalCopies = totalCopies; } public String getAvailableCopies() { return availableCopies; } public void setAvailableCopies(String availableCopies) { this.availableCopies = availableCopies; } public String getCreateTime() { return createTime; } public void setCreateTime(String createTime) { this.createTime = createTime; } public String getCreateUser() { return createUser; } public void setCreateUser(String createUser) { this.createUser = createUser; } public String getUpdateTime() { return updateTime; } public void setUpdateTime(String updateTime) { this.updateTime = updateTime; } public String getUpdateUser() { return updateUser; } public void setUpdateUser(String updateUser) { this.updateUser = updateUser; }}<template> <div class="mod-config"> <el-form :inline="true" :model="dataForm" @keyup.enter.native="getDataList()"> <el-form-item> <el-input v-model="dataForm.key" placeholder="参数名" clearable></el-input> </el-form-item> <el-form-item> <el-button @click="getDataList()">查询</el-button> <el-button v-if="isAuth('generator:books:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:books:delete')" type="danger" @click="deleteHandle()" :disabled="dataListSelections.length <= 0">批量删除</el-button> </el-form-item> </el-form> <el-table :data="dataList" border v-loading="dataListLoading" @selection-change="selectionChangeHandle" style="width: 100%;"> <el-table-column type="selection" header-align="center" align="center" width="50"> </el-table-column> <el-table-column prop="title" header-align="center" align="center" label="title"> </el-table-column> <el-table-column prop="author" header-align="center" align="center" label="author"> </el-table-column> <el-table-column prop="totalCopies" header-align="center" align="center" label="total\_copies"> </el-table-column> <el-table-column prop="availableCopies" header-align="center" align="center" label="available\_copies"> </el-table-column> <el-table-column prop="createTime" header-align="center" align="center" label="create\_time"> </el-table-column> <el-table-column prop="createUser" header-align="center" align="center" label="create\_user"> </el-table-column> <el-table-column prop="updateTime" header-align="center" align="center" label="update\_time"> </el-table-column> <el-table-column prop="updateUser" header-align="center" align="center" label="update\_user"> </el-table-column> <el-table-column fixed="right" header-align="center" align="center" width="150" label="操作"> <template slot-scope="scope"> <el-button type="text" size="small" @click="addOrUpdateHandle(scope.row.booksId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.booksId)">删除</el-button> </template> </el-table-column> </el-table> <el-pagination @size-change="sizeChangeHandle" @current-change="currentChangeHandle" :current-page="pageIndex" :page-sizes="[10, 20, 50, 100]" :page-size="pageSize" :total="totalPage" layout="total, sizes, prev, pager, next, jumper"> </el-pagination> <!-- 弹窗, 新增 / 修改 --> <add-or-update v-if="addOrUpdateVisible" ref="addOrUpdate" @refreshDataList="getDataList"></add-or-update> </div></template><script> import AddOrUpdate from './books-add-or-update' export default { data () { return { dataForm: { key: '' }, dataList: [], pageIndex: 1, pageSize: 10, totalPage: 0, dataListLoading: false, dataListSelections: [], addOrUpdateVisible: false } }, components: { AddOrUpdate }, activated () { this.getDataList() }, methods: { // 获取数据列表 getDataList () { this.dataListLoading = true this.$http({ url: this.$http.adornUrl('/generator/books/list'), method: 'get', params: this.$http.adornParams({ 'page': this.pageIndex, 'limit': this.pageSize, 'key': this.dataForm.key }) }).then(({data}) => { if (data && data.code === 0) { this.dataList = data.page.list this.totalPage = data.page.totalCount } else { this.dataList = [] this.totalPage = 0 } this.dataListLoading = false }) }, // 每页数 sizeChangeHandle (val) { this.pageSize = val this.pageIndex = 1 this.getDataList() }, // 当前页 currentChangeHandle (val) { this.pageIndex = val this.getDataList() }, // 多选 selectionChangeHandle (val) { this.dataListSelections = val }, // 新增 / 修改 addOrUpdateHandle (id) { this.addOrUpdateVisible = true this.$nextTick(() => { this.$refs.addOrUpdate.init(id) }) }, // 删除 deleteHandle (id) { var ids = id ? [id] : this.dataListSelections.map(item => { return item.booksId }) this.$confirm(`确定对[id=${ids.join(',')}]进行[${id ? '删除' : '批量删除'}]操作?`, '提示', { confirmButtonText: '确定', cancelButtonText: '取消', type: 'warning' }).then(() => { this.$http({ url: this.$http.adornUrl('/generator/books/delete'), method: 'post', data: this.$http.adornData(ids, false) }).then(({data}) => { if (data && data.code === 0) { this.$message({ message: '操作成功', type: 'success', duration: 1500, onClose: () => { this.getDataList() } }) } else { this.$message.error(data.msg) } }) }) } } }</script><template> <el-dialog :title="!dataForm.booksId ? '新增' : '修改'" :close-on-click-modal="false" :visible.sync="visible"> <el-form :model="dataForm" :rules="dataRule" ref="dataForm" @keyup.enter.native="dataFormSubmit()" label-width="80px"> <el-form-item label="title" prop="title"> <el-input v-model="dataForm.title" placeholder="title"></el-input> </el-form-item> <el-form-item label="author" prop="author"> <el-input v-model="dataForm.author" placeholder="author"></el-input> </el-form-item> <el-form-item label="total\_copies" prop="totalCopies"> <el-input v-model="dataForm.totalCopies" placeholder="total\_copies"></el-input> </el-form-item> <el-form-item label="available\_copies" prop="availableCopies"> <el-input v-model="dataForm.availableCopies" placeholder="available\_copies"></el-input> </el-form-item> <el-form-item label="create\_time" prop="createTime"> <el-input v-model="dataForm.createTime" placeholder="create\_time"></el-input> </el-form-item> <el-form-item label="create\_user" prop="createUser"> <el-input v-model="dataForm.createUser" placeholder="create\_user"></el-input> </el-form-item> <el-form-item label="update\_time" prop="updateTime"> <el-input v-model="dataForm.updateTime" placeholder="update\_time"></el-input> </el-form-item> <el-form-item label="update\_user" prop="updateUser"> <el-input v-model="dataForm.updateUser" placeholder="update\_user"></el-input> </el-form-item> </el-form> <span slot="footer" class="dialog-footer"> <el-button @click="visible = false">取消</el-button> <el-button type="primary" @click="dataFormSubmit()">确定</el-button> </span> </el-dialog></template><script> export default { data () { return { visible: false, dataForm: { title: '', author: '', totalCopies: '', availableCopies: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { title: [{ required: true, message: 'title不能为空', trigger: 'blur' }], author: [{ required: true, message: 'author不能为空', trigger: 'blur' }], totalCopies: [{ required: true, message: 'total\_copies不能为空', trigger: 'blur' }], availableCopies: [{ required: true, message: 'available\_copies不能为空', trigger: 'blur' }], createTime: [{ required: true, message: 'create\_time不能为空', trigger: 'blur' }], createUser: [{ required: true, message: 'create\_user不能为空', trigger: 'blur' }], updateTime: [{ required: true, message: 'update\_time不能为空', trigger: 'blur' }], updateUser: [{ required: true, message: 'update\_user不能为空', trigger: 'blur' }] } } }, methods: { init (id) { this.dataForm.booksId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.booksId) { this.$http({ url: this.$http.adornUrl(`/generator/books/info/${this.dataForm.booksId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.title = data.books.title this.dataForm.author = data.books.author this.dataForm.totalCopies = data.books.totalCopies this.dataForm.availableCopies = data.books.availableCopies this.dataForm.createTime = data.books.createTime this.dataForm.createUser = data.books.createUser this.dataForm.updateTime = data.books.updateTime this.dataForm.updateUser = data.books.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/books/${this.dataForm.booksId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'title': 'title' || undefined, 'author': 'author' || undefined, 'totalCopies': 'totalCopies' || undefined, 'availableCopies': 'availableCopies' || undefined, 'createTime': 'createTime' || undefined, 'createUser': 'createUser' || undefined, 'updateTime': 'updateTime' || undefined, 'updateUser': 'updateUser' || undefined }) }).then(({data}) => { if (data && data.code === 0) { this.$message({ message: '操作成功', type: 'success', duration: 1500, onClose: () => { this.visible = false this.$emit('refreshDataList') } }) } else { this.$message.error(data.msg) } }) } }) } } }</script>/\*\* \* @description borrow\_records \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/borrowRecords")public class BorrowRecordsController { @Resource private BorrowRecordsService borrowRecordsService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(BorrowRecords borrowRecords){ return borrowRecordsService.insert(borrowRecords); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return borrowRecordsService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(BorrowRecords borrowRecords){ return borrowRecordsService.update(borrowRecords); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return borrowRecordsService.load(id); } /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ @RequestMapping("/pageList") public Map<String, Object> pageList(@RequestParam(required = false, defaultValue = "0") int offset, @RequestParam(required = false, defaultValue = "10") int pagesize) { return borrowRecordsService.pageList(offset, pagesize); }}/\*\* \* @description borrow\_records \* @author 大大的盆 \*/public interface BorrowRecordsService { /\*\* \* 新增 \*/ public Object insert(BorrowRecords borrowRecords); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(BorrowRecords borrowRecords); /\*\* \* 根据主键 id 查询 \*/ public BorrowRecords load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description borrow\_records \* @author 大大的盆 \*/@Servicepublic class BorrowRecordsServiceImpl implements BorrowRecordsService { @Resource private BorrowRecordsMapper borrowRecordsMapper; @Override public Object insert(BorrowRecords borrowRecords) { // valid if (borrowRecords == null) { return ReturnT.error("必要参数缺失"); } borrowRecordsMapper.insert(borrowRecords); return ReturnT.success(); } @Override public Object delete(int id) { int ret = borrowRecordsMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(BorrowRecords borrowRecords) { int ret = borrowRecordsMapper.update(borrowRecords); return ret>0?ReturnT.success():ReturnT.error(); } @Override public BorrowRecords load(int id) { return borrowRecordsMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<BorrowRecords> pageList = borrowRecordsMapper.pageList(offset, pagesize); int totalCount = borrowRecordsMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description borrow\_records \* @author 大大的盆 \*/@Mapper@Repositorypublic interface BorrowRecordsMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(BorrowRecords borrowRecords); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(BorrowRecords borrowRecords); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ BorrowRecords load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<BorrowRecords> pageList(int offset,int pagesize); /\*\* \* 查询 分页查询 count \* @author 大大的盆 \*\*/ int pageListCount(int offset,int pagesize);}<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd"><mapper namespace="HBJZGCXY.com.dao.BorrowRecordsMapper"> <resultMap id="BaseResultMap" type="HBJZGCXY.com.entity.BorrowRecords" > <result column="user\_id" property="userId" /> <result column="book\_id" property="bookId" /> <result column="borrow\_date" property="borrowDate" /> <result column="return\_date" property="returnDate" /> <result column="status" property="status" /> <result column="create\_time" property="createTime" /> <result column="create\_user" property="createUser" /> <result column="update\_time" property="updateTime" /> <result column="update\_user" property="updateUser" /> </resultMap> <sql id="Base\_Column\_List"> user\_id, book\_id, borrow\_date, return\_date, status, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="HBJZGCXY.com.entity.BorrowRecords"> INSERT INTO borrow\_records <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != userId and '' != userId"> user\_id, </if> <if test="null != bookId and '' != bookId"> book\_id, </if> <if test="null != borrowDate and '' != borrowDate"> borrow\_date, </if> <if test="null != returnDate and '' != returnDate"> return\_date, </if> <if test="null != status and '' != status"> status, </if> <if test="null != createTime and '' != createTime"> create\_time, </if> <if test="null != createUser and '' != createUser"> create\_user, </if> <if test="null != updateTime and '' != updateTime"> update\_time, </if> <if test="null != updateUser and '' != updateUser"> update\_user </if> </trim> <trim prefix="values (" suffix=")" suffixOverrides=","> <if test="null != userId and '' != userId"> #{userId}, </if> <if test="null != bookId and '' != bookId"> #{bookId}, </if> <if test="null != borrowDate and '' != borrowDate"> #{borrowDate}, </if> <if test="null != returnDate and '' != returnDate"> #{returnDate}, </if> <if test="null != status and '' != status"> #{status}, </if> <if test="null != createTime and '' != createTime"> #{createTime}, </if> <if test="null != createUser and '' != createUser"> #{createUser}, </if> <if test="null != updateTime and '' != updateTime"> #{updateTime}, </if> <if test="null != updateUser and '' != updateUser"> #{updateUser} </if> </trim> </insert> <delete id="delete" > DELETE FROM borrow\_records WHERE id = #{id} </delete> <update id="update" parameterType="HBJZGCXY.com.entity.BorrowRecords"> UPDATE borrow\_records <set> <if test="null != userId and '' != userId">user\_id = #{userId},</if> <if test="null != bookId and '' != bookId">book\_id = #{bookId},</if> <if test="null != borrowDate and '' != borrowDate">borrow\_date = #{borrowDate},</if> <if test="null != returnDate and '' != returnDate">return\_date = #{returnDate},</if> <if test="null != status and '' != status">status = #{status},</if> <if test="null != createTime and '' != createTime">create\_time = #{createTime},</if> <if test="null != createUser and '' != createUser">create\_user = #{createUser},</if> <if test="null != updateTime and '' != updateTime">update\_time = #{updateTime},</if> <if test="null != updateUser and '' != updateUser">update\_user = #{updateUser}</if> </set> WHERE id = #{id} </update> <select id="load" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM borrow\_records WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM borrow\_records LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM borrow\_records </select></mapper>/\*\* \* @description borrow\_records \* @author 大大的盆 \*/public class BorrowRecords implements Serializable { private static final long serialVersionUID = 1L; private String userId; private String bookId; private String borrowDate; private String returnDate; private String status; private String createTime; private String createUser; private String updateTime; private String updateUser; public BorrowRecords() { } public String getUserId() { return userId; } public void setUserId(String userId) { this.userId = userId; } public String getBookId() { return bookId; } public void setBookId(String bookId) { this.bookId = bookId; } public String getBorrowDate() { return borrowDate; } public void setBorrowDate(String borrowDate) { this.borrowDate = borrowDate; } public String getReturnDate() { return returnDate; } public void setReturnDate(String returnDate) { this.returnDate = returnDate; } public String getStatus() { return status; } public void setStatus(String status) { this.status = status; } public String getCreateTime() { return createTime; } public void setCreateTime(String createTime) { this.createTime = createTime; } public String getCreateUser() { return createUser; } public void setCreateUser(String createUser) { this.createUser = createUser; } public String getUpdateTime() { return updateTime; } public void setUpdateTime(String updateTime) { this.updateTime = updateTime; } public String getUpdateUser() { return updateUser; } public void setUpdateUser(String updateUser) { this.updateUser = updateUser; }}<template> <div class="mod-config"> <el-form :inline="true" :model="dataForm" @keyup.enter.native="getDataList()"> <el-form-item> <el-input v-model="dataForm.key" placeholder="参数名" clearable></el-input> </el-form-item> <el-form-item> <el-button @click="getDataList()">查询</el-button> <el-button v-if="isAuth('generator:borrowRecords:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:borrowRecords:delete')" type="danger" @click="deleteHandle()" :disabled="dataListSelections.length <= 0">批量删除</el-button> </el-form-item> </el-form> <el-table :data="dataList" border v-loading="dataListLoading" @selection-change="selectionChangeHandle" style="width: 100%;"> <el-table-column type="selection" header-align="center" align="center" width="50"> </el-table-column> <el-table-column prop="userId" header-align="center" align="center" label="user\_id"> </el-table-column> <el-table-column prop="bookId" header-align="center" align="center" label="book\_id"> </el-table-column> <el-table-column prop="borrowDate" header-align="center" align="center" label="borrow\_date"> </el-table-column> <el-table-column prop="returnDate" header-align="center" align="center" label="return\_date"> </el-table-column> <el-table-column prop="status" header-align="center" align="center" label="status"> </el-table-column>then(({data}) => { if (data && data.code === 0) { this.dataList = data.page.list this.totalPage = data.page.totalCount } else { this.dataList = [] this.totalPage = 0 } this.dataListLoading = false }) }, // 每页数 sizeChangeHandle (val) { this.pageSize = val this.pageIndex = 1 this.getDataList() }, // 当前页 currentChangeHandle (val) { this.pageIndex = val this.getDataList() }, // 多选 selectionChangeHandle (val) { this.dataListSelections = val }, // 新增 / 修改 addOrUpdateHandle (id) { this.addOrUpdateVisible = true this.$nextTick(() => { this.$refs.addOrUpdate.init(id) }) }, // 删除 deleteHandle (id) { var ids = id ? [id] : this.dataListSelections.map(item => { return item.usersId }) this.$confirm(`确定对[id=${ids.join(',')}]进行[${id ? '删除' : '批量删除'}]操作?`, '提示', { confirmButtonText: '确定', cancelButtonText: '取消', type: 'warning' }).then(() => { this.$http({ url: this.$http.adornUrl('/generator/users/delete'), method: 'post', data: this.$http.adornData(ids, false) }).then(({data}) => { if (data && data.code === 0) { this.$message({ message: '操作成功', type: 'success', duration: 1500, onClose: () => { this.getDataList() } }) } else { this.$message.error(data.msg) } }) }) } } }</script><template> <el-dialog :title="!dataForm.usersId ? '新增' : '修改'" :close-on-click-modal="false" :visible.sync="visible"> <el-form :model="dataForm" :rules="dataRule" ref="dataForm" @keyup.enter.native="dataFormSubmit()" label-width="80px"> <el-form-item label="password" prop="password"> <el-input v-model="dataForm.password" placeholder="password"></el-input> </el-form-item> <el-form-item label="role" prop="role"> <el-input v-model="dataForm.role" placeholder="role"></el-input> </el-form-item> <el-form-item label="create\_time" prop="createTime"> <el-input v-model="dataForm.createTime" placeholder="create\_time"></el-input> </el-form-item> <el-form-item label="create\_user" prop="createUser"> <el-input v-model="dataForm.createUser" placeholder="create\_user"></el-input> </el-form-item> <el-form-item label="update\_time" prop="updateTime"> <el-input v-model="dataForm.updateTime" placeholder="update\_time"></el-input> </el-form-item> <el-form-item label="update\_user" prop="updateUser"> <el-input v-model="dataForm.updateUser" placeholder="update\_user"></el-input> </el-form-item> </el-form> <span slot="footer" class="dialog-footer"> <el-button @click="visible = false">取消</el-button> <el-button type="primary" @click="dataFormSubmit()">确定</el-button> </span> </el-dialog></template><script> export default { data () { return { visible: false, dataForm: { password: '', role: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { password: [{ required: true, message: 'password不能为空', trigger: 'blur' }], role: [{ required: true, message: 'role不能为空', trigger: 'blur' }], createTime: [{ required: true, message: 'create\_time不能为空', trigger: 'blur' }], createUser: [{ required: true, message: 'create\_user不能为空', trigger: 'blur' }], updateTime: [{ required: true, message: 'update\_time不能为空', trigger: 'blur' }], updateUser: [{ required: true, message: 'update\_user不能为空', trigger: 'blur' }] } } }, methods: { init (id) { this.dataForm.usersId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.usersId) { this.$http({ url: this.$http.adornUrl(`/generator/users/info/${this.dataForm.usersId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.password = data.users.password this.dataForm.role = data.users.role this.dataForm.createTime = data.users.createTime this.dataForm.createUser = data.users.createUser this.dataForm.updateTime = data.users.updateTime this.dataForm.updateUser = data.users.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/users/${this.dataForm.usersId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'password': 'password' || undefined, 'role': 'role' || undefined, 'createTime': 'createTime' || undefined, 'createUser': 'createUser' || undefined, 'updateTime': 'updateTime' || undefined, 'updateUser': 'updateUser' || undefined }) }).then(({data}) => { if (data && data.code === 0) { this.$message({ message: '操作成功', type: 'success', duration: 1500, onClose: () => { this.visible = false this.$emit('refreshDataList') } }) } else { this.$message.error(data.msg) } }) } }) } } }</script>/\*\* \* @description statistics\_reports \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/statisticsReports")public class StatisticsReportsController { @Resource private StatisticsReportsService statisticsReportsService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(StatisticsReports statisticsReports){ return statisticsReportsService.insert(statisticsReports); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return statisticsReportsService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(StatisticsReports statisticsReports){ return statisticsReportsService.update(statisticsReports); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return statisticsReportsService.load(id); } /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ @RequestMapping("/pageList") public Map<String, Object> pageList(@RequestParam(required = false, defaultValue = "0") int offset, @RequestParam(required = false, defaultValue = "10") int pagesize) { return statisticsReportsService.pageList(offset, pagesize); }}/\*\* \* @description statistics\_reports \* @author 大大的盆 \*/public interface StatisticsReportsService { /\*\* \* 新增 \*/ public Object insert(StatisticsReports statisticsReports); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(StatisticsReports statisticsReports); /\*\* \* 根据主键 id 查询 \*/ public StatisticsReports load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description statistics\_reports \* @author 大大的盆 \*/@Servicepublic class StatisticsReportsServiceImpl implements StatisticsReportsService { @Resource private StatisticsReportsMapper statisticsReportsMapper; @Override public Object insert(StatisticsReports statisticsReports) { // valid if (statisticsReports == null) { return ReturnT.error("必要参数缺失"); } statisticsReportsMapper.insert(statisticsReports); return ReturnT.success(); } @Override public Object delete(int id) { int ret = statisticsReportsMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(StatisticsReports statisticsReports) { int ret = statisticsReportsMapper.update(statisticsReports); return ret>0?ReturnT.success():ReturnT.error(); } @Override public StatisticsReports load(int id) { return statisticsReportsMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<StatisticsReports> pageList = statisticsReportsMapper.pageList(offset, pagesize); int totalCount = statisticsReportsMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\*

\* \* @description statistics\_reports \* @author 大大的盆 \*/@Mapper@Repositorypublic interface StatisticsReportsMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(StatisticsReports statisticsReports); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(StatisticsReports statisticsReports); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ StatisticsReports load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<StatisticsReports> pageList(int offset,int pagesize); /\*\* \* 查询 分页查询 count \* @author 大大的盆 \*\*/ int pageListCount(int offset,int pagesize);}<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd"><mapper namespace="HBJZGCXY.com.dao.StatisticsReportsMapper"> <resultMap id="BaseResultMap" type="HBJZGCXY.com.entity.StatisticsReports" > <result column="report\_date" property="reportDate" /> <result column="total\_borrows" property="totalBorrows" /> <result column="total\_returns" property="totalReturns" /> <result column="popular\_book\_id" property="popularBookId" /> <result column="create\_time" property="createTime" /> <result column="create\_user" property="createUser" /> <result column="update\_time" property="updateTime" /> <result column="update\_user" property="updateUser" /> </resultMap> <sql id="Base\_Column\_List"> report\_date, total\_borrows, total\_returns, popular\_book\_id, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="HBJZGCXY.com.entity.StatisticsReports"> INSERT INTO statistics\_reports <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != reportDate and '' != reportDate"> report\_date, </if> <if test="null != totalBorrows and '' != totalBorrows"> total\_borrows, </if> <if test="null != totalReturns and '' != totalReturns"> total\_returns, </if> <if test="null != popularBookId and '' != popularBookId"> popular\_book\_id, </if> <if test="null != createTime and '' != createTime"> create\_time, </if> <if test="null != createUser and '' != createUser"> create\_user, </if> <if test="null != updateTime and '' != updateTime"> update\_time, </if> <if test="null != updateUser and '' != updateUser"> update\_user </if> </trim> <trim prefix="values (" suffix=")" suffixOverrides=","> <if test="null != reportDate and '' != reportDate"> #{reportDate}, </if> <if test="null != totalBorrows and '' != totalBorrows"> #{totalBorrows}, </if> <if test="null != totalReturns and '' != totalReturns"> #{totalReturns}, </if> <if test="null != popularBookId and '' != popularBookId"> #{popularBookId}, </if> <if test="null != createTime and '' != createTime"> #{createTime}, </if> <if test="null != createUser and '' != createUser"> #{createUser}, </if> <if test="null != updateTime and '' != updateTime"> #{updateTime}, </if> <if test="null != updateUser and '' != updateUser"> #{updateUser} </if> </trim> </insert> <delete id="delete" > DELETE FROM statistics\_reports WHERE id = #{id} </delete> <update id="update" parameterType="HBJZGCXY.com.entity.StatisticsReports"> UPDATE statistics\_reports <set> <if test="null != reportDate and '' != reportDate">report\_date = #{reportDate},</if> <if test="null != totalBorrows and '' != totalBorrows">total\_borrows = #{totalBorrows},</if> <if test="null != totalReturns and '' != totalReturns">total\_returns = #{totalReturns},</if> <if test="null != popularBookId and '' != popularBookId">popular\_book\_id = #{popularBookId},</if> <if test="null != createTime and '' != createTime">create\_time = #{createTime},</if> <if test="null != createUser and '' != createUser">create\_user = #{createUser},</if> <if test="null != updateTime and '' != updateTime">update\_time = #{updateTime},</if> <if test="null != updateUser and '' != updateUser">update\_user = #{updateUser}</if> </set> WHERE id = #{id} </update> <select id="load" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM statistics\_reports WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM statistics\_reports LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM statistics\_reports </select></mapper>/\*\* \* @description statistics\_reports \* @author 大大的盆 \*/public class StatisticsReports implements Serializable { private static final long serialVersionUID = 1L; private String reportDate; private String totalBorrows; private String totalReturns; private String popularBookId; private String createTime; private String createUser; private String updateTime; private String updateUser; public StatisticsReports() { } public String getReportDate() { return reportDate; } public void setReportDate(String reportDate) { this.reportDate = reportDate; } public String getTotalBorrows() { return totalBorrows; } public void setTotalBorrows(String totalBorrows) { this.totalBorrows = totalBorrows; } public String getTotalReturns() { return totalReturns; } public void setTotalReturns(String totalReturns) { this.totalReturns = totalReturns; } public String getPopularBookId() { return popularBookId; } public void setPopularBookId(String popularBookId) { this.popularBookId = popularBookId; } public String getCreateTime() { return createTime; } public void setCreateTime(String createTime) { this.createTime = createTime; } public String getCreateUser() { return createUser; } public void setCreateUser(String createUser) { this.createUser = createUser; } public String getUpdateTime() { return updateTime; } public void setUpdateTime(String updateTime) { this.updateTime = updateTime; } public String getUpdateUser() { return updateUser; } public void setUpdateUser(String updateUser) { this.updateUser = updateUser; }}<template> <div class="mod-config"> <el-form :inline="true" :model="dataForm" @keyup.enter.native="getDataList()"> <el-form-item> <el-input v-model="dataForm.key" placeholder="参数名" clearable></el-input> </el-form-item> <el-form-item> <el-button @click="getDataList()">查询</el-button> <el-button v-if="isAuth('generator:statisticsReports:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:statisticsReports:delete')" type="danger" @click="deleteHandle()" :disabled="dataListSelections.length <= 0">批量删除</el-button> </el-form-item> </el-form> <el-table :data="dataList" border v-loading="dataListLoading" @selection-change="selectionChangeHandle" style="width: 100%;"> <el-table-column type="selection" header-align="center" align="center" width="50"> </el-table-column> <el-table-column prop="reportDate" header-align="center" align="center" label="report\_date"> </el-table-column> <el-table-column prop="totalBorrows" header-align="center" align="center" label="total\_borrows"> </el-table-column> <el-table-column prop="totalReturns" header-align="center" align="center" label="total\_returns"> </el-table-column> <el-table-column prop="popularBookId" header-align="center" align="center" label="popular\_book\_id"> </el-table-column> <el-table-column prop="createTime" header-align="center" align="center" label="create\_time"> </el-table-column> <el-table-column prop="createUser" header-align="center" align="center" label="create\_user"> </el-table-column> <el-table-column prop="updateTime" header-align="center" align="center" label="update\_time"> </el-table-column> <el-table-column prop="updateUser" header-align="center" align="center" label="update\_user"> </el-table-column> <el-table-column fixed="right" header-align="center" align="center" width="150" label="操作"> <template slot-scope="scope"> <el-button type="text" size="small" @click="addOrUpdateHandle(scope.row.statisticsReportsId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.statisticsReportsId)">删除</el-button> </template> </el-table-column> </el-table> <el-pagination @size-change="sizeChangeHandle" @current-change="currentChangeHandle" :current-page="pageIndex" :page-sizes="[10, 20, 50, 100]" :page-size="pageSize" :total="totalPage" layout="total, sizes, prev, pager, next, jumper"> </el-pagination> <!-- 弹窗, 新增 / 修改 --> <add-or-update v-if="addOrUpdateVisible" ref="addOrUpdate" @refreshDataList="getDataList"></add-or-update> </div></template><script> import AddOrUpdate from './statisticsReports-add-or-update' export default { data () { return { dataForm: { key: '' }, dataList: [], pageIndex: 1, pageSize: 10, totalPage: 0, dataListLoading: false, dataListSelections: [], addOrUpdateVisible: false } }, components: { AddOrUpdate }, activated () { this.getDataList() }, methods: { // 获取数据列表 getDataList () { this.dataListLoading = true this.$http({ url: this.$http.adornUrl('/generator/statisticsReports/list'), method: 'get', params: this.$http.adornParams({ 'page': this.pageIndex, 'limit': this.pageSize, 'key': this.dataForm.key }) }).then(({data}) => { if (data && data.code === 0) { this.dataList = data.page.list this.totalPage = data.page.totalCount } else { this.dataList = [] this.totalPage = 0 } this.dataListLoading = false }) }, // 每页数 sizeChangeHandle (val) { this.pageSize = val this.pageIndex = 1 this.getDataList() }, // 当前页 currentChangeHandle (val) { this.pageIndex = val this.getDataList() }, // 多选 selectionChangeHandle (val) { this.dataListSelections = val }, // 新增 / 修改 addOrUpdateHandle (id) { this.addOrUpdateVisible = true this.$nextTick(() => { this.$refs.addOrUpdate.init(id) }) }, // 删除 deleteHandle (id) { var ids = id ? [id] : this.dataListSelections.map(item => { return item.statisticsReportsId }) this.$confirm(`确定对[id=${ids.join(',')}]进行[${id ? '删除' : '批量删除'}]操作?`, '提示', { confirmButtonText: '确定', cancelButtonText: '取消', type: 'warning' }).then(() => { this.$http({ url: this.$http.adornUrl('/generator/statisticsReports/delete'), method: 'post', data: this.$http.adornData(ids, false) }).then(({data}) => { if (data && data.code === 0) { this.$message({ message: '操作成功', type: 'success', duration: 1500, onClose: () => { this.getDataList() } }) } else { this.$message.error(data.msg) } }) }) } } }</script><template> <el-dialog :title="!dataForm.statisticsReportsId ? '新增' : '修改'" :close-on-click-modal="false" :visible.sync="visible"> <el-form :model="dataForm" :rules="dataRule" ref="dataForm" @keyup.enter.native="dataFormSubmit()" label-width="80px"> <el-form-item label="report\_date" prop="reportDate"> <el-input v-model="dataForm.reportDate" placeholder="report\_date"></el-input> </el-form-item> <el-form-item label="total\_borrows" prop="totalBorrows"> <el-input v-model="dataForm.totalBorrows" placeholder="total\_borrows"></el-input> </el-form-item> <el-form-item label="total\_returns" prop="totalReturns"> <el-input v-model="dataForm.totalReturns" placeholder="total\_returns"></el-input> </el-form-item> <el-form-item label="popular\_book\_id" prop="popularBookId"> <el-input v-model="dataForm.popularBookId" placeholder="popular\_book\_id"></el-input> </el-form-item> <el-form-item label="create\_time" prop="createTime"> <el-input v-model="dataForm.createTime" placeholder="create\_time"></el-input> </el-form-item> <el-form-item label="create\_user" prop="createUser"> <el-input v-model="dataForm.createUser" placeholder="create\_user"></el-input> </el-form-item> <el-form-item label="update\_time" prop="updateTime"> <el-input v-model="dataForm.updateTime" placeholder="update\_time"></el-input> </el-form-item> <el-form-item label="update\_user" prop="updateUser"> <el-input v-model="dataForm.updateUser" placeholder="update\_user"></el-input> </el-form-item> </el-form> <span slot="footer" class="dialog-footer"> <el-button @click="visible = false">取消</el-button> <el-button type="primary" @click="dataFormSubmit()">确定</el-button> </span> </el-dialog></template><script> export default { data () { return { visible: false, dataForm: { reportDate: '', totalBorrows: '', totalReturns: '', popularBookId: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { reportDate: [{ required: true, message: 'report\_date不能为空', trigger: 'blur' }], totalBorrows: [{ required: true, message: 'total\_borrows不能为空', trigger: 'blur' }], totalReturns: [{ required: true, message: 'total\_returns不能为空', trigger: 'blur' }], popularBookId: [{ required: true, message: 'popular\_book\_id不能为空', trigger: 'blur' }], createTime: [{ required: true, message: 'create\_time不能为空', trigger: 'blur' }], createUser: [{ required: true, message: 'create\_user不能为空', trigger: 'blur' }], updateTime: [{ required: true, message: 'update\_time不能为空', trigger: 'blur' }], updateUser: [{ required: true, message: 'update\_user不能为空', trigger: 'blur' }] } } }, methods: { init (id) { this.dataForm.statisticsReportsId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.statisticsReportsId) { this.$http({ url: this.$http.adornUrl(`/generator/statisticsReports/info/${this.dataForm.statisticsReportsId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.reportDate = data.statisticsReports.reportDate this.dataForm.totalBorrows = data.statisticsReports.totalBorrows this.dataForm.totalReturns = data.statisticsReports.totalReturns this.dataForm.popularBookId = data.statisticsReports.popularBookId this.dataForm.createTime = data.statisticsReports.createTime this.dataForm.createUser = data.statisticsReports.createUser this.dataForm.updateTime = data.statisticsReports.updateTime this.dataForm.updateUser = data.statisticsReports.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/statisticsReports/${this.dataForm.statisticsReportsId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'reportDate': 'reportDate' || undefined, 'totalBorrows': 'totalBorrows' || undefined, 'totalReturns': 'totalReturns' || undefined, 'popularBookId': 'popularBookId' || undefined, 'createTime': 'createTime' || undefined, 'createUser': 'createUser' || undefined, 'updateTime': 'updateTime' || undefined, 'updateUser': 'updateUser' || undefined }) }).then(({data}) => { if (data && data.code === 0) { this.$message({ message: '操作成功', type: 'success', duration: 1500, onClose: () => { this.visible = false this.$emit('refreshDataList') } }) } else { this.$message.error(data.msg) } }) } }) } } }</script>/\*\* \* @description roles \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/roles")public class RolesController { @Resource private RolesService rolesService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(Roles roles){ return rolesService.insert(roles); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return rolesService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(Roles roles){ return rolesService.update(roles); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return rolesService.load(id); } /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ @RequestMapping("/pageList") public Map<String, Object> pageList(@RequestParam(required = false, defaultValue = "0") int offset, @RequestParam(required = false, defaultValue = "10") int pagesize) { return rolesService.pageList(offset, pagesize); }}/\*\* \* @description roles \* @author 大大的盆 \*/public interface RolesService { /\*\* \* 新增 \*/ public Object insert(Roles roles); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(Roles roles); /\*\* \* 根据主键 id 查询 \*/ public Roles load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description roles \* @author 大大的盆 \*/@Servicepublic class RolesServiceImpl implements RolesService { @Resource private RolesMapper rolesMapper; @Override public Object insert(Roles roles) { // valid if (roles == null) { return ReturnT.error("必要参数缺失"); } rolesMapper.insert(roles); return ReturnT.success(); } @Override public Object delete(int id) { int ret = rolesMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(Roles roles) { int ret = rolesMapper.update(roles); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Roles load(int id) { return rolesMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<Roles> pageList = rolesMapper.pageList(offset, pagesize); int totalCount = rolesMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description roles \* @author 大大的盆 \*/@Mapper@Repositorypublic interface RolesMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(Roles roles); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(Roles roles); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ Roles load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<Roles> pageList(int offset,int pagesize); /\*\* \* 查询 分页查询 count \* @author 大大的盆 \*\*/ int pageListCount(int offset,int pagesize);}<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd"><mapper namespace="HBJZGCXY.com.dao.RolesMapper"> <resultMap id="BaseResultMap" type="HBJZGCXY.com.entity.Roles" > <result column="permissions" property="permissions" /> <result column="create\_time" property="createTime" /> <result column="create\_user" property="createUser" /> <result column="update\_time" property="updateTime" /> <result column="update\_user" property="updateUser" /> </resultMap> <sql id="Base\_Column\_List"> permissions, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="HBJZGCXY.com.entity.Roles"> INSERT INTO roles <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != permissions and '' != permissions"> permissions, </if> <if test="null != createTime and '' != createTime"> create\_time, </if> <if test="null != createUser and '' != createUser"> create\_user, </if> <if test="null != updateTime and '' != updateTime"> update\_time, </if> <if test="null != updateUser and '' != updateUser"> update\_user </if> </trim> <trim prefix="values (" suffix=")" suffixOverrides=","> <if test="null != permissions and '' != permissions"> #{permissions}, </if> <if test="null != createTime and '' != createTime"> #{createTime}, </if> <if test="null != createUser and '' != createUser"> #{createUser}, </if> <if test="null != updateTime and '' != updateTime"> #{updateTime}, </if> <if test="null != updateUser and '' != updateUser"> #{updateUser} </if> </trim> </insert> <delete id="delete" > DELETE FROM roles WHERE id = #{id} </delete> <update id="update" parameterType="HBJZGCXY.com.entity.Roles"> UPDATE roles <set> <if test="null != permissions and '' != permissions">permissions = #{permissions},</if> <if test="null != createTime and '' != createTime">create\_time = #{createTime},</if> <if test="null != createUser and '' != createUser">create\_user = #{createUser},</if> <if test="null != updateTime and '' != updateTime">update\_time = #{updateTime},</if> <if test="null != updateUser and '' != updateUser">update\_user = #{updateUser}</if> </set> WHERE id = #{id} </update> <select id="load" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM roles WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM roles LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM roles </select></mapper>/\*\* \* @description roles \* @author 大大的盆 \*/public class Roles implements Serializable { private static final long serialVersionUID = 1L; private String permissions; private String createTime; private String createUser; private String updateTime; private String updateUser; public Roles() { } public String getPermissions() { return permissions; } public void setPermissions(String permissions) { this.permissions = permissions; } public String getCreateTime() { return createTime; } public void setCreateTime(String createTime) { this.createTime = createTime; } public String getCreateUser() { return createUser; } public void setCreateUser(String createUser) { this.createUser = createUser; } public String getUpdateTime() { return updateTime; } public void setUpdateTime(String updateTime) { this.updateTime = updateTime; } public String getUpdateUser() { return updateUser; } public void setUpdateUser(String updateUser) { this.updateUser = updateUser; }}<template> <div class="mod-config"> <el-form :inline="true" :model="dataForm" @keyup.enter.native="getDataList()"> <el-form-item> <el-input v-model="dataForm.key" placeholder="参数名" clearable></el-input> </el-form-item> <el-form-item> <el-button @click="getDataList()">查询</el-button> <el-button v-if="isAuth('generator:roles:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:roles:delete')" type="danger" @click="deleteHandle()" :disabled="dataListSelections.length <= 0">批量删除</el-button> </el-form-item> </el-form> <el-table :data="dataList" border v-loading="dataListLoading" @selection-change="selectionChangeHandle" style="width: 100%;"> <el-table-column type="selection" header-align="center" align="center" width="50"> </el-table-column> <el-table-column prop="permissions" header-align="center" align="center" label="permissions"> </el-table-column> <el-table-column prop="createTime" header-align="center" align="center" label="create\_time"> </el-table-column> <el-table-column prop="createUser" header-align="center" align="center" label="create\_user"> </el-table-column> <el-table-column prop="updateTime" header-align="center" align="center" label="update\_time"> </el-table-column> <el-table-column prop="updateUser" header-align="center" align="center" label="update\_user"> </el-table-column> <el-table-column fixed="right" header-align="center" align="center" width="150" label="操作"> <template slot-scope="scope"> <el-button type="text" size="small" @click="addOrUpdateHandle(scope.row.rolesId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.rolesId)">删除</el-button> </template> </el-table-column> </el-table> <el-pagination @size-change="sizeChangeHandle" @current-change="currentChangeHandle" :current-page="pageIndex" :page-sizes="[10, 20, 50, 100]" :page-size="pageSize" :total="totalPage" layout="total, sizes, prev, pager, next, jumper"> </el-pagination> <!-- 弹窗, 新增 / 修改 --> <add-or-update v-if="addOrUpdateVisible" ref="addOrUpdate" @refreshDataList="getDataList"></add-or-update> </div></template><script> import AddOrUpdate from './roles-add-or-update' export default { data () { return { dataForm: { key: '' }, dataList: [], pageIndex: 1, pageSize: 10, totalPage: 0, dataListLoading: false, dataListSelections: [], addOrUpdateVisible: false } }, components: { AddOrUpdate }, activated () { this.getDataList() }, methods: { // 获取数据列表 getDataList () { this.dataListLoading = true this.$http({ url: this.$http.adornUrl('/generator/roles/list'), method: 'get', params: this.$http.adornParams({ 'page': this.pageIndex, 'limit': this.pageSize, 'key': this.dataForm.key }) }).then(({data}) => { if (data && data.code === 0) { this.dataList = data.page.list this.totalPage = data.page.totalCount } else { this.dataList = [] this.totalPage = 0 } this.dataListLoading = false }) }, // 每页数 sizeChangeHandle (val) { this.pageSize = val this.pageIndex = 1 this.getDataList() }, // 当前页 currentChangeHandle (val) { this.pageIndex = val this.getDataList() }, // 多选 selectionChangeHandle (val) { this.dataListSelections = val }, // 新增 / 修改 addOrUpdateHandle (id) { this.addOrUpdateVisible = true this.$nextTick(() => { this.$refs.addOrUpdate.init(id) }) }, // 删除 deleteHandle (id) { var ids = id ? [id] : this.dataListSelections.map(item => { return item.rolesId }) this.$confirm(`确定对[id=${ids.join(',')}]进行[${id ? '删除' : '批量删除'}]操作?`, '提示', { confirmButtonText: '确定', cancelButtonText: '取消', type: 'warning' }).then(() => { this.$http({ url: this.$http.adornUrl('/generator/roles/delete'), method: 'post', data: this.$http.adornData(ids, false) }).then(({data}) => { if (data && data.code === 0) { this.$message({ message: '操作成功', type: 'success', duration: 1500, onClose: () => { this.getDataList() } }) } else { this.$message.error(data.msg) } }) }) } } }</script><template> <el-dialog :title="!dataForm.rolesId ? '新增' : '修改'" :close-on-click-modal="false" :visible.sync="visible"> <el-form :model="dataForm" :rules="dataRule" ref="dataForm" @keyup.enter.native="dataFormSubmit()" label-width="80px"> <el-form-item label="permissions" prop="permissions"> <el-input v-model="dataForm.permissions" placeholder="permissions"></el-input> </el-form-item> <el-form-item label="create\_time" prop="createTime"> <el-input v-model="dataForm.createTime" placeholder="create\_time"></el-input> </el-form-item> <el-form-item label="create\_user" prop="createUser"> <el-input v-model="dataForm.createUser" placeholder="create\_user"></el-input> </el-form-item> <el-form-item label="update\_time" prop="updateTime"> <el-input v-model="dataForm.updateTime" placeholder="update\_time"></el-input> </el-form-item> <el-form-item label="update\_user" prop="updateUser"> <el-input v-model="dataForm.updateUser" placeholder="update\_user"></el-input> </el-form-item> </el-form> <span slot="footer" class="dialog-footer"> <el-button @click="visible = false">取消</el-button> <el-button type="primary" @click="dataFormSubmit()">确定</el-button> </span> </el-dialog></template><script> export default { data () { return { visible: false, dataForm: { permissions: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { permissions: [{ required: true, message: 'permissions不能为空', trigger: 'blur' }], createTime: [{ required: true, message: 'create\_time不能为空', trigger: 'blur' }], createUser: [{ required: true, message: 'create\_user不能为空', trigger: 'blur' }], updateTime: [{ required: true, message: 'update\_time不能为空', trigger: 'blur' }], updateUser: [{ required: true, message: 'update\_user不能为空', trigger: 'blur' }] } } }, methods: { init (id) { this.dataForm.rolesId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.rolesId) { this.$http({ url: this.$http.adornUrl(`/generator/roles/info/${this.dataForm.rolesId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.permissions = data.roles.permissions this.dataForm.createTime = data.roles.createTime this.dataForm.createUser = data.roles.createUser this.dataForm.updateTime = data.roles.updateTime this.dataForm.updateUser = data.roles.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/roles/${this.dataForm.rolesId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'permissions': 'permissions' || undefined, 'createTime': 'createTime' || undefined, 'createUser': 'createUser' || undefined, 'updateTime': 'updateTime' || undefined, 'updateUser': 'updateUser' || undefined }) }).then(({data}) => { if (data && data.code === 0) { this.$message({ message: '操作成功', type: 'success', duration: 1500, onClose: () => { this.visible = false this.$emit('refreshDataList') } }) } else { this.$message.error(data.msg) } }) } }) } } }</script>