<!DOCTYPE html><html lang="zh-CN"><head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <title>瑞贤美味速递平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>瑞贤美味速递平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item active"><span>餐厅信息浏览</span></li></a> <a ><li class="rz-nav-item"><span>菜品选择</span></li></a> <a ><li class="rz-nav-item"><span>订单确认</span></li></a> <a ><li class="rz-nav-item"><span>在线支付</span></li></a> <a ><li class="rz-nav-item"><span>订单跟踪</span></li></a> <a ><li class="rz-nav-item"><span>用户评价</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>餐厅信息浏览</h2> <div class="filter"> <input type="text" placeholder="搜索餐厅..." /> <select> <option value="">选择菜系</option> <option value="中餐">中餐</option> <option value="西餐">西餐</option> <option value="快餐">快餐</option> </select> <input type="number" placeholder="最小评分" min="1" max="5" /> <button>筛选</button> </div> <ul class="restaurant-list"> <li class="restaurant-item"> <h3>餐厅名称1</h3> <p>地址: 餐厅地址1</p> <p>联系电话: 123456789</p> <p>营业时间: 11:00 - 22:00</p> <p>用户评价: ★★★★☆</p> <button class="menu-button">查看菜单</button> </li> <li class="restaurant-item"> <h3>餐厅名称2</h3> <p>地址: 餐厅地址2</p> <p>联系电话: 987654321</p> <p>营业时间: 10:00 - 21:00</p> <p>用户评价: ★★★★★</p> <button class="menu-button">查看菜单</button> </li> <!-- 更多餐厅信息 --> </ul> </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>瑞贤美味速递平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>瑞贤美味速递平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>餐厅信息浏览</span></li></a> <a ><li class="rz-nav-item active"><span>菜品选择</span></li></a> <a ><li class="rz-nav-item"><span>订单确认</span></li></a> <a ><li class="rz-nav-item"><span>在线支付</span></li></a> <a ><li class="rz-nav-item"><span>订单跟踪</span></li></a> <a ><li class="rz-nav-item"><span>用户评价</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>菜单</h2> <div class="dish-item"> <img src="dish1.jpg" alt="菜品1" class="dish-image"> <div class="dish-details"> <div class="dish-name">菜品1</div> <div class="dish-price">¥20.00</div> <div class="dish-description">这是菜品1的描述，味道鲜美，营养丰富。</div> <div class="quantity"> <button class="add-to-cart">添加到购物车</button> </div> </div> </div> <div class="dish-item"> <img src="dish2.jpg" alt="菜品2" class="dish-image"> <div class="dish-details"> <div class="dish-name">菜品2</div> <div class="dish-price">¥35.00</div> <div class="dish-description">这是菜品2的描述，口感独特，深受喜爱。</div> <div class="quantity"> <button class="add-to-cart">添加到购物车</button> </div> </div> </div> <div class="cart-summary">购物车总价：¥55.00</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>瑞贤美味速递平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>瑞贤美味速递平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>餐厅信息浏览</span></li></a> <a ><li class="rz-nav-item"><span>菜品选择</span></li></a> <a ><li class="rz-nav-item active"><span>订单确认</span></li></a> <a ><li class="rz-nav-item"><span>在线支付</span></li></a> <a ><li class="rz-nav-item"><span>订单跟踪</span></li></a> <a ><li class="rz-nav-item"><span>用户评价</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>订单确认</h2> <div class="order-summary"> <h3>所选菜品</h3> <p>菜品名称：红烧肉</p> <p>数量：2</p> <p>餐厅信息：美味餐厅</p> <p>配送地址：北京市朝阳区XX路XX号</p> <p>预计配送时间：30分钟</p> <p>运费：￥5.00</p> </div> <div class="order-actions"> <button class="btn">修改订单</button> <button class="btn">提交订单</button> </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>瑞贤美味速递平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>瑞贤美味速递平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>餐厅信息浏览</span></li></a> <a ><li class="rz-nav-item"><span>菜品选择</span></li></a> <a ><li class="rz-nav-item"><span>订单确认</span></li></a> <a ><li class="rz-nav-item active"><span>在线支付</span></li></a> <a ><li class="rz-nav-item"><span>订单跟踪</span></li></a> <a ><li class="rz-nav-item"><span>用户评价</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>在线支付</h2> <form> <div class="payment-method"> <label><input type="radio" name="payment" value="credit\_card"> 信用卡</label> <label><input type="radio" name="payment" value="alipay"> 支付宝</label> <label><input type="radio" name="payment" value="wechat\_pay"> 微信支付</label> </div> <div> <label for="amount">支付金额：</label> <input type="number" id="amount" name="amount" required> </div> <div> <label for="card\_info">支付信息：</label> <input type="text" id="card\_info" name="card\_info" placeholder="输入相关支付信息" required> </div> <button type="submit" class="btn-confirm">确认支付</button> </form> </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>瑞贤美味速递平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>瑞贤美味速递平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>餐厅信息浏览</span></li></a> <a ><li class="rz-nav-item"><span>菜品选择</span></li></a> <a ><li class="rz-nav-item"><span>订单确认</span></li></a> <a ><li class="rz-nav-item"><span>在线支付</span></li></a> <a ><li class="rz-nav-item active"><span>订单跟踪</span></li></a> <a ><li class="rz-nav-item"><span>用户评价</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>订单跟踪</h2> <div class="order-status"> <h3>当前配送状态</h3> <ul> <li>接单</li> <li>备餐</li> <li>出发配送</li> <li>预计到达时间: 预计 20 分钟后</li> </ul> </div> <button class="refresh-button">刷新状态</button> <button class="contact-button">联系配送员</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>瑞贤美味速递平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>瑞贤美味速递平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>餐厅信息浏览</span></li></a> <a ><li class="rz-nav-item"><span>菜品选择</span></li></a> <a ><li class="rz-nav-item"><span>订单确认</span></li></a> <a ><li class="rz-nav-item"><span>在线支付</span></li></a> <a ><li class="rz-nav-item"><span>订单跟踪</span></li></a> <a ><li class="rz-nav-item active"><span>用户评价</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>用户评价</h2> <form> <div class="rz-form-group"> <label for="restaurant-rating">餐厅评分:</label> <div class="rz-rating"> <input type="radio" id="restaurant-5" name="restaurant-rating" value="5"> <label for="restaurant-5">★</label> <input type="radio" id="restaurant-4" name="restaurant-rating" value="4"> <label for="restaurant-4">★</label> <input type="radio" id="restaurant-3" name="restaurant-rating" value="3"> <label for="restaurant-3">★</label> <input type="radio" id="restaurant-2" name="restaurant-rating" value="2"> <label for="restaurant-2">★</label> <input type="radio" id="restaurant-1" name="restaurant-rating" value="1"> <label for="restaurant-1">★</label> </div> </div> <div class="rz-form-group"> <label for="dish-rating">菜品评分:</label> <div class="rz-rating"> <input type="radio" id="dish-5" name="dish-rating" value="5"> <label for="dish-5">★</label> <input type="radio" id="dish-4" name="dish-rating" value="4"> <label for="dish-4">★</label> <input type="radio" id="dish-3" name="dish-rating" value="3"> <label for="dish-3">★</label> <input type="radio" id="dish-2" name="dish-rating" value="2"> <label for="dish-2">★</label> <input type="radio" id="dish-1" name="dish-rating" value="1"> <label for="dish-1">★</label> </div> </div> <div class="rz-form-group"> <label for="user-comments">个人意见和建议:</label> <textarea id="user-comments" rows="4" placeholder="请留下您的评论..."></textarea> </div> <button type="submit" class="rz-button">提交评价</button> </form> </div> </main> </div></body></html>

/\*\* \* @description user \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/user")public class UserController { @Resource private UserService userService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(User user){ return userService.insert(user); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return userService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(User user){ return userService.update(user); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return userService.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 userService.pageList(offset, pagesize); }}/\*\* \* @description user \* @author 大大的盆 \*/public interface UserService { /\*\* \* 新增 \*/ public Object insert(User user); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(User user); /\*\* \* 根据主键 id 查询 \*/ public User load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description user \* @author 大大的盆 \*/@Servicepublic class UserServiceImpl implements UserService { @Resource private UserMapper userMapper; @Override public Object insert(User user) { // valid if (user == null) { return ReturnT.error("必要参数缺失"); } userMapper.insert(user); return ReturnT.success(); } @Override public Object delete(int id) { int ret = userMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(User user) { int ret = userMapper.update(user); return ret>0?ReturnT.success():ReturnT.error(); } @Override public User load(int id) { return userMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<User> pageList = userMapper.pageList(offset, pagesize); int totalCount = userMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description user \* @author 大大的盆 \*/@Mapper@Repositorypublic interface UserMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(User user); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(User user); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ User load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<User> 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="GDL.com.dao.UserMapper"> <resultMap id="BaseResultMap" type="GDL.com.entity.User" > <result column="username" property="username" /> <result column="password" property="password" /> <result column="email" property="email" /> <result column="phone\_number" property="phoneNumber" /> <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"> username, password, email, phone\_number, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="GDL.com.entity.User"> INSERT INTO user <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != username and '' != username"> username, </if> <if test="null != password and '' != password"> password, </if> <if test="null != email and '' != email"> email, </if> <if test="null != phoneNumber and '' != phoneNumber"> phone\_number, </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 != username and '' != username"> #{username}, </if> <if test="null != password and '' != password"> #{password}, </if> <if test="null != email and '' != email"> #{email}, </if> <if test="null != phoneNumber and '' != phoneNumber"> #{phoneNumber}, </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 user WHERE id = #{id} </delete> <update id="update" parameterType="GDL.com.entity.User"> UPDATE user <set> <if test="null != username and '' != username">username = #{username},</if> <if test="null != password and '' != password">password = #{password},</if> <if test="null != email and '' != email">email = #{email},</if> <if test="null != phoneNumber and '' != phoneNumber">phone\_number = #{phoneNumber},</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 user WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM user LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM user </select></mapper>/\*\* \* @description user \* @author 大大的盆 \*/public class User implements Serializable { private static final long serialVersionUID = 1L; private String username; private String password; private String email; private String phoneNumber; private String createTime; private String createUser; private String updateTime; private String updateUser; public User() { } public String getUsername() { return username; } public void setUsername(String username) { this.username = username; } public String getPassword() { return password; } public void setPassword(String password) { this.password = password; } public String getEmail() { return email; } public void setEmail(String email) { this.email = email; } public String getPhoneNumber() { return phoneNumber; } public void setPhoneNumber(String phoneNumber) { this.phoneNumber = phoneNumber; } 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:user:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:user: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="username" header-align="center" align="center" label="username"> </el-table-column> <el-table-column prop="password" header-align="center" align="center" label="password"> </el-table-column> <el-table-column prop="email" header-align="center" align="center" label="email"> </el-table-column> <el-table-column prop="phoneNumber" header-align="center" align="center" label="phone\_number"> </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.userId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.userId)">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 './user-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/user/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.$refs.addOrUpdate.init(id) }) }, // 删除 deleteHandle (id) { var ids = id ? [id] : this.dataListSelections.map(item => { return item.userId }) this.$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/user/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.userId ? '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="username" prop="username"> <el-input v-model="dataForm.username" placeholder="username"></el-input> </el-form-item> <el-form-item label="password" prop="password"> <el-input v-model="dataForm.password" placeholder="password"></el-input> </el-form-item> <el-form-item label="email" prop="email"> <el-input v-model="dataForm.email" placeholder="email"></el-input> </el-form-item> <el-form-item label="phone\_number" prop="phoneNumber"> <el-input v-model="dataForm.phoneNumber" placeholder="phone\_number"></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: { username: '', password: '', email: '', phoneNumber: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { username: [{ required: true, message: 'username不能为空', trigger: 'blur' }], password: [{ required: true, message: 'password不能为空', trigger: 'blur' }], email: [{ required: true, message: 'email不能为空', trigger: 'blur' }], phoneNumber: [{ required: true, message: 'phone\_number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.userId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.userId) { this.$http({ url: this.$http.adornUrl(`/generator/user/info/${this.dataForm.userId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.username = data.user.username this.dataForm.password = data.user.password this.dataForm.email = data.user.email this.dataForm.phoneNumber = data.user.phoneNumber this.dataForm.createTime = data.user.createTime this.dataForm.createUser = data.user.createUser this.dataForm.updateTime = data.user.updateTime this.dataForm.updateUser = data.user.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/user/${this.dataForm.userId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'username': 'username' || undefined, 'password': 'password' || undefined, 'email': 'email' || undefined, 'phoneNumber': 'phoneNumber' || 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 restaurant \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/restaurant")public class RestaurantController { @Resource private RestaurantService restaurantService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(Restaurant restaurant){ return restaurantService.insert(restaurant); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return restaurantService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(Restaurant restaurant){ return restaurantService.update(restaurant); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return restaurantService.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 restaurantService.pageList(offset, pagesize); }}/\*\* \* @description restaurant \* @author 大大的盆 \*/public interface RestaurantService { /\*\* \* 新增 \*/ public Object insert(Restaurant restaurant); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(Restaurant restaurant); /\*\* \* 根据主键 id 查询 \*/ public Restaurant load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description restaurant \* @author 大大的盆 \*/@Servicepublic class RestaurantServiceImpl implements RestaurantService { @Resource private RestaurantMapper restaurantMapper; @Override public Object insert(Restaurant restaurant) { // valid if (restaurant == null) { return ReturnT.error("必要参数缺失"); } restaurantMapper.insert(restaurant); return ReturnT.success(); } @Override public Object delete(int id) { int ret = restaurantMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(Restaurant restaurant) { int ret = restaurantMapper.update(restaurant); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Restaurant load(int id) { return restaurantMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<Restaurant> pageList = restaurantMapper.pageList(offset, pagesize); int totalCount = restaurantMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description restaurant \* @author 大大的盆 \*/@Mapper@Repositorypublic interface RestaurantMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(Restaurant restaurant); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(Restaurant restaurant); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ Restaurant load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<Restaurant> 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="GDL.com.dao.RestaurantMapper"> <resultMap id="BaseResultMap" type="GDL.com.entity.Restaurant" > <result column="restaurant\_name" property="restaurantName" /> <result column="address" property="address" /> <result column="phone\_number" property="phoneNumber" /> <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"> restaurant\_name, address, phone\_number, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="GDL.com.entity.Restaurant"> INSERT INTO restaurant <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != restaurantName and '' != restaurantName"> restaurant\_name, </if> <if test="null != address and '' != address"> address, </if> <if test="null != phoneNumber and '' != phoneNumber"> phone\_number, </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 != restaurantName and '' != restaurantName"> #{restaurantName}, </if> <if test="null != address and '' != address"> #{address}, </if> <if test="null != phoneNumber and '' != phoneNumber"> #{phoneNumber}, </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 restaurant WHERE id = #{id} </delete> <update id="update" parameterType="GDL.com.entity.Restaurant"> UPDATE restaurant <set> <if test="null != restaurantName and '' != restaurantName">restaurant\_name = #{restaurantName},</if> <if test="null != address and '' != address">address = #{address},</if> <if test="null != phoneNumber and '' != phoneNumber">phone\_number = #{phoneNumber},</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 restaurant WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM restaurant LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM restaurant </select></mapper>/\*\* \* @description restaurant \* @author 大大的盆 \*/public class Restaurant implements Serializable { private static final long serialVersionUID = 1L; private String restaurantName; private String address; private String phoneNumber; private String createTime; private String createUser; private String updateTime; private String updateUser; public Restaurant() { } public String getRestaurantName() { return restaurantName; } public void setRestaurantName(String restaurantName) { this.restaurantName = restaurantName; } public String getAddress() { return address; } public void setAddress(String address) { this.address = address; } public String getPhoneNumber() { return phoneNumber; } public void setPhoneNumber(String phoneNumber) { this.phoneNumber = phoneNumber; } 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:restaurant:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:restaurant: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="restaurantName" header-align="center" align="center" label="restaurant\_name"> </el-table-column> <el-table-column prop="address" header-align="center" align="center" label="address"> </el-table-column> <el-table-column prop="phoneNumber" header-align="center" align="center" label="phone\_number"> </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.restaurantId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.restaurantId)">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 './restaurant-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.$http.adornUrl('/generator/restaurant/list'), method: 'get', params: this.$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 = [] <el-dialog <el-form :model="dataForm" :rules="dataRule" ref="dataForm" @keyup.enter.native="dataFormSubmit()" label-width="80px"> <el-form-item label="user\_id" prop="userId"> <el-input v-model="dataForm.userId" placeholder="user\_id"></el-input> </el-form-item> <el-form-item label="restaurant\_id" prop="restaurantId"> <el-input v-model="dataForm.restaurantId" placeholder="restaurant\_id"></el-input> </el-form-item> <el-form-item label="total\_amount" prop="totalAmount"> <el-input v-model="dataForm.totalAmount" placeholder="total\_amount"></el-input> </el-form-item> <el-form-item label="order\_status" prop="orderStatus"> <el-input v-model="dataForm.orderStatus" placeholder="order\_status"></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: { userId: '', restaurantId: '', totalAmount: '', orderStatus: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { userId: [{ required: true, message: 'user\_id不能为空', trigger: 'blur' }], restaurantId: [{ required: true, message: 'restaurant\_id不能为空', trigger: 'blur' }], totalAmount: [{ required: true, message: 'total\_amount不能为空', trigger: 'blur' }], orderStatus: [{ required: true, message: 'order\_status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.ordersId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.ordersId) { this.$http({ url: this.$http.adornUrl(`/generator/orders/info/${this.dataForm.ordersId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.userId = data.orders.userId this.dataForm.restaurantId = data.orders.restaurantId this.dataForm.totalAmount = data.orders.totalAmount this.dataForm.orderStatus = data.orders.orderStatus this.dataForm.createTime = data.orders.createTime this.dataForm.createUser = data.orders.createUser this.dataForm.updateTime = data.orders.updateTime this.dataForm.updateUser = data.orders.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/orders/${this.dataForm.ordersId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'userId': 'userId' || undefined, 'restaurantId': 'restaurantId' || undefined, 'totalAmount': 'totalAmount' || undefined, 'orderStatus': 'orderStatus' || undefined, 'createTime': 'createTime' || undefined, 'createUser': 'createUser' || undefined, 'updateTime': 'updateTime' || undefined, 'updateUser': 'updateUser' || undefined }) }).then(({data}) => { if (data && data.code === 0) { this.$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 payment \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/payment")public class PaymentController { @Resource private PaymentService paymentService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(Payment payment){ return paymentService.insert(payment); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return paymentService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(Payment payment){ return paymentService.update(payment); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return paymentService.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 paymentService.pageList(offset, pagesize); }}/\*\* \* @description payment \* @author 大大的盆 \*/public interface PaymentService { /\*\* \* 新增 \*/ public Object insert(Payment payment); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(Payment payment); /\*\* \* 根据主键 id 查询 \*/ public Payment load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description payment \* @author 大大的盆 \*/@Servicepublic class PaymentServiceImpl implements PaymentService { @Resource private PaymentMapper paymentMapper; @Override public Object insert(Payment payment) { // valid if (payment == null) { return ReturnT.error("必要参数缺失"); } paymentMapper.insert(payment); return ReturnT.success(); } @Override public Object delete(int id) { int ret = paymentMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(Payment payment) { int ret = paymentMapper.update(payment); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Payment load(int id) { return paymentMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<Payment> pageList = paymentMapper.pageList(offset, pagesize); int totalCount = paymentMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description payment \* @author 大大的盆 \*/@Mapper@Repositorypublic interface PaymentMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(Payment payment); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(Payment payment); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ Payment load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<Payment> 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="GDL.com.dao.PaymentMapper"> <resultMap id="BaseResultMap" type="GDL.com.entity.Payment" > <result column="order\_id" property="orderId" /> <result column="payment\_method" property="paymentMethod" /> <result column="payment\_status" property="paymentStatus" /> <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"> order\_id, payment\_method, payment\_status, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="GDL.com.entity.Payment"> INSERT INTO payment <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != orderId and '' != orderId"> order\_id, </if> <if test="null != paymentMethod and '' != paymentMethod"> payment\_method, </if> <if test="null != paymentStatus and '' != paymentStatus"> payment\_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 != orderId and '' != orderId"> #{orderId}, </if> <if test="null != paymentMethod and '' != paymentMethod"> #{paymentMethod}, </if> <if test="null != paymentStatus and '' != paymentStatus"> #{paymentStatus}, </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 payment WHERE id = #{id} </delete> <update id="update" parameterType="GDL.com.entity.Payment"> UPDATE payment <set> <if test="null != orderId and '' != orderId">order\_id = #{orderId},</if> <if test="null != paymentMethod and '' != paymentMethod">payment\_method = #{paymentMethod},</if> <if test="null != paymentStatus and '' != paymentStatus">payment\_status = #{paymentStatus},</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 payment WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM payment LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM payment </select></mapper>/\*\* \* @description payment \* @author 大大的盆 \*/public class Payment implements Serializable { private static final long serialVersionUID = 1L; private String orderId; private String paymentMethod; private String paymentStatus; private String createTime; private String createUser; private String updateTime; private String updateUser; public Payment() { } public String getOrderId() { return orderId; } public void setOrderId(String orderId) { this.orderId = orderId; } public String getPaymentMethod() { return paymentMethod; } public void setPaymentMethod(String paymentMethod) { this.paymentMethod = paymentMethod; } public String getPaymentStatus() { return paymentStatus; } public void setPaymentStatus(String paymentStatus) { this.paymentStatus = paymentStatus; } 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:payment:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:payment: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="orderId" header-align="center" align="center" label="order\_id"> </el-table-column> <el-table-column prop="paymentMethod" header-align="center" align="center" label="payment\_method"> </el-table-column> <el-table-column prop="paymentStatus" header-align="center" align="center" label="payment\_status"> </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.paymentId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.paymentId)">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 './payment-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/payment/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.$refs.addOrUpdate.init(id) }) }, // 删除 deleteHandle (id) { var ids = id ? [id] : this.dataListSelections.map(item => { return item.paymentId }) this.$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/payment/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.paymentId ? '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="order\_id" prop="orderId"> <el-input v-model="dataForm.orderId" placeholder="order\_id"></el-input> </el-form-item> <el-form-item label="payment\_method" prop="paymentMethod"> <el-input v-model="dataForm.paymentMethod" placeholder="payment\_method"></el-input> </el-form-item> <el-form-item label="payment\_status" prop="paymentStatus"> <el-input v-model="dataForm.paymentStatus" placeholder="payment\_status"></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: { orderId: '', paymentMethod: '', paymentStatus: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { orderId: [{ required: true, message: 'order\_id不能为空', trigger: 'blur' }], paymentMethod: [{ required: true, message: 'payment\_method不能为空', trigger: 'blur' }], paymentStatus: [{ required: true, message: 'payment\_status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.paymentId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.paymentId) { this.$http({ url: this.$http.adornUrl(`/generator/payment/info/${this.dataForm.paymentId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.orderId = data.payment.orderId this.dataForm.paymentMethod = data.payment.paymentMethod this.dataForm.paymentStatus = data.payment.paymentStatus this.dataForm.createTime = data.payment.createTime this.dataForm.createUser = data.payment.createUser this.dataForm.updateTime = data.payment.updateTime this.dataForm.updateUser = data.payment.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/payment/${this.dataForm.paymentId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'orderId': 'orderId' || undefined, 'paymentMethod': 'paymentMethod' || undefined, 'paymentStatus': 'paymentStatus' || 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 review \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/review")public class ReviewController { @Resource private ReviewService reviewService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(Review review){ return reviewService.insert(review); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return reviewService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(Review review){ return reviewService.update(review); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return reviewService.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 reviewService.pageList(offset, pagesize); }}/\*\* \* @description review \* @author 大大的盆 \*/public interface ReviewService { /\*\* \* 新增 \*/ public Object insert(Review review); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(Review review); /\*\* \* 根据主键 id 查询 \*/ public Review load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description review \* @author 大大的盆 \*/@Servicepublic class ReviewServiceImpl implements ReviewService { @Resource private ReviewMapper reviewMapper; @Override public Object insert(Review review) { // valid if (review == null) { return ReturnT.error("必要参数缺失"); } reviewMapper.insert(review); return ReturnT.success(); } @Override public Object delete(int id) { int ret = reviewMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(Review review) { int ret = reviewMapper.update(review); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Review load(int id) { return reviewMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<Review> pageList = reviewMapper.pageList(offset, pagesize); int totalCount = reviewMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description review \* @author 大大的盆 \*/@Mapper@Repositorypublic interface ReviewMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(Review review); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(Review review); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ Review load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<Review> 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="GDL.com.dao.ReviewMapper"> <resultMap id="BaseResultMap" type="GDL.com.entity.Review" > <result column="order\_id" property="orderId" /> <result column="user\_id" property="userId" /> <result column="rating" property="rating" /> <result column="comments" property="comments" /> <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"> order\_id, user\_id, rating, comments, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="GDL.com.entity.Review"> INSERT INTO review <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != orderId and '' != orderId"> order\_id, </if> <if test="null != userId and '' != userId"> user\_id, </if> <if test="null != rating and '' != rating"> rating, </if> <if test="null != comments and '' != comments"> comments, </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 != orderId and '' != orderId"> #{orderId}, </if> <if test="null != userId and '' != userId"> #{userId}, </if> <if test="null != rating and '' != rating"> #{rating}, </if> <if test="null != comments and '' != comments"> #{comments}, </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 review WHERE id = #{id} </delete> <update id="update" parameterType="GDL.com.entity.Review"> UPDATE review <set> <if test="null != orderId and '' != orderId">order\_id = #{orderId},</if> <if test="null != userId and '' != userId">user\_id = #{userId},</if> <if test="null != rating and '' != rating">rating = #{rating},</if> <if test="null != comments and '' != comments">comments = #{comments},</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 review WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM review LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM review </select></mapper>/\*\* \* @description review \* @author 大大的盆 \*/public class Review implements Serializable { private static final long serialVersionUID = 1L; private String orderId; private String userId; private String rating; private String comments; private String createTime; private String createUser; private String updateTime; private String updateUser; public Review() { } public String getOrderId() { return orderId; } public void setOrderId(String orderId) { this.orderId = orderId; } public String getUserId() { return userId; } public void setUserId(String userId) { this.userId = userId; } public String getRating() { return rating; } public void setRating(String rating) { this.rating = rating; } public String getComments() { return comments; } public void setComments(String comments) { this.comments = comments; } 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:review:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:review: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="orderId" header-align="center" align="center" label="order\_id"> </el-table-column> <el-table-column prop="userId" header-align="center" align="center" label="user\_id"> </el-table-column> <el-table-column prop="rating" header-align="center" align="center" label="rating"> </el-table-column> <el-table-column prop="comments" header-align="center" align="center" label="comments"> </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.reviewId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.reviewId)">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 './review-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.$http.adornUrl('/generator/review/list'), method: 'get', params: this.$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.reviewId }) 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.$http.adornUrl('/generator/review/delete'), method: 'post', data: this.$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.reviewId ? '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="order\_id" prop="orderId"> <el-input v-model="dataForm.orderId" placeholder="order\_id"></el-input> </el-form-item> <el-form-item label="user\_id" prop="userId"> <el-input v-model="dataForm.userId" placeholder="user\_id"></el-input> </el-form-item> <el-form-item label="rating" prop="rating"> <el-input v-model="dataForm.rating" placeholder="rating"></el-input> </el-form-item> <el-form-item label="comments" prop="comments"> <el-input v-model="dataForm.comments" placeholder="comments"></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: { orderId: '', userId: '', rating: '', comments: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { orderId: [{ required: true, message: 'order\_id不能为空', trigger: 'blur' }], userId: [{ required: true, message: 'user\_id不能为空', trigger: 'blur' }], rating: [{ required: true, message: 'rating不能为空', trigger: 'blur' }], comments: [{ required: true, message: 'comments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.reviewId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.reviewId) { this.$http({ url: this.$http.adornUrl(`/generator/review/info/${this.dataForm.reviewId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.orderId = data.review.orderId this.dataForm.userId = data.review.userId this.dataForm.rating = data.review.rating this.dataForm.comments = data.review.comments this.dataForm.createTime = data.review.createTime this.dataForm.createUser = data.review.createUser this.dataForm.updateTime = data.review.updateTime this.dataForm.updateUser = data.review.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/review/${this.dataForm.reviewId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'orderId': 'orderId' || undefined, 'userId': 'userId' || undefined, 'rating': 'rating' || undefined, 'comments': 'comments' || undefined, 'createTime': 'createTime' || undefined, 'createUser': 'createUser' || undefined, 'updateTime': 'updateTime' || undefined, 'updateUser': 'updateUser' || undefined }) }).then(({data}) => { if (data && data.code === 0) { this.$message({ message: '操作成功', type: 'success', duration: 1500, onClose: () => { this.visible = false this.$emit('refreshDataList') } }) } else { this.$message.error(data.msg) } }) } }) } } }</script>