<!DOCTYPE html><html lang="zh-CN"><head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <title>易晞茶叶销售平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>易晞茶叶销售平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item active"><span>商品展示</span></li></a> <a ><li class="rz-nav-item"><span>购物车</span></li></a> <a ><li class="rz-nav-item"><span>支付界面</span></li></a> <a ><li class="rz-nav-item"><span>订单管理</span></li></a> <a ><li class="rz-nav-item"><span>评论与评价</span></li></a> <a ><li class="rz-nav-item"><span>积分系统</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>茶叶产品展示</h2> <div class="product-card"> <img src="https://example.com/tea1.jpg" alt="茶叶1" class="product-image"> <div class="product-title">绿茶</div> <div class="product-price">价格: ¥50</div> <div class="product-rating">评分: ★★★★☆</div> <div class="product-description">优质绿茶，清香四溢，回甘悠长。</div> <button class="view-detail">查看详情</button> </div> <div class="product-card"> <img src="https://example.com/tea2.jpg" alt="茶叶2" class="product-image"> <div class="product-title">红茶</div> <div class="product-price">价格: ¥60</div> <div class="product-rating">评分: ★★★★★</div> <div class="product-description">醇厚红茶，适合搭配早餐。</div> <button class="view-detail">查看详情</button> </div> <div class="product-card"> <img src="https://example.com/tea3.jpg" alt="茶叶3" class="product-image"> <div class="product-title">乌龙茶</div> <div class="product-price">价格: ¥70</div> <div class="product-rating">评分: ★★★★☆</div> <div class="product-description">独特的香气，回味无穷。</div> <button class="view-detail">查看详情</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>易晞茶叶销售平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>易晞茶叶销售平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>商品展示</span></li></a> <a ><li class="rz-nav-item active"><span>购物车</span></li></a> <a ><li class="rz-nav-item"><span>支付界面</span></li></a> <a ><li class="rz-nav-item"><span>订单管理</span></li></a> <a ><li class="rz-nav-item"><span>评论与评价</span></li></a> <a ><li class="rz-nav-item"><span>积分系统</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>购物车</h2> <div class="cart-item"> <span>茶叶名称</span> <span>数量</span> <span>单价</span> <span>操作</span> </div> <div class="cart-item"> <span>绿茶</span> <input type="number" value="1" min="1"> <span>¥30</span> <button>删除</button> </div> <div class="cart-item"> <span>红茶</span> <input type="number" value="2" min="1"> <span>¥40</span> <button>删除</button> </div> <div class="total"> 总金额: ¥110 </div> <div class="coupon-input"> <input type="text" placeholder="输入优惠券代码"> <button>应用</button> </div> <button class="checkout-button">结算</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>易晞茶叶销售平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>易晞茶叶销售平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>商品展示</span></li></a> <a ><li class="rz-nav-item"><span>购物车</span></li></a> <a ><li class="rz-nav-item active"><span>支付界面</span></li></a> <a ><li class="rz-nav-item"><span>订单管理</span></li></a> <a ><li class="rz-nav-item"><span>评论与评价</span></li></a> <a ><li class="rz-nav-item"><span>积分系统</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>确认支付信息</h2> <div class="order-summary"> <h3>订单信息</h3> <p>商品清单:</p> <ul> <li>商品A - ￥100</li> <li>商品B - ￥200</li> </ul> <p><strong>总金额: ￥300</strong></p> <p><strong>运费: ￥10</strong></p> <p><strong>应付金额: ￥310</strong></p> </div> <div class="payment-method"> <h3>选择支付方式</h3> <select> <option value="credit\_card">信用卡</option> <option value="alipay">支付宝</option> <option value="wechat">微信支付</option> </select> </div> <div class="payment-info"> <h3>输入支付信息</h3> <input type="text" placeholder="卡号/支付宝账号/微信账号" required /> <input type="password" placeholder="支付密码" required /> </div> <button class="payment-button">支付</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>易晞茶叶销售平台</title> <link rel="stylesheet"></head><body> <header class="rz-header"> <h1>易晞茶叶销售平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>商品展示</span></li></a> <a ><li class="rz-nav-item"><span>购物车</span></li></a> <a ><li class="rz-nav-item"><span>支付界面</span></li></a> <a ><li class="rz-nav-item active"><span>订单管理</span></li></a> <a ><li class="rz-nav-item"><span>评论与评价</span></li></a> <a ><li class="rz-nav-item"><span>积分系统</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>订单管理</h2> <div class="order-item"> <div> <strong>订单编号:</strong> 123456 <strong>状态:</strong> 待付款 <button class="button" onclick="this.parentElement.parentElement.classList.toggle('expanded')">查看详情</button> </div> <div class="order-details"> <p><strong>商品:</strong> 茶叶A</p> <p><strong>数量:</strong> 2</p> <p><strong>总价:</strong> ¥200</p> <button class="button">申请退款</button> <button class="button">投诉</button> </div> </div> <div class="order-item"> <div> <strong>订单编号:</strong> 654321 <strong>状态:</strong> 已完成 <button class="button" onclick="this.parentElement.parentElement.classList.toggle('expanded')">查看详情</button> </div> <div class="order-details"> <p><strong>商品:</strong> 茶叶B</p> <p><strong>数量:</strong> 1</p> <p><strong>总价:</strong> ¥100</p> <button class="button">申请退款</button> <button class="button">投诉</button> </div> </div> <!-- 可以继续添加更多订单 --> </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>易晞茶叶销售平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>易晞茶叶销售平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>商品展示</span></li></a> <a ><li class="rz-nav-item"><span>购物车</span></li></a> <a ><li class="rz-nav-item"><span>支付界面</span></li></a> <a ><li class="rz-nav-item"><span>订单管理</span></li></a> <a ><li class="rz-nav-item active"><span>评论与评价</span></li></a> <a ><li class="rz-nav-item"><span>积分系统</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>对已购买茶叶的评价</h2> <div class="review-form"> <label for="rating">星级评分:</label> <select id="rating"> <option value="5">★★★★★</option> <option value="4">★★★★☆</option> <option value="3">★★★☆☆</option> <option value="2">★★☆☆☆</option> <option value="1">★☆☆☆☆</option> </select> <label for="comment">文字评论:</label> <textarea id="comment" placeholder="分享您的使用体验和感受..."></textarea> <button type="button">提交评价</button> </div> <div class="review-list"> <h3>用户评价</h3> <div class="review-item"> <div class="stars">★★★★★</div> <p>这款茶叶的口感非常好，值得推荐！</p> </div> <div class="review-item"> <div class="stars">★★★★☆</div> <p>整体不错，但希望包装能更好一些。</p> </div> <!-- 可以在这里添加更多评论项 --> </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>易晞茶叶销售平台</title> <link rel="stylesheet"> <link rel="stylesheet" ></head><body> <header class="rz-header"> <h1>易晞茶叶销售平台</h1> </header> <div class="rz-container"> <nav class="rz-sidebar"> <ul class="rz-nav-list"> <a ><li class="rz-nav-item"><span>商品展示</span></li></a> <a ><li class="rz-nav-item"><span>购物车</span></li></a> <a ><li class="rz-nav-item"><span>支付界面</span></li></a> <a ><li class="rz-nav-item"><span>订单管理</span></li></a> <a ><li class="rz-nav-item"><span>评论与评价</span></li></a> <a ><li class="rz-nav-item active"><span>积分系统</span></li></a> </ul> </nav> <main class="rz-main-content"> <div id="content" class="rz-card"> <h2>积分系统</h2> <p>当前积分余额: <strong>200</strong> 分</p> <h3>积分记录</h3> <div class="record-list"> <div class="record-item">2023-10-01 购物获得积分: <strong>100</strong> 分</div> <div class="record-item">2023-10-05 购物使用积分: <strong>-50</strong> 分</div> <div class="record-item">2023-10-10 购物获得积分: <strong>150</strong> 分</div> </div> <h3>积分活动</h3> <div class="activity"> <p>参与积分活动，赢取额外积分！活动时间：2023年10月1日至2023年10月31日。</p> </div> </div> </main> </div></body></html>

/\*\* \* @description products \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/products")public class ProductsController { @Resource private ProductsService productsService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(Products products){ return productsService.insert(products); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return productsService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(Products products){ return productsService.update(products); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return productsService.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 productsService.pageList(offset, pagesize); }}/\*\* \* @description products \* @author 大大的盆 \*/public interface ProductsService { /\*\* \* 新增 \*/ public Object insert(Products products); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(Products products); /\*\* \* 根据主键 id 查询 \*/ public Products load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description products \* @author 大大的盆 \*/@Servicepublic class ProductsServiceImpl implements ProductsService { @Resource private ProductsMapper productsMapper; @Override public Object insert(Products products) { // valid if (products == null) { return ReturnT.error("必要参数缺失"); } productsMapper.insert(products); return ReturnT.success(); } @Override public Object delete(int id) { int ret = productsMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(Products products) { int ret = productsMapper.update(products); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Products load(int id) { return productsMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<Products> pageList = productsMapper.pageList(offset, pagesize); int totalCount = productsMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description products \* @author 大大的盆 \*/@Mapper@Repositorypublic interface ProductsMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(Products products); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(Products products); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ Products load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<Products> 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.ProductsMapper"> <resultMap id="BaseResultMap" type="HBJZGCXY.com.entity.Products" > <result column="name" property="name" /> <result column="description" property="description" /> <result column="price" property="price" /> <result column="stock" property="stock" /> <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"> name, description, price, stock, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="HBJZGCXY.com.entity.Products"> INSERT INTO products <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != name and '' != name"> name, </if> <if test="null != description and '' != description"> description, </if> <if test="null != price and '' != price"> price, </if> <if test="null != stock and '' != stock"> stock, </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 != name and '' != name"> #{name}, </if> <if test="null != description and '' != description"> #{description}, </if> <if test="null != price and '' != price"> #{price}, </if> <if test="null != stock and '' != stock"> #{stock}, </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 products WHERE id = #{id} </delete> <update id="update" parameterType="HBJZGCXY.com.entity.Products"> UPDATE products <set> <if test="null != name and '' != name">name = #{name},</if> <if test="null != description and '' != description">description = #{description},</if> <if test="null != price and '' != price">price = #{price},</if> <if test="null != stock and '' != stock">stock = #{stock},</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 products WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM products LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM products </select></mapper>/\*\* \* @description products \* @author 大大的盆 \*/public class Products implements Serializable { private static final long serialVersionUID = 1L; private String name; private String description; private String price; private String stock; private String createTime; private String createUser; private String updateTime; private String updateUser; public Products() { } public String getName() { return name; } public void setName(String name) { this.name = name; } public String getDescription() { return description; } public void setDescription(String description) { this.description = description; } public String getPrice() { return price; } public void setPrice(String price) { this.price = price; } public String getStock() { return stock; } public void setStock(String stock) { this.stock = stock; } 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:products:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:products: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="name" header-align="center" align="center" label="name"> </el-table-column> <el-table-column prop="description" header-align="center" align="center" label="description"> </el-table-column> <el-table-column prop="price" header-align="center" align="center" label="price"> </el-table-column> <el-table-column prop="stock" header-align="center" align="center" label="stock"> </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.productsId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.productsId)">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 './products-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/products/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.$refs.addOrUpdate.init(id) }) }, // 删除 deleteHandle (id) { var ids = id ? [id] : this.dataListSelections.map(item => { return item.productsId }) this.$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/products/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.productsId ? '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="name" prop="name"> <el-input v-model="dataForm.name" placeholder="name"></el-input> </el-form-item> <el-form-item label="description" prop="description"> <el-input v-model="dataForm.description" placeholder="description"></el-input> </el-form-item> <el-form-item label="price" prop="price"> <el-input v-model="dataForm.price" placeholder="price"></el-input> </el-form-item> <el-form-item label="stock" prop="stock"> <el-input v-model="dataForm.stock" placeholder="stock"></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: { name: '', description: '', price: '', stock: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { name: [{ required: true, message: 'name不能为空', trigger: 'blur' }], description: [{ required: true, message: 'description不能为空', trigger: 'blur' }], price: [{ required: true, message: 'price不能为空', trigger: 'blur' }], stock: [{ required: true, message: 'stock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.productsId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.productsId) { this.$http({ url: this.$http.adornUrl(`/generator/products/info/${this.dataForm.productsId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.name = data.products.name this.dataForm.description = data.products.description this.dataForm.price = data.products.price this.dataForm.stock = data.products.stock this.dataForm.createTime = data.products.createTime this.dataForm.createUser = data.products.createUser this.dataForm.updateTime = data.products.updateTime this.dataForm.updateUser = data.products.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/products/${this.dataForm.productsId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'name': 'name' || undefined, 'description': 'description' || undefined, 'price': 'price' || undefined, 'stock': 'stock' || 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 users \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/users")public class UsersController { @Resource private UsersService usersService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(Users users){ return usersService.insert(users); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return usersService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(Users users){ return usersService.update(users); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return usersService.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 usersService.pageList(offset, pagesize); }}/\*\* \* @description users \* @author 大大的盆 \*/public interface UsersService { /\*\* \* 新增 \*/ public Object insert(Users users); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(Users users); /\*\* \* 根据主键 id 查询 \*/ public Users load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description users \* @author 大大的盆 \*/@Servicepublic class UsersServiceImpl implements UsersService { @Resource private UsersMapper usersMapper; @Override public Object insert(Users users) { // valid if (users == null) { return ReturnT.error("必要参数缺失"); } usersMapper.insert(users); return ReturnT.success(); } @Override public Object delete(int id) { int ret = usersMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(Users users) { int ret = usersMapper.update(users); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Users load(int id) { return usersMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<Users> pageList = usersMapper.pageList(offset, pagesize); int totalCount = usersMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description users \* @author 大大的盆 \*/@Mapper@Repositorypublic interface UsersMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(Users users); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(Users users); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ Users load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<Users> 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.UsersMapper"> <resultMap id="BaseResultMap" type="HBJZGCXY.com.entity.Users" > <result column="username" property="username" /> <result column="password" property="password" /> <result column="email" property="email" /> <result column="points" property="points" /> <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, points, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="HBJZGCXY.com.entity.Users"> INSERT INTO users <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 != points and '' != points"> points, </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 != points and '' != points"> #{points}, </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 users WHERE id = #{id} </delete> <update id="update" parameterType="HBJZGCXY.com.entity.Users"> UPDATE users <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 != points and '' != points">points = #{points},</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 users WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM users LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM users </select></mapper>/\*\* \* @description users \* @author 大大的盆 \*/public class Users implements Serializable { private static final long serialVersionUID = 1L; private String username; private String password; private String email; private String points; private String createTime; private String createUser; private String updateTime; private String updateUser; public Users() { } 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 getPoints() { return points; } public void setPoints(String points) { this.points = points; } 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:users:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:users: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="points" header-align="center" align="center" label="points"> </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.usersId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.usersId)">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 './users-add-or-update' export default { data () { return { dataForm: { key: '' }, dataList: [], pageIndex: 1, pageSize: 10, totalPage: 0, dataListLoading: false, dataListSelections: [], addOrUpdateVisible: false } }, components: { AddOrUpdate },

}

}<el-form-item label="order\_date" prop="orderDate"> <el-input v-model="dataForm.orderDate" placeholder="order\_date"></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="status" prop="status"> <el-input v-model="dataForm.status" placeholder="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: '', orderDate: '', totalAmount: '', status: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { userId: [{ required: true, message: 'user\_id不能为空', trigger: 'blur' }], orderDate: [{ required: true, message: 'order\_date不能为空', trigger: 'blur' }], totalAmount: [{ required: true, message: 'total\_amount不能为空', trigger: 'blur' }], status: [{ required: true, message: '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.orderDate = data.orders.orderDate this.dataForm.totalAmount = data.orders.totalAmount this.dataForm.status = data.orders.status 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.$http.adornData({ 'userId': 'userId' || undefined, 'orderDate': 'orderDate' || undefined, 'totalAmount': 'totalAmount' || undefined, 'status': 'status' || undefined, 'createTime': 'createTime' || undefined, 'createUser': 'createUser' || undefined, 'updateTime': 'updateTime' || undefined, 'updateUser': 'updateUser' || undefined }) }).then(({data}) => { if (data && data.code === 0) { this.$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 shopping\_cart \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/shoppingCart")public class ShoppingCartController { @Resource private ShoppingCartService shoppingCartService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(ShoppingCart shoppingCart){ return shoppingCartService.insert(shoppingCart); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return shoppingCartService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(ShoppingCart shoppingCart){ return shoppingCartService.update(shoppingCart); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return shoppingCartService.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 shoppingCartService.pageList(offset, pagesize); }}/\*\* \* @description shopping\_cart \* @author 大大的盆 \*/public interface ShoppingCartService { /\*\* \* 新增 \*/ public Object insert(ShoppingCart shoppingCart); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(ShoppingCart shoppingCart); /\*\* \* 根据主键 id 查询 \*/ public ShoppingCart load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description shopping\_cart \* @author 大大的盆 \*/@Servicepublic class ShoppingCartServiceImpl implements ShoppingCartService { @Resource private ShoppingCartMapper shoppingCartMapper; @Override public Object insert(ShoppingCart shoppingCart) { // valid if (shoppingCart == null) { return ReturnT.error("必要参数缺失"); } shoppingCartMapper.insert(shoppingCart); return ReturnT.success(); } @Override public Object delete(int id) { int ret = shoppingCartMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(ShoppingCart shoppingCart) { int ret = shoppingCartMapper.update(shoppingCart); return ret>0?ReturnT.success():ReturnT.error(); } @Override public ShoppingCart load(int id) { return shoppingCartMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<ShoppingCart> pageList = shoppingCartMapper.pageList(offset, pagesize); int totalCount = shoppingCartMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description shopping\_cart \* @author 大大的盆 \*/@Mapper@Repositorypublic interface ShoppingCartMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(ShoppingCart shoppingCart); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(ShoppingCart shoppingCart); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ ShoppingCart load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<ShoppingCart> 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.ShoppingCartMapper"> <resultMap id="BaseResultMap" type="HBJZGCXY.com.entity.ShoppingCart" > <result column="user\_id" property="userId" /> <result column="product\_id" property="productId" /> <result column="quantity" property="quantity" /> <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, product\_id, quantity, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="HBJZGCXY.com.entity.ShoppingCart"> INSERT INTO shopping\_cart <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != userId and '' != userId"> user\_id, </if> <if test="null != productId and '' != productId"> product\_id, </if> <if test="null != quantity and '' != quantity"> quantity, </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 != productId and '' != productId"> #{productId}, </if> <if test="null != quantity and '' != quantity"> #{quantity}, </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 shopping\_cart WHERE id = #{id} </delete> <update id="update" parameterType="HBJZGCXY.com.entity.ShoppingCart"> UPDATE shopping\_cart <set> <if test="null != userId and '' != userId">user\_id = #{userId},</if> <if test="null != productId and '' != productId">product\_id = #{productId},</if> <if test="null != quantity and '' != quantity">quantity = #{quantity},</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 shopping\_cart WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM shopping\_cart LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM shopping\_cart </select></mapper>/\*\* \* @description shopping\_cart \* @author 大大的盆 \*/public class ShoppingCart implements Serializable { private static final long serialVersionUID = 1L; private String userId; private String productId; private String quantity; private String createTime; private String createUser; private String updateTime; private String updateUser; public ShoppingCart() { } public String getUserId() { return userId; } public void setUserId(String userId) { this.userId = userId; } public String getProductId() { return productId; } public void setProductId(String productId) { this.productId = productId; } public String getQuantity() { return quantity; } public void setQuantity(String quantity) { this.quantity = quantity; } 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:shoppingCart:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:shoppingCart: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="productId" header-align="center" align="center" label="product\_id"> </el-table-column> <el-table-column prop="quantity" header-align="center" align="center" label="quantity"> </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.shoppingCartId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.shoppingCartId)">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 './shoppingCart-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/shoppingCart/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.$refs.addOrUpdate.init(id) }) }, // 删除 deleteHandle (id) { var ids = id ? [id] : this.dataListSelections.map(item => { return item.shoppingCartId }) this.$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/shoppingCart/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.shoppingCartId ? '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="user\_id" prop="userId"> <el-input v-model="dataForm.userId" placeholder="user\_id"></el-input> </el-form-item> <el-form-item label="product\_id" prop="productId"> <el-input v-model="dataForm.productId" placeholder="product\_id"></el-input> </el-form-item> <el-form-item label="quantity" prop="quantity"> <el-input v-model="dataForm.quantity" placeholder="quantity"></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: '', productId: '', quantity: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { userId: [{ required: true, message: 'user\_id不能为空', trigger: 'blur' }], productId: [{ required: true, message: 'product\_id不能为空', trigger: 'blur' }], quantity: [{ required: true, message: 'quantity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.shoppingCartId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.shoppingCartId) { this.$http({ url: this.$http.adornUrl(`/generator/shoppingCart/info/${this.dataForm.shoppingCartId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.userId = data.shoppingCart.userId this.dataForm.productId = data.shoppingCart.productId this.dataForm.quantity = data.shoppingCart.quantity this.dataForm.createTime = data.shoppingCart.createTime this.dataForm.createUser = data.shoppingCart.createUser this.dataForm.updateTime = data.shoppingCart.updateTime this.dataForm.updateUser = data.shoppingCart.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/shoppingCart/${this.dataForm.shoppingCartId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'userId': 'userId' || undefined, 'productId': 'productId' || undefined, 'quantity': 'quantity' || 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 point\_transactions \* @author 大大的盆 \*/@RestController@RequestMapping(value = "/pointTransactions")public class PointTransactionsController { @Resource private PointTransactionsService pointTransactionsService; /\*\* \* 新增 \* @author 大大的盆 \*\*/ @RequestMapping("/insert") public Object insert(PointTransactions pointTransactions){ return pointTransactionsService.insert(pointTransactions); } /\*\* \* 刪除 \* @author 大大的盆 \*\*/ @RequestMapping("/delete") public Object delete(int id){ return pointTransactionsService.delete(id); } /\*\* \* 更新 \* @author 大大的盆 \*\*/ @RequestMapping("/update") public Object update(PointTransactions pointTransactions){ return pointTransactionsService.update(pointTransactions); } /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ @RequestMapping("/load") public Object load(int id){ return pointTransactionsService.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 pointTransactionsService.pageList(offset, pagesize); }}/\*\* \* @description point\_transactions \* @author 大大的盆 \*/public interface PointTransactionsService { /\*\* \* 新增 \*/ public Object insert(PointTransactions pointTransactions); /\*\* \* 删除 \*/ public Object delete(int id); /\*\* \* 更新 \*/ public Object update(PointTransactions pointTransactions); /\*\* \* 根据主键 id 查询 \*/ public PointTransactions load(int id); /\*\* \* 分页查询 \*/ public Map<String,Object> pageList(int offset, int pagesize);}/\*\* \* @description point\_transactions \* @author 大大的盆 \*/@Servicepublic class PointTransactionsServiceImpl implements PointTransactionsService { @Resource private PointTransactionsMapper pointTransactionsMapper; @Override public Object insert(PointTransactions pointTransactions) { // valid if (pointTransactions == null) { return ReturnT.error("必要参数缺失"); } pointTransactionsMapper.insert(pointTransactions); return ReturnT.success(); } @Override public Object delete(int id) { int ret = pointTransactionsMapper.delete(id); return ret>0?ReturnT.success():ReturnT.error(); } @Override public Object update(PointTransactions pointTransactions) { int ret = pointTransactionsMapper.update(pointTransactions); return ret>0?ReturnT.success():ReturnT.error(); } @Override public PointTransactions load(int id) { return pointTransactionsMapper.load(id); } @Override public Map<String,Object> pageList(int offset, int pagesize) { List<PointTransactions> pageList = pointTransactionsMapper.pageList(offset, pagesize); int totalCount = pointTransactionsMapper.pageListCount(offset, pagesize); // result Map<String, Object> result = new HashMap<String, Object>(); result.put("pageList", pageList); result.put("totalCount", totalCount); return result; }}/\*\* \* @description point\_transactions \* @author 大大的盆 \*/@Mapper@Repositorypublic interface PointTransactionsMapper { /\*\* \* 新增 \* @author 大大的盆 \*\*/ int insert(PointTransactions pointTransactions); /\*\* \* 刪除 \* @author 大大的盆 \*\*/ int delete(int id); /\*\* \* 更新 \* @author 大大的盆 \*\*/ int update(PointTransactions pointTransactions); /\*\* \* 查询 根据主键 id 查询 \* @author 大大的盆 \*\*/ PointTransactions load(int id); /\*\* \* 查询 分页查询 \* @author 大大的盆 \*\*/ List<PointTransactions> 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.PointTransactionsMapper"> <resultMap id="BaseResultMap" type="HBJZGCXY.com.entity.PointTransactions" > <result column="user\_id" property="userId" /> <result column="points" property="points" /> <result column="transaction\_type" property="transactionType" /> <result column="transaction\_date" property="transactionDate" /> <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, points, transaction\_type, transaction\_date, create\_time, create\_user, update\_time, update\_user </sql> <insert id="insert" useGeneratedKeys="true" keyColumn="id" keyProperty="id" parameterType="HBJZGCXY.com.entity.PointTransactions"> INSERT INTO point\_transactions <trim prefix="(" suffix=")" suffixOverrides=","> <if test="null != userId and '' != userId"> user\_id, </if> <if test="null != points and '' != points"> points, </if> <if test="null != transactionType and '' != transactionType"> transaction\_type, </if> <if test="null != transactionDate and '' != transactionDate"> transaction\_date, </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 != points and '' != points"> #{points}, </if> <if test="null != transactionType and '' != transactionType"> #{transactionType}, </if> <if test="null != transactionDate and '' != transactionDate"> #{transactionDate}, </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 point\_transactions WHERE id = #{id} </delete> <update id="update" parameterType="HBJZGCXY.com.entity.PointTransactions"> UPDATE point\_transactions <set> <if test="null != userId and '' != userId">user\_id = #{userId},</if> <if test="null != points and '' != points">points = #{points},</if> <if test="null != transactionType and '' != transactionType">transaction\_type = #{transactionType},</if> <if test="null != transactionDate and '' != transactionDate">transaction\_date = #{transactionDate},</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 point\_transactions WHERE id = #{id} </select> <select id="pageList" resultMap="BaseResultMap"> SELECT <include refid="Base\_Column\_List" /> FROM point\_transactions LIMIT #{offset}, #{pageSize} </select> <select id="pageListCount" resultType="java.lang.Integer"> SELECT count(1) FROM point\_transactions </select></mapper>/\*\* \* @description point\_transactions \* @author 大大的盆 \*/public class PointTransactions implements Serializable { private static final long serialVersionUID = 1L; private String userId; private String points; private String transactionType; private String transactionDate; private String createTime; private String createUser; private String updateTime; private String updateUser; public PointTransactions() { } public String getUserId() { return userId; } public void setUserId(String userId) { this.userId = userId; } public String getPoints() { return points; } public void setPoints(String points) { this.points = points; } public String getTransactionType() { return transactionType; } public void setTransactionType(String transactionType) { this.transactionType = transactionType; } public String getTransactionDate() { return transactionDate; } public void setTransactionDate(String transactionDate) { this.transactionDate = transactionDate; } 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:pointTransactions:save')" type="primary" @click="addOrUpdateHandle()">新增</el-button> <el-button v-if="isAuth('generator:pointTransactions: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="points" header-align="center" align="center" label="points"> </el-table-column> <el-table-column prop="transactionType" header-align="center" align="center" label="transaction\_type"> </el-table-column> <el-table-column prop="transactionDate" header-align="center" align="center" label="transaction\_date"> </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.pointTransactionsId)">修改</el-button> <el-button type="text" size="small" @click="deleteHandle(scope.row.pointTransactionsId)">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 './pointTransactions-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.$http.adornUrl('/generator/pointTransactions/list'), method: 'get', params: this.$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.pointTransactionsId }) 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.$http.adornUrl('/generator/pointTransactions/delete'), method: 'post', data: this.$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.pointTransactionsId ? '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="user\_id" prop="userId"> <el-input v-model="dataForm.userId" placeholder="user\_id"></el-input> </el-form-item> <el-form-item label="points" prop="points"> <el-input v-model="dataForm.points" placeholder="points"></el-input> </el-form-item> <el-form-item label="transaction\_type" prop="transactionType"> <el-input v-model="dataForm.transactionType" placeholder="transaction\_type"></el-input> </el-form-item> <el-form-item label="transaction\_date" prop="transactionDate"> <el-input v-model="dataForm.transactionDate" placeholder="transaction\_date"></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: '', points: '', transactionType: '', transactionDate: '', createTime: '', createUser: '', updateTime: '', updateUser: '' }, dataRule: { userId: [{ required: true, message: 'user\_id不能为空', trigger: 'blur' }], points: [{ required: true, message: 'points不能为空', trigger: 'blur' }], transactionType: [{ required: true, message: 'transaction\_type不能为空', trigger: 'blur' }], transactionDate: [{ required: true, message: 'transaction\_date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.pointTransactionsId = id || 0 this.visible = true this.$nextTick(() => { this.$refs['dataForm'].resetFields() if (this.dataForm.pointTransactionsId) { this.$http({ url: this.$http.adornUrl(`/generator/pointTransactions/info/${this.dataForm.pointTransactionsId}`), method: 'get', params: this.$http.adornParams() }).then(({data}) => { if (data && data.code === 0) { this.dataForm.userId = data.pointTransactions.userId this.dataForm.points = data.pointTransactions.points this.dataForm.transactionType = data.pointTransactions.transactionType this.dataForm.transactionDate = data.pointTransactions.transactionDate this.dataForm.createTime = data.pointTransactions.createTime this.dataForm.createUser = data.pointTransactions.createUser this.dataForm.updateTime = data.pointTransactions.updateTime this.dataForm.updateUser = data.pointTransactions.updateUser } }) } }) }, // 表单提交 dataFormSubmit () { this.$refs['dataForm'].validate((valid) => { if (valid) { this.$http({ url: this.$http.adornUrl(`/generator/pointTransactions/${this.dataForm.pointTransactionsId? 'save' : 'update'}`), method: 'post', data: this.$http.adornData({ 'userId': 'userId' || undefined, 'points': 'points' || undefined, 'transactionType': 'transactionType' || undefined, 'transactionDate': 'transactionDate' || undefined, 'createTime': 'createTime' || undefined, 'createUser': 'createUser' || undefined, 'updateTime': 'updateTime' || undefined, 'updateUser': 'updateUser' || undefined }) }).then(({data}) => { if (data && data.code === 0) { this.$message({ message: '操作成功', type: 'success', duration: 1500, onClose: () => { this.visible = false this.$emit('refreshDataList') } }) } else { this.$message.error(data.msg) } }) } }) } } }</script>