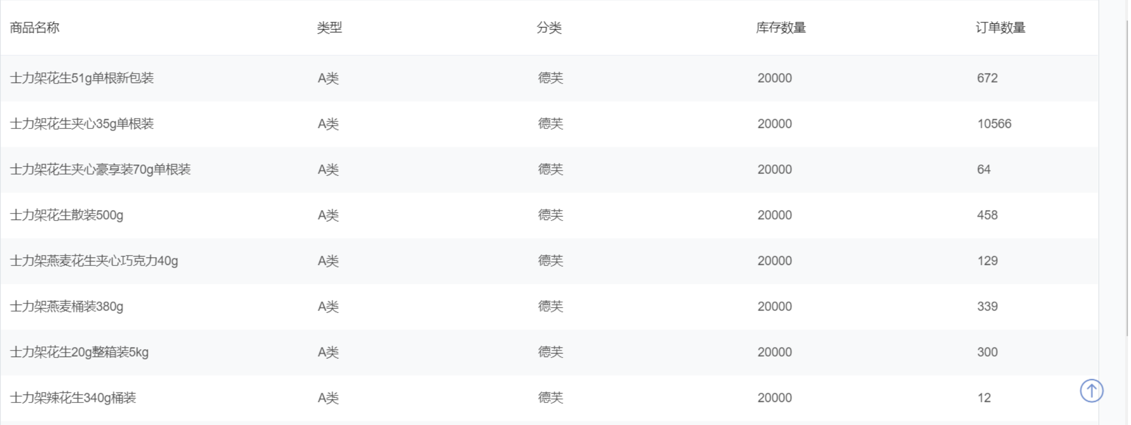
商品管理系统

功能描述

- 商品列表展示

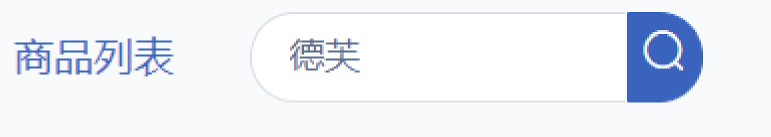
  商品显示信息分别有商品名称、类型、分类、库存数量、订单数量



商品名称	类型	分类	库存数量	订单数量
士力架花生51g单根新包装	A类	德芙	20000	672
士力架花生夹心35g单根装	A类	德芙	20000	10566
士力架花生夹心豪享装70g单根装	A类	德芙	20000	64
士力架花生散装500g	A类	德芙	20000	458
士力架燕麦花生夹心巧克力40g	A类	德芙	20000	129
士力架燕麦桶装380g	A类	德芙	20000	339
士力架花生20g整箱装5kg	A类	德芙	20000	300
士力架辣花生340g桶装	A类	德芙	20000	12

- 商品搜索功能

  可以根据商品名称、类型、分类进行模糊搜索



核心代码实现

整体采用Springboot + mybaties + mysql + vue实现

数据库模型

数据库采用mysql数据库，简历商品单品表

```sql
CREATE TABLE `hjc_sku` (
  `id` int NOT NULL AUTO_INCREMENT,
  `title` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
  `description` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci DEFAULT '',
  `category` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
  `order_num` int NOT NULL,
  `stock` int DEFAULT NULL,
  `type` varchar(255) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL DEFAULT '',
  `create_time` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `update_time` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `delete_time` datetime DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=232 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

id： 商品id

title：商品名称

description：商品描述信息

category：商品分类

order_num：订单数量

stock：库存量

type：商品类型

数据库映射的Model对象

```java
/**
 * @作者 7七月
 * @微信公号 林间有风
 * @开源项目 $ http://talelin.com
 * @免费专栏 $ http://course.talelin.com
 * @我的课程 $ http://imooc.com/t/4294850
 * @创建时间 2020-05-14 05:20
 */
package io.github.talelin.latticy.model;

import com.baomidou.mybatisplus.annotation.IdType;
import com.baomidou.mybatisplus.annotation.TableId;
import com.baomidou.mybatisplus.annotation.TableLogic;
import com.baomidou.mybatisplus.annotation.TableName;
import com.fasterxml.jackson.annotation.JsonIgnore;
import lombok.Getter;
import lombok.Setter;

import java.util.Date;

/**
 * @author TaleLin
 */
@Getter
@Setter
@TableName("hjc_sku")
public class HjcSkuDO {

    @TableId(value = "id", type = IdType.AUTO)
    private Integer id;

    private String title;

    private String description;

    private String category;

    private Integer orderNum;

    private Integer stock;

    private String type;

    @JsonIgnore
    private Date createTime;

    @JsonIgnore
    private Date updateTime;

    @JsonIgnore
    @TableLogic
    private Date deleteTime;
}
```

mapper数据访问层

- 获取所有商品信息

```
<select id="getAllSkuByKeywords" resultMap="BaseResultMap">
        SELECT * FROM hjc_sku
        <where>
            <if test="keywords != null and keywords != ''">and (
                instr(hjc_sku.type, #{keywords}) +
                instr(hjc_sku.title, #{keywords}) +
                instr(hjc_sku.category, #{keywords})
                ) > 0</if>
        </where>
</select>
```

- 根据关键字进行商品搜索

```
<select id="getAllSkuByKeywords" resultMap="BaseResultMap">
        SELECT * FROM hjc_sku
        <where>
            <if test="keywords != null and keywords != ''">and (
                instr(hjc_sku.type, #{keywords}) +
                instr(hjc_sku.title, #{keywords}) +
                instr(hjc_sku.category, #{keywords})
                ) > 0</if>
        </where>
</select>
```

Service业务层

```
@Service
public class HjcSkuService {

    @Autowired
    private HjcSkuMapper hjcSkuMapper;

    /**
    获取所有商品信息
    */
    public Page<HjcSkuDO> getAllSku(Integer page, Integer count){
        Page<LogDO> pager = new Page<>(page, count);
        return hjcSkuMapper.getAllSku(pager);
    }

    /**
    根据关键字进行商品搜索
    */
    public Page<HjcSkuDO> getAllSkuByKeywords(Integer page, Integer count, String keywords){
        Page<LogDO> pager = new Page<>(page, count);
        return hjcSkuMapper.getAllSkuByKeywords(pager,keywords);
    }

}
```

Controller控制层

```
@RequestMapping("cms/hjcSku")
@RestController
public class HjcSkuController {

    @Autowired
    private HjcSkuService hjcSkuService;


    /**
    获取所有商品信息
    */
    @GetMapping("")
    public PageResponseVO searchSkuList(
            @RequestParam(name = "count", required = false, defaultValue = "15")
            @Min(value = 1, message = "{page.count.min}")
            @Max(value = 30, message = "{page.count.max}") Integer count,
            @RequestParam(name = "page", required = false, defaultValue = "0")
            @Min(value = 0, message = "{page.number.min}") Integer page) {
        IPage<HjcSkuDO> iPage = hjcSkuService.getAllSku(page, count);
        return PageUtil.build(iPage);
    }

    /**
    根据关键字进行商品搜索
    */
    @GetMapping("/search")
    public PageResponseVO searchSkuList(
            @RequestParam(name = "keyword", required = false, defaultValue = "") String keyword,
            @RequestParam(name = "count", required = false, defaultValue = "15")
            @Min(value = 1, message = "{page.count.min}")
            @Max(value = 30, message = "{page.count.max}") Integer count,
            @RequestParam(name = "page", required = false, defaultValue = "0")
            @Min(value = 0, message = "{page.number.min}") Integer page) {
        IPage<HjcSkuDO> iPage = hjcSkuService.getAllSkuByKeywords(page, count, keyword);
        return PageUtil.build(iPage);
    }

}
```

前端展示

```
<template>
  <div>
    <div v-if="!showEdit" class="container">
      <div class="header">
        <div class="title">商品列表</div>
        <div class="header-left" v-permission="'搜索商品'">
          <lin-search @query="onQueryChange" ref="searchKeyword"/>
          <lin-date-picker @dateChange="handleDateChange" ref="searchDate" class="date"></lin-date-picker>
        </div>
      </div>
      <transition name="fade">
        <div class="search" v-if="keyword">
          <p class="search-tip">
            搜索“<span class="search-keyword">{{ keyword }}</span
          >”， 找到 <span class="search-num">{{ totalNums }}</span> 条商品信息
          </p>
          <button class="search-back" @click="backInit">返回全部商品</button>
        </div>
      </transition>
      <el-table stripe v-loading="loading" :data="tableData">
        <el-table-column :show-overflow-tooltip="true" prop="title" label="商品名称" width="350"/>
        <el-table-column :show-overflow-tooltip="true" prop="type" label="类型" width="250"/>
        <el-table-column :show-overflow-tooltip="true" prop="category" label="分类" width="250"/>
        <el-table-column prop="stock" label="库存数量" width="250"/>
        <el-table-column prop="order_num" label="订单数量" width="250"/>
      </el-table>

      <div class="pagination">
        <el-pagination
          @current-change="handleCurrentChange"
          :background="true"
          :page-size="pageCount"
          :current-page="currentPage"
          v-if="refreshPagination"
          layout="prev, pager, next, jumper"
          :total="totalNums"
        />
      </div>
    </div>
  </div>
</template>
```