

1-项目背景介绍

1.1 项目背景

项目背景介绍：

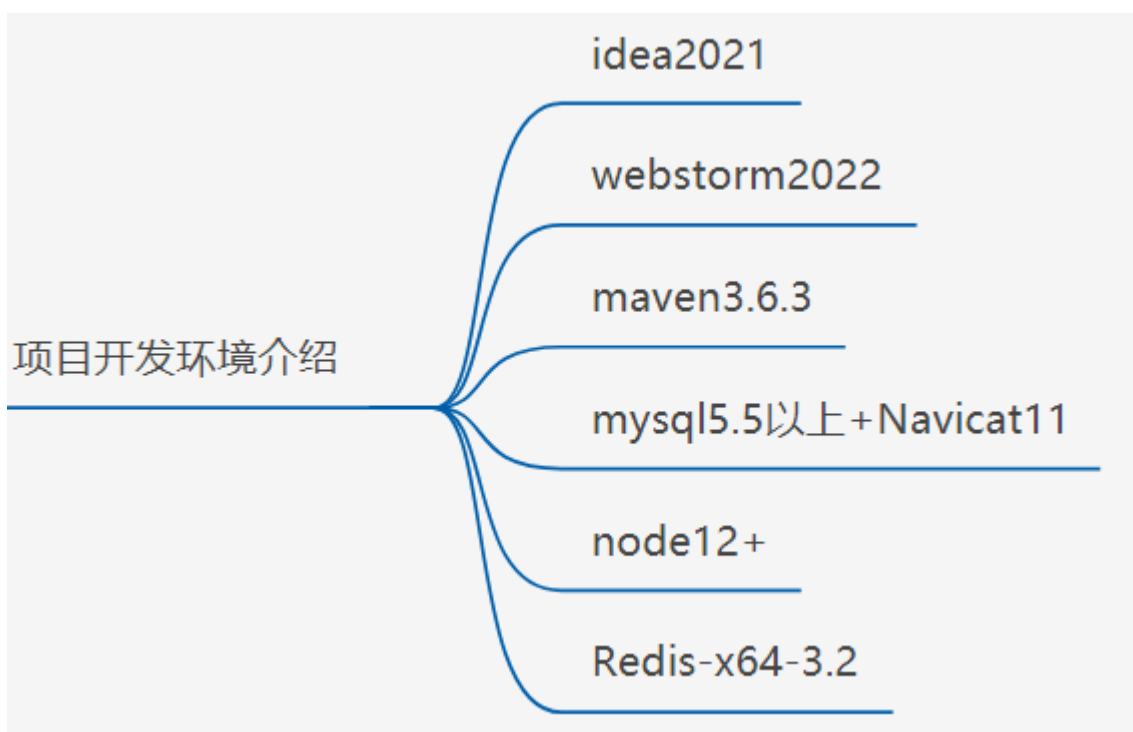
应届毕业生小明，顺利的通过了一家小型IT研发公司的面试，在定级薪资待遇时，技术总监发给小明一份原型文档，该原型是公司刚启动的医保药品进销存管理系统，技术总监和小明说，在8小时内完成药品厂家与药品信息管理二个模块功能，框架和登录自己搭，最好能有权限功能，日志功能等，技术栈不限，但是如果能用最新最合理的技术栈有加分，完成的时间越快，功能越完善，加分项越多。最后根据分数在来定级薪资待遇。

小明忐忑的打开文档，当看到文档的一瞬间，小明嘴边露出了一丝微笑，心道：“幸亏毕业设计是基于若依框架二次开发的，就这原型需求1小时搞定，看来这次可以谈个好待遇了”。

2-项目初始化

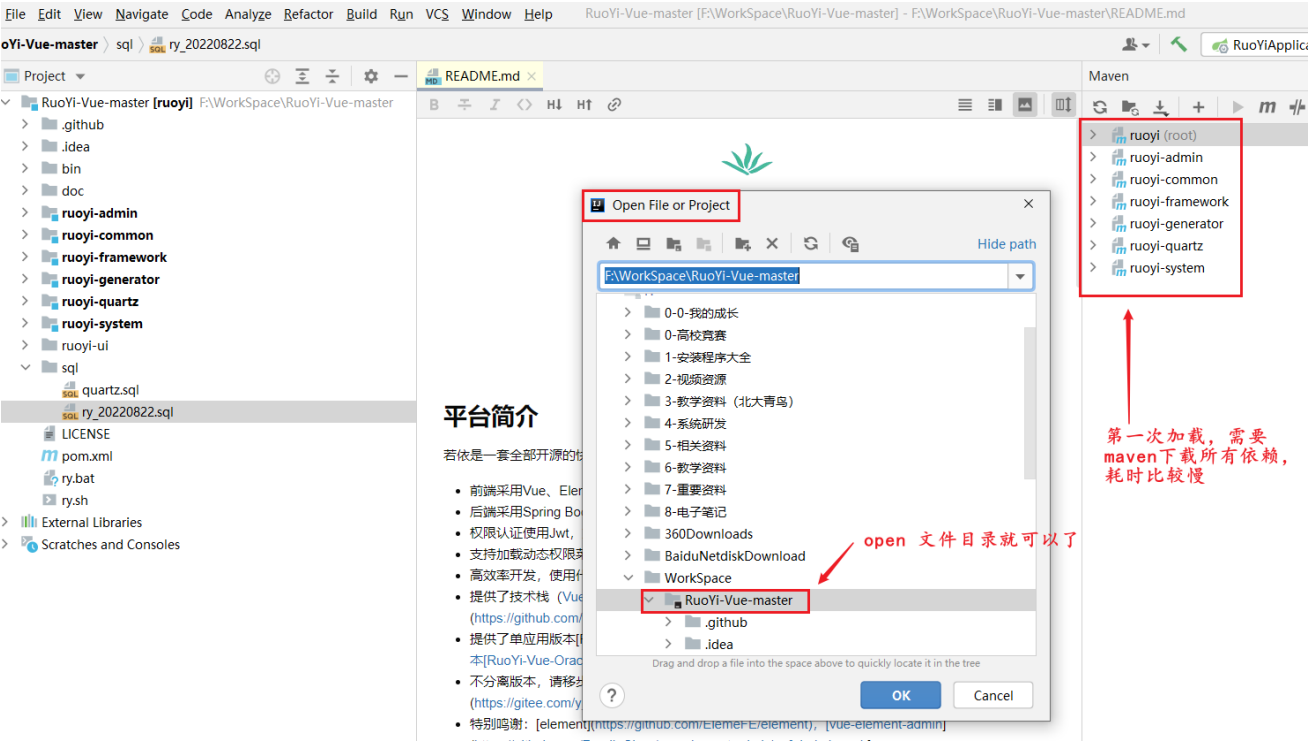
2.1-若依框架介绍与下载

2.2-项目开发环境介绍



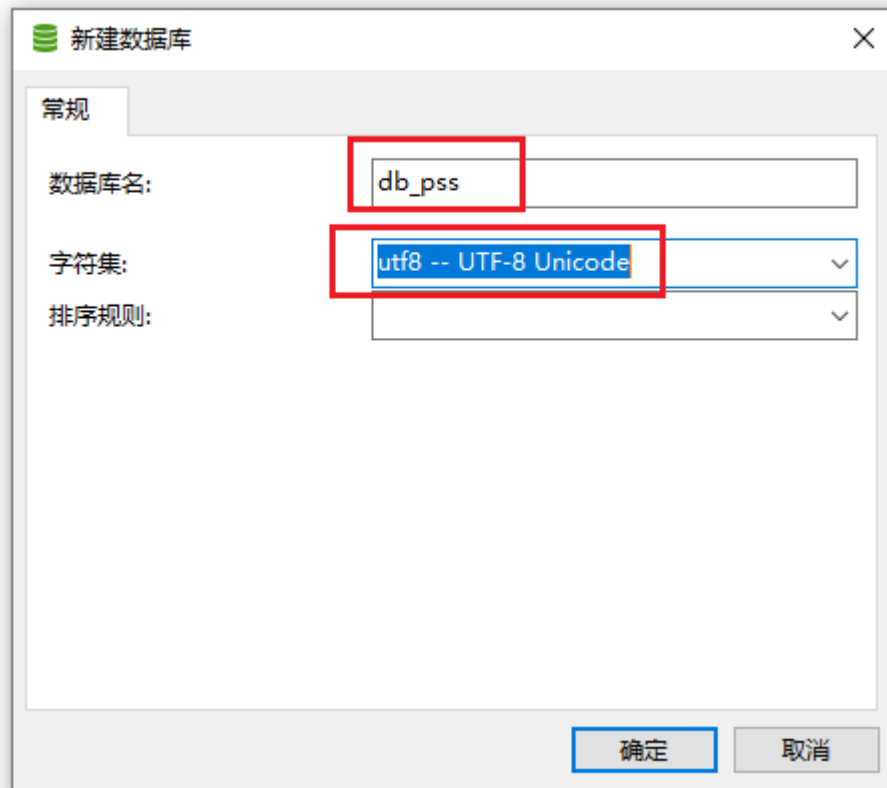
2.3-后端源码加载及配置

步骤1: idea打开项目

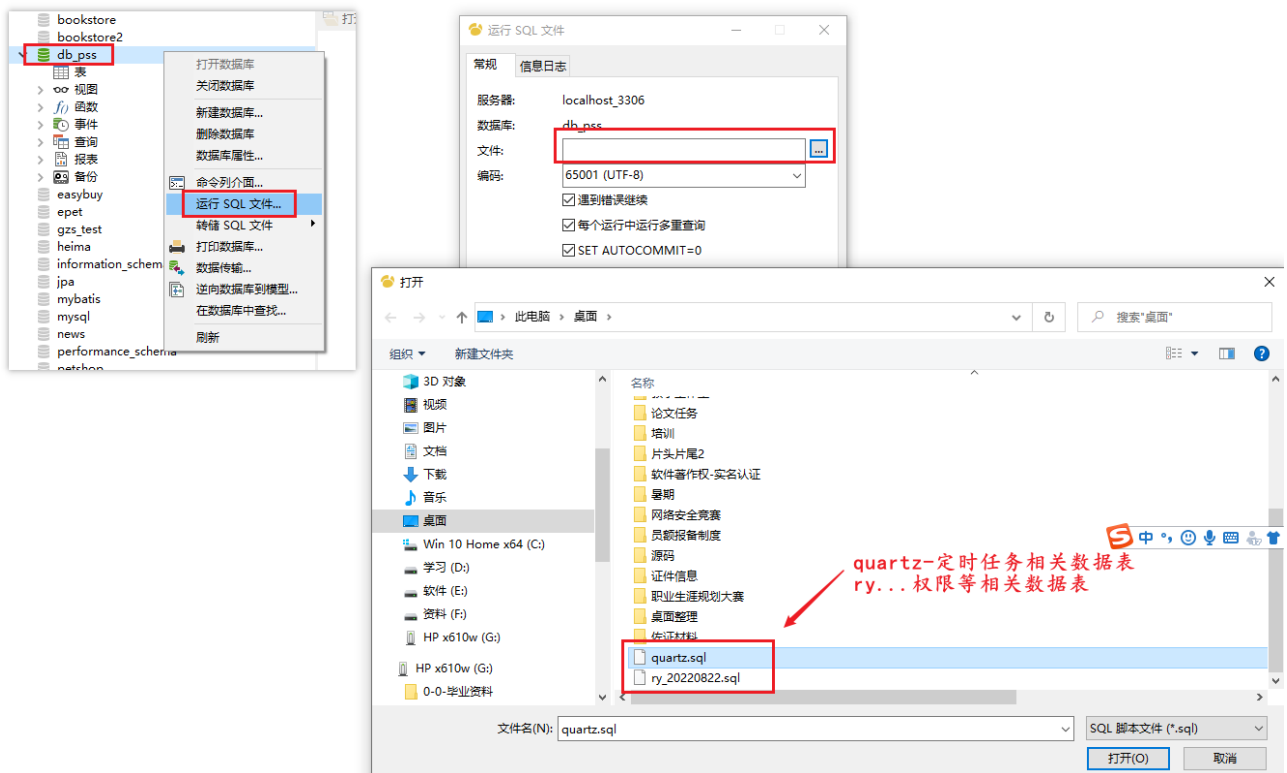


步骤2: 初始化数据库

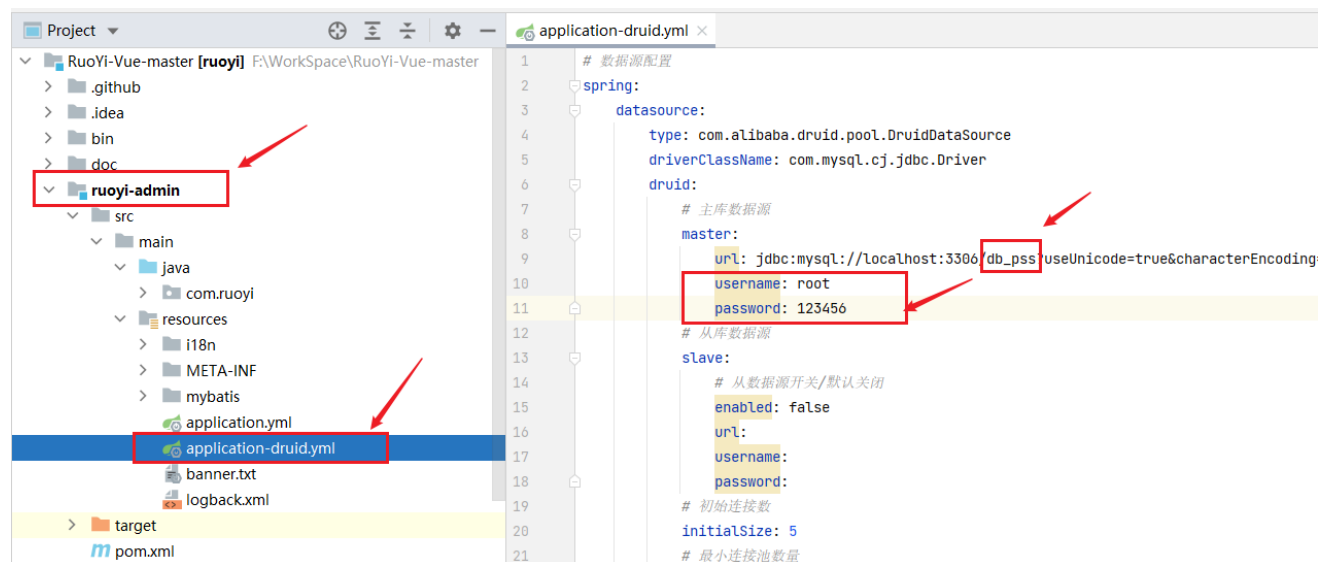
创建数据库



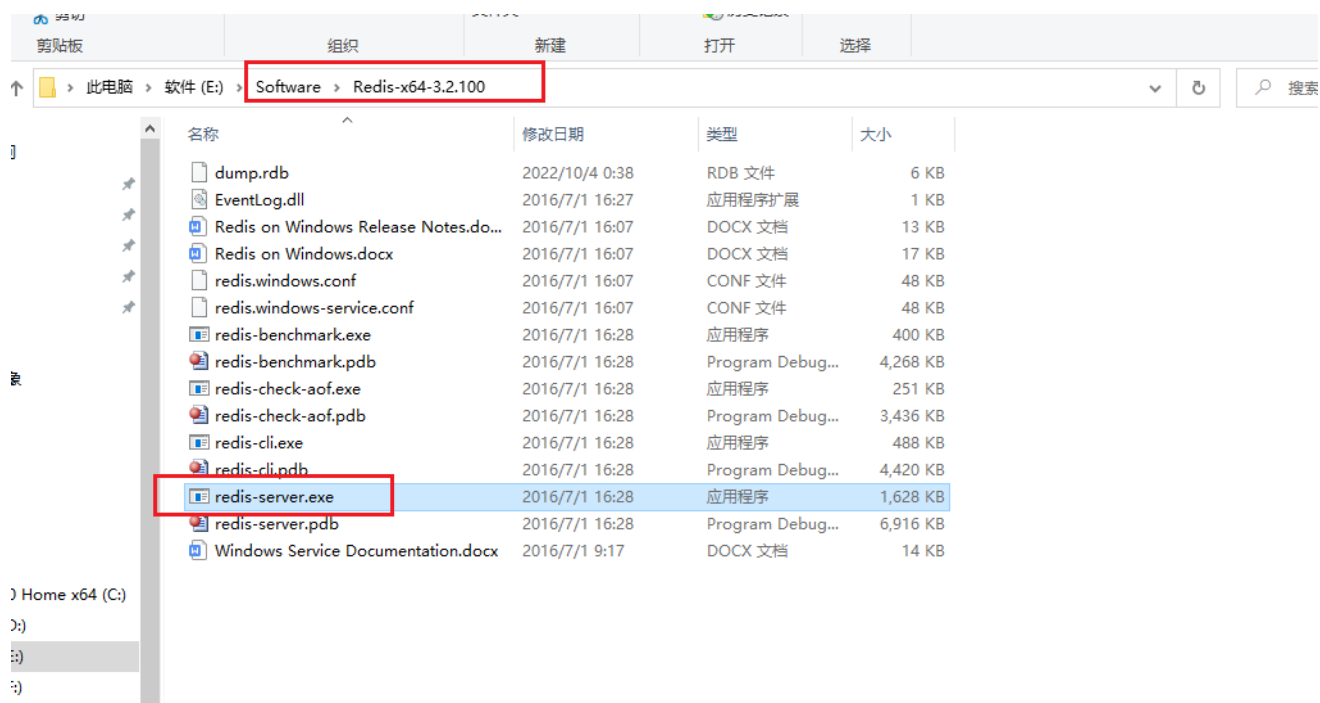
运行SQL脚本



步骤3: 修改后端yaml文件



步骤4: 运行redis服务器



```
E:\Software\Redis-x64-3.2.100\redis-server.exe

[21224] 05 Oct 19:00:33.661 # Warning: no config file specified, using the default config. In order to specify a config
file use E:\Software\Redis-x64-3.2.100\redis-server.exe /path/to/redis.conf

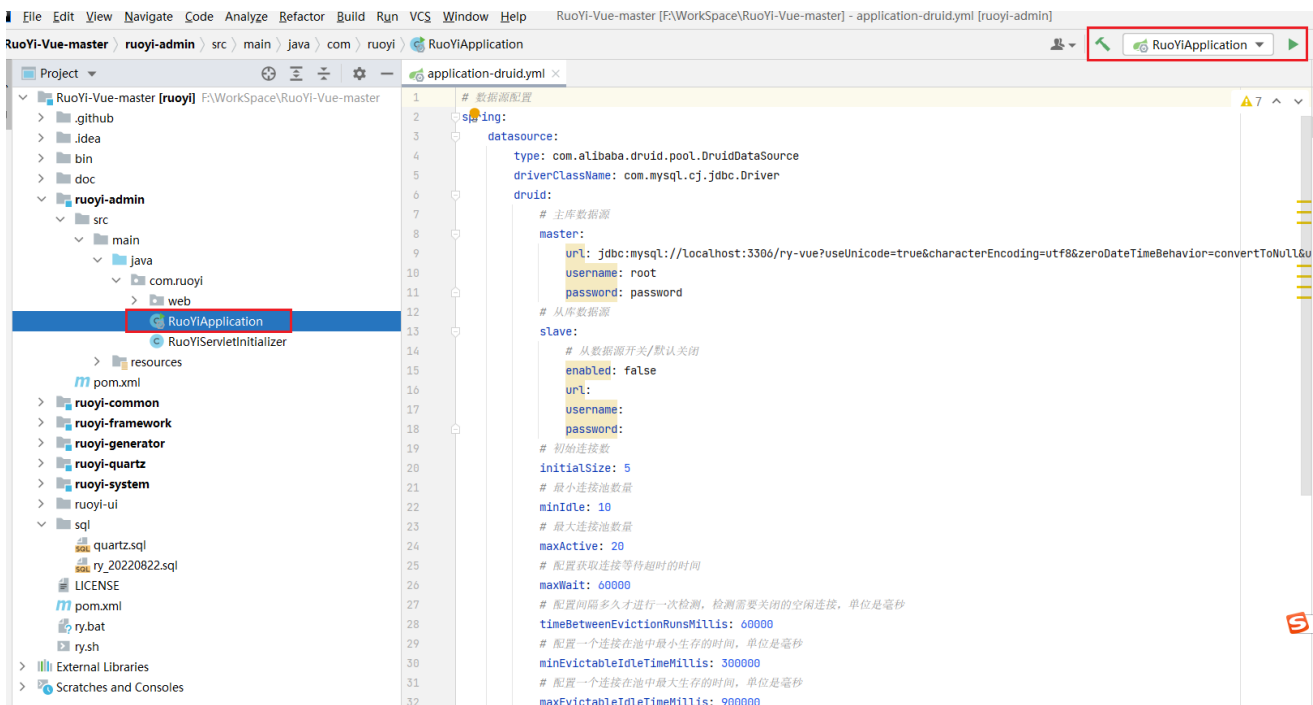
Redis 3.2.100 (00000000/0) 64 bit

Running in standalone mode
Port: 6379
PID: 21224

http://redis.io

[21224] 05 Oct 19:00:33.669 # Server started, Redis version 3.2.100
[21224] 05 Oct 19:00:33.670 * DB loaded from disk: 0.001 seconds
[21224] 05 Oct 19:00:33.670 * The server is now ready to accept connections on port 6379
```

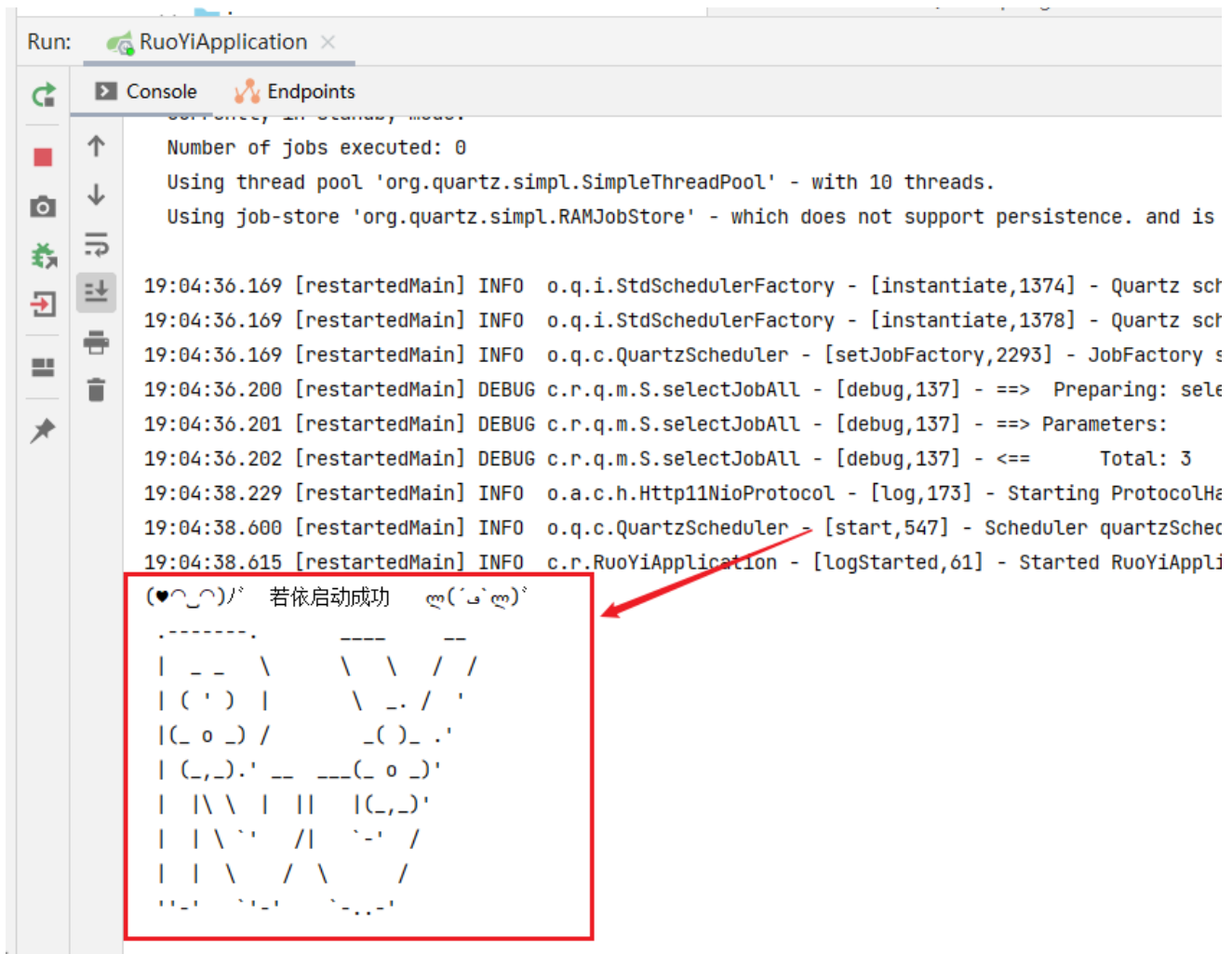
步骤5: 运行启动文件



The screenshot shows an IDE with the following components:

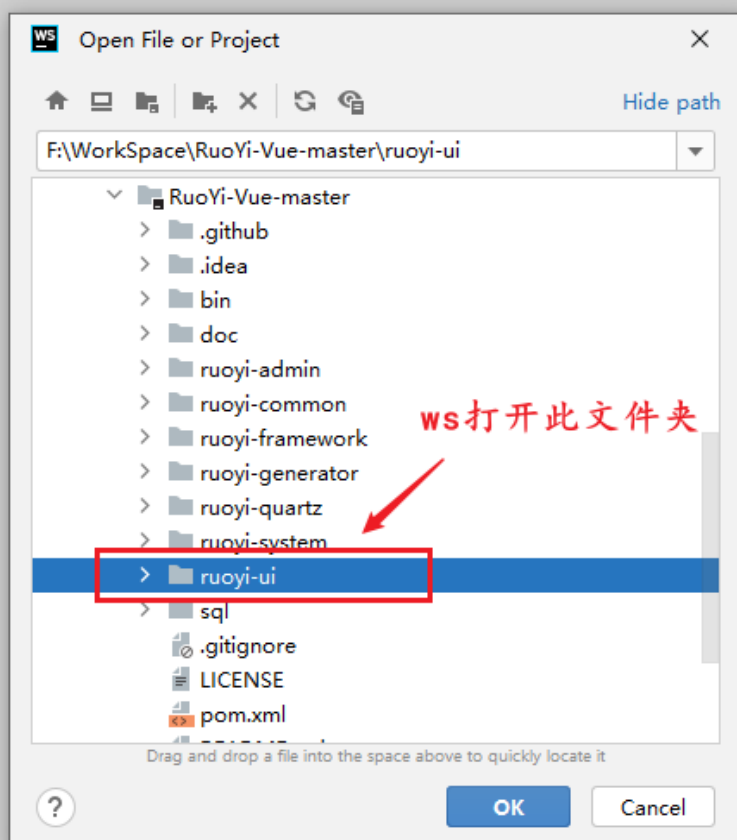
- Project Explorer:** Displays the project structure of `RuoYi-Vue-master`. The `web` directory under `src/main/java/com/ruoyi` is selected, containing `RuoYiApplication` and `RuoYiServletInitializer`.
- Editor:** Shows the `application-druid.yml` configuration file. The configuration is for a Druid data source, including master and slave settings, connection pool parameters, and eviction settings.
- Run Configuration:** A red box highlights the `RuoYiApplication` configuration in the top right corner of the IDE.

```
# 数据源配置
spring:
  datasource:
    type: com.alibaba.druid.pool.DruidDataSource
    driverClassName: com.mysql.cj.jdbc.Driver
    druid:
      # 主库数据源
      master:
        url: jdbc:mysql://localhost:3306/ry-vue?useUnicode=true&characterEncoding=utf8&zeroDateBehavior=convertToNull&
        username: root
        password: password
      # 从库数据源
      slave:
        # 从数据源开关/默认关闭
        enabled: false
        url:
        username:
        password:
      # 初始连接数
      initialSize: 5
      # 最小连接池数量
      minIdle: 10
      # 最大连接池数量
      maxActive: 20
      # 配置获取连接等待超时的时间
      maxWait: 60000
      # 配置间隔多久才进行一次检测，检测需要关闭的空闲连接，单位是毫秒
      timeBetweenEvictionRunsMillis: 60000
      # 配置一个连接在池中最小生存的时间，单位是毫秒
      minEvictableIdleTimeMillis: 300000
      # 配置一个连接在池中最大生存的时间，单位是毫秒
      maxEvictableIdleTimeMillis: 900000
```

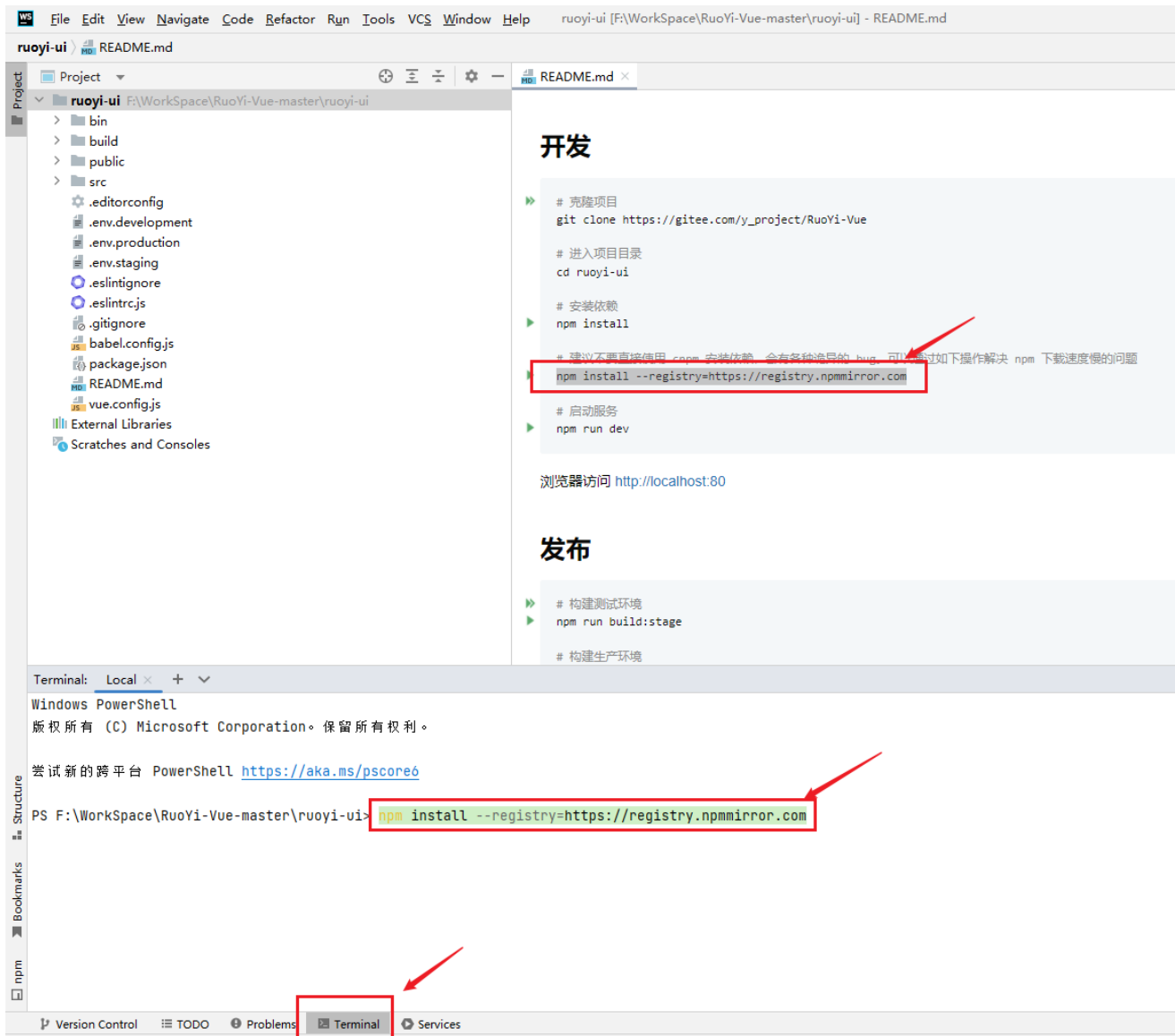


2.4-前端源码加载

步骤1: 打开前端项目文件



步骤2：安装依赖



步骤3：启动服务

package.json
README.md
vue.config.js
External Libraries
Scratches and Consoles

```
npm install --registry=https://registry.npmmir  
  
# 启动服务  
npm run dev
```

浏览器访问 <http://localhost:80>

发布

```
>> # 构建测试环境  
> npm run build:stage  
  
# 构建生产环境
```

Terminal: Local x + v

```
ipm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@~2.3.2 (node_modules\chokidar\node_modules\fs  
ipm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"d  
ipm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@^1.2.7 (node_modules\watchpack-chokidar2\node.  
ipm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"  
ipm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@^1.2.7 (node_modules\webpack-dev-server\node_r  
ipm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"  
ipm WARN eslint-loader@2.2.1 requires a peer of eslint@>=1.6.0 <7.0.0 but none is installed. You must :
```

added 1575 packages from 949 contributors in 105.181s

.17 packages are looking for funding

run `npm fund` for details

'S F:\WorkSpace\RuoYi-Vue-master\ruoyi-ui> npm run dev

启动服务

3-模块开发-生产厂家

3.1 后端开发

3.1.1 设计厂家与药品数据表

根据页面原型及参考作者原有表 进行设计

首页 / 药品购销存管理 / 生产厂家维护

生产厂家维护

厂家名称

请输入厂家名称

关键字

请输入关键字

厂家电话

请输入厂家电话

状态

厂家状态

搜索

重置

+新增

修改

删除

导出

	厂家ID	厂家名称	厂家编码	联系人	电话	关键字	状态	创建时间	操作
<input type="checkbox"/>	101	云南白药集团股份有限公司	000011	两毅米	023-88887777	ynby	正常	2022-08-26 16:15:29	修改 删除
<input type="checkbox"/>	102	哈药集团股份有限公司	000022	两毅米	023-88887777	hyjt	正常	2022-08-26 16:15:29	修改 删除
<input type="checkbox"/>	103	太极集团有限公司	000033	两毅米	023-88887777	jt	正常	2022-08-26 16:15:29	修改 删除
<input type="checkbox"/>	104	上海医药集团有限公司	000033	两毅米	023-99998888	shyy	正常	2022-08-26 20:12:57	修改 删除

共 4 条 10条/页 < 1 > 前往 1 页

post_id	post_code	post_name	post_sort	status	create_by	create_time	update_by	update_time	remark
1	ceo	董事长	1	0	admin	2022-10-05 18:55:53		(Null)	
2	se	项目经理	2	0	admin	2022-10-05 18:55:53		(Null)	
3	hr	人力资源	3	0	admin	2022-10-05 18:55:53		(Null)	
4	user	普通员工	4	0	admin	2022-10-05 18:55:53		(Null)	

所有表都有这四个字段

3.1.2 创建子模块及各层

步骤1：创建子模块

RuoYi-Vue-master Iruovii

New

Module...

File

Scratch File

Directory

HTML File

Stylesheet

JavaScript File

TypeScript File

package.json File

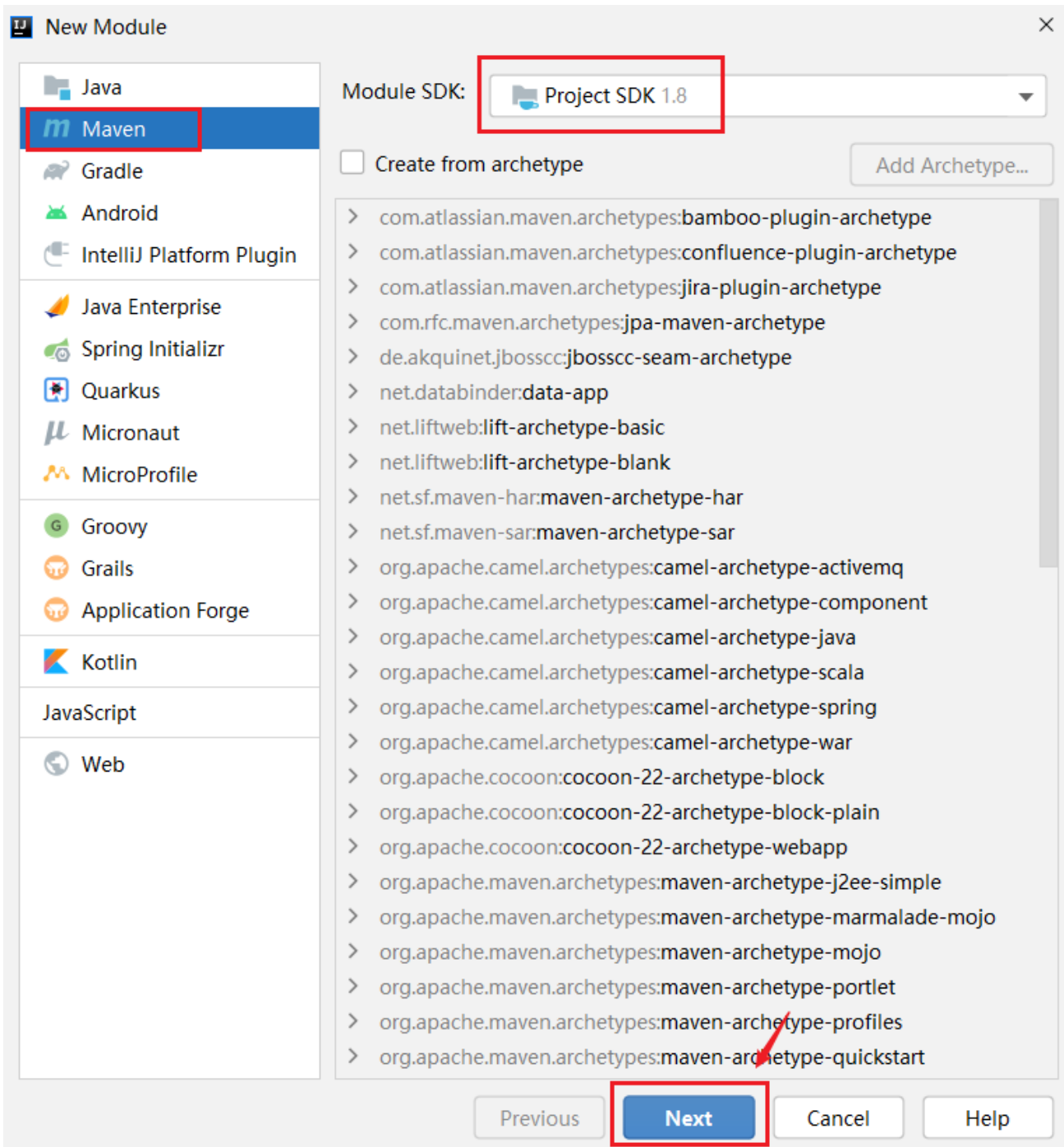
Vue Component

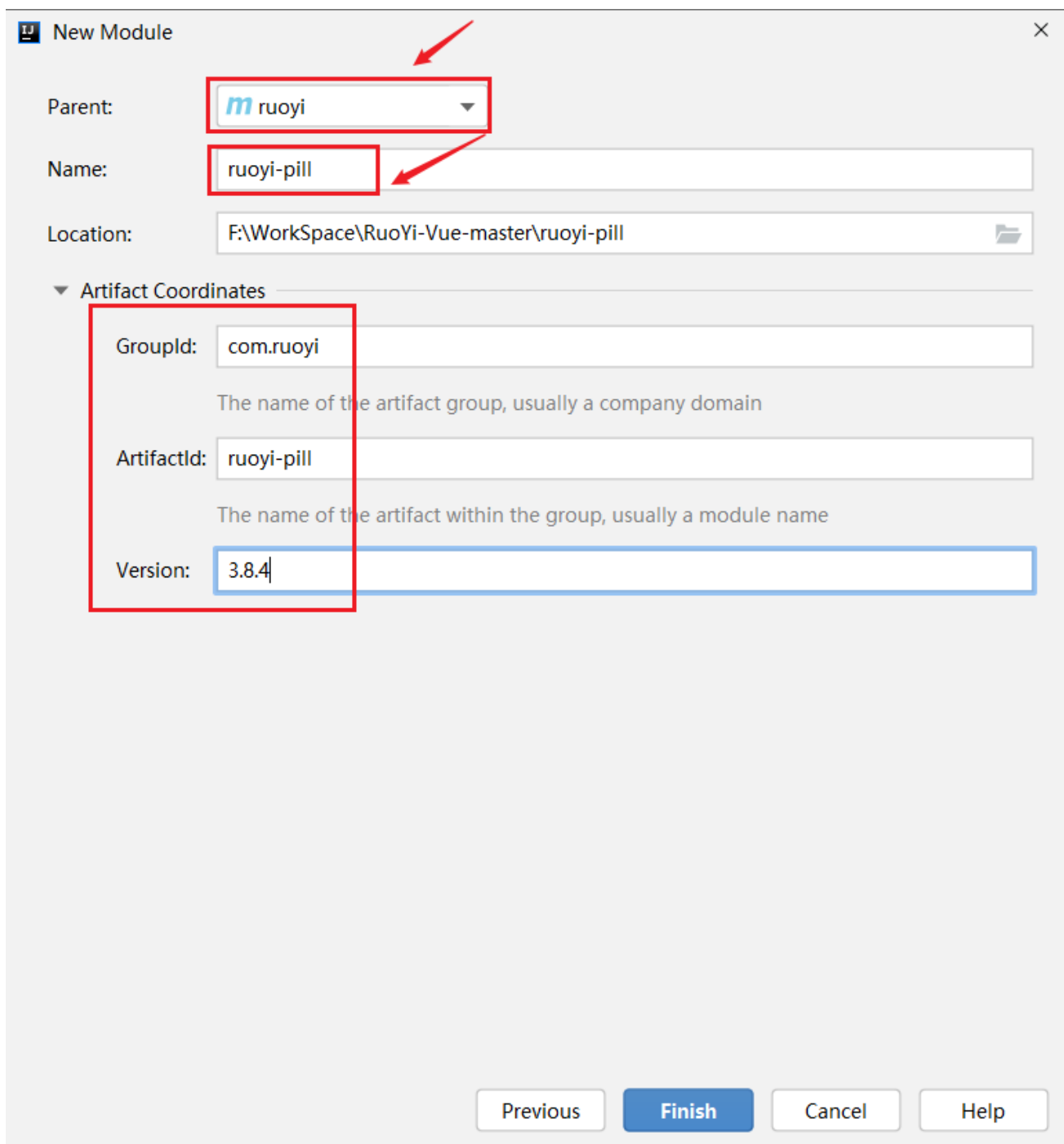
Kotlin Script

Kotlin Worksheet

OpenAPI Specification

子模块



The image shows a 'New Module' dialog box with several fields. The 'Parent' field is a dropdown menu showing 'm ruoyi'. The 'Name' field contains 'ruoyi-pill'. The 'Location' field contains 'F:\Workspace\RuoYi-Vue-master\ruoyi-pill'. The 'Artifact Coordinates' section is expanded, showing 'GroupId' as 'com.ruoyi', 'ArtifactId' as 'ruoyi-pill', and 'Version' as '3.8.4'. Red arrows point to the 'Parent' and 'Name' fields. A red box highlights the 'Artifact Coordinates' section. The 'Finish' button is highlighted in blue.

New Module

Parent:

Name:

Location:

▼ Artifact Coordinates

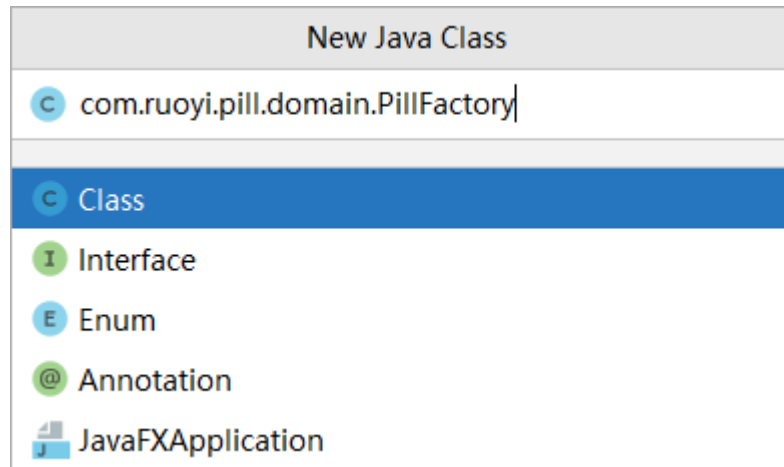
GroupId:
The name of the artifact group, usually a company domain

ArtifactId:
The name of the artifact within the group, usually a module name

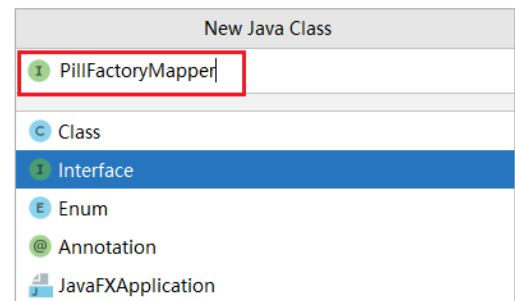
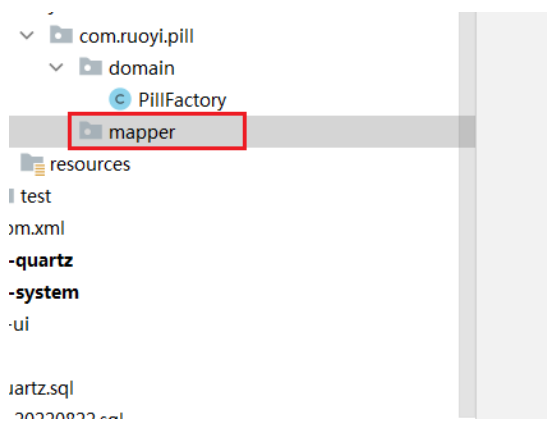
Version:

Previous Finish Cancel Help

步骤2：创建实体层

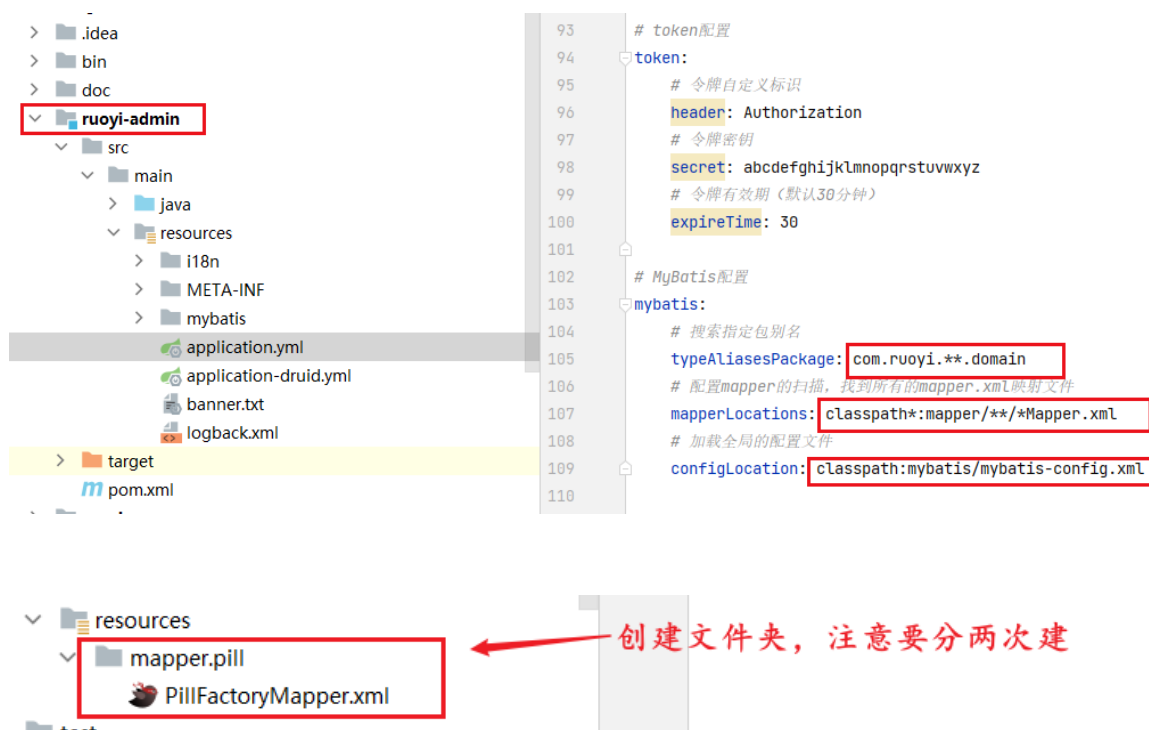


步骤3：创建数据访问层



映射XML文件

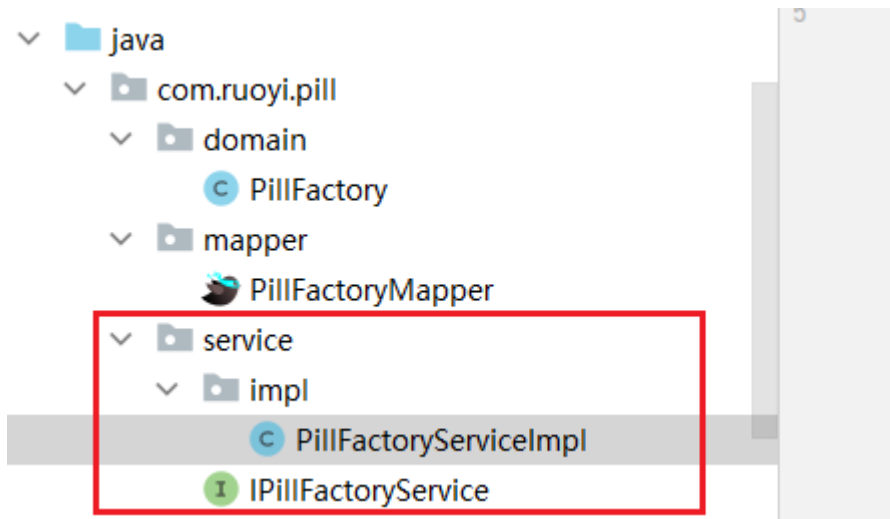
根据配置文件application.yml



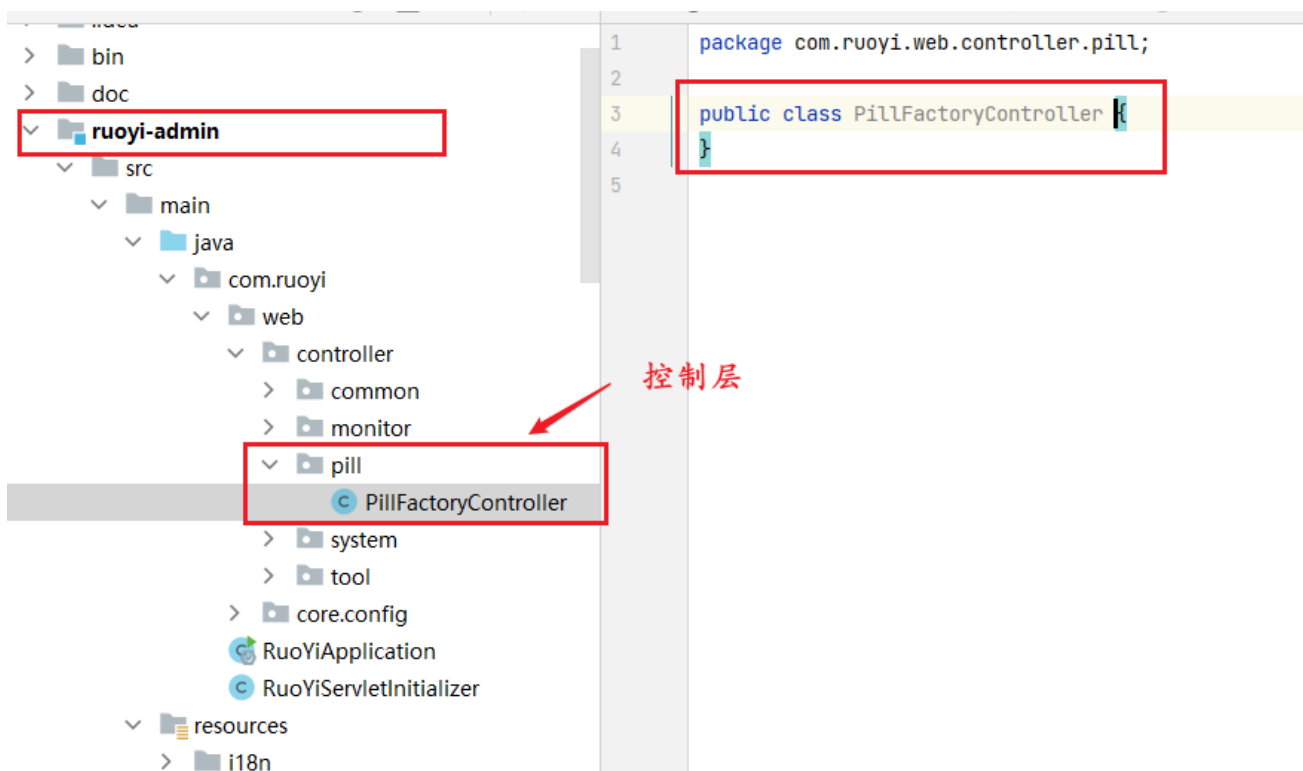
PillFactoryMapper.xml原始代码

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <!DOCTYPE mapper
3     PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
4     "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
5 <mapper namespace="com.ruoyi.pill.mapper.PillFactoryMapper">
6
7 </mapper>
```

步骤4: 创建服务层



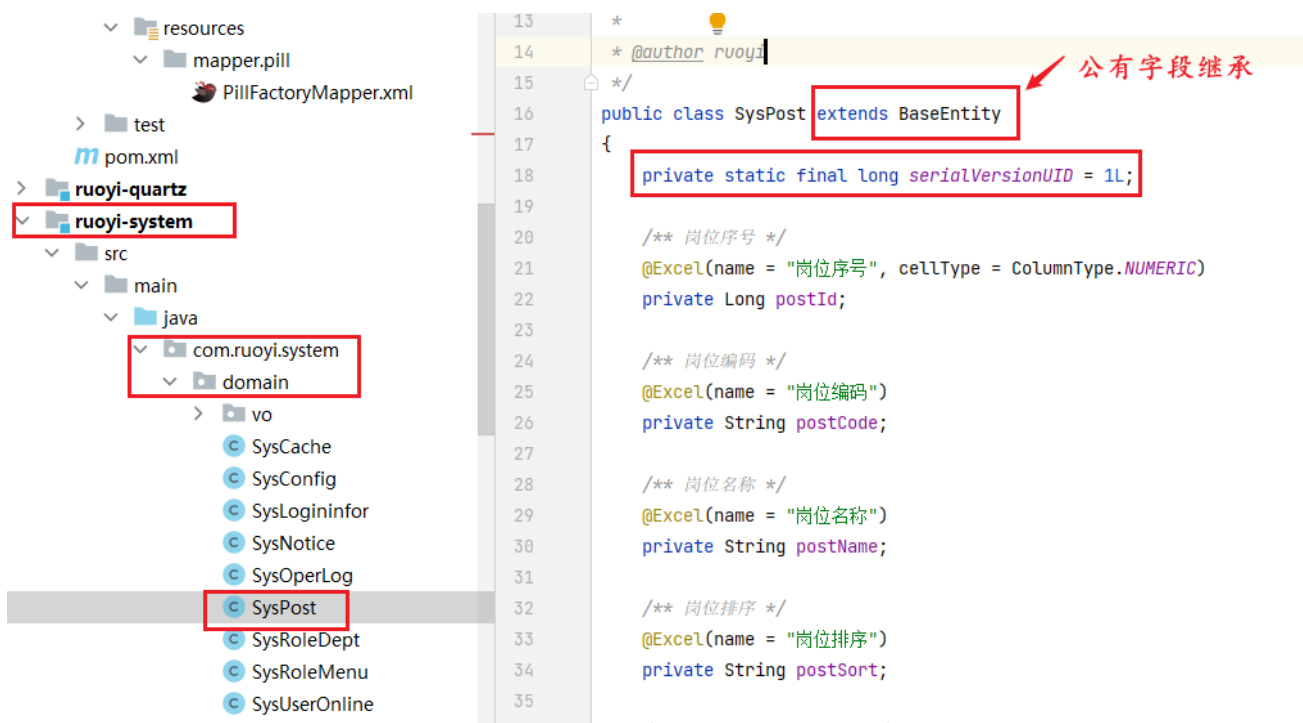
步骤5: 创建控制层



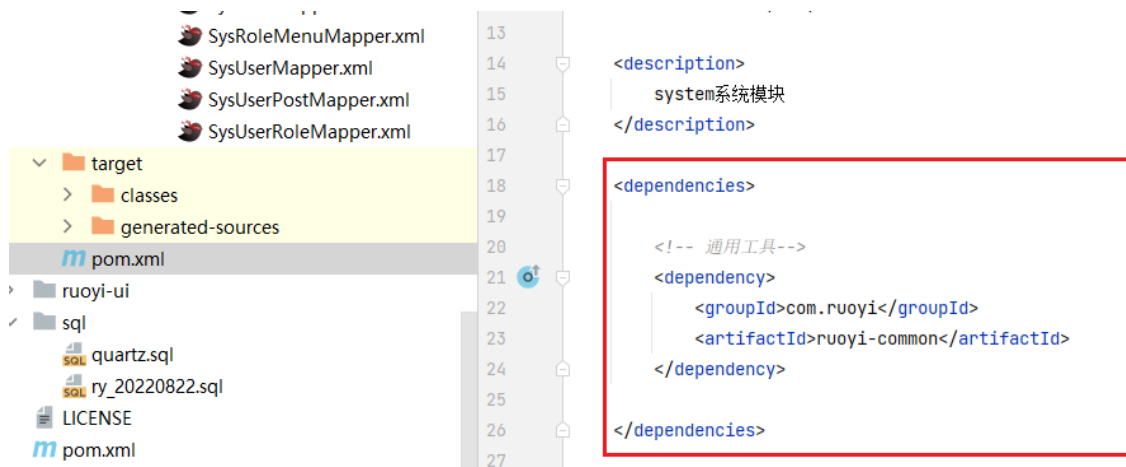
3.1.3 常规开发-查询生产厂家

步骤1: 实体层

参考



参考POM代码



参考代码

```
1 public class PillFactory extends BaseEntity {
2     private static final long serialVersionUID = 1L;
3     private Long factoryId;
4     private String factoryName;
5     private String factoryCode;
6     private String contact;
7     private String phone;
8     private String keyword;
9     private String status;
10
11     public Long getFactoryId() {
12         return factoryId;
13     }
14
15     public void setFactoryId(Long factoryId) {
16         this.factoryId = factoryId;
17     }
18
19     public String getFactoryName() {
20         return factoryName;
21     }
22
23     public void setFactoryName(String factoryName) {
24         this.factoryName = factoryName;
25     }
26
27     public String getFactoryCode() {
28         return factoryCode;
29     }
30
31     public void setFactoryCode(String factoryCode) {
32         this.factoryCode = factoryCode;
33     }
34
35     public String getContact() {
36         return contact;
```



```

37     }
38
39     public void setContact(String contact) {
40         this.contact = contact;
41     }
42
43     public String getPhone() {
44         return phone;
45     }
46
47     public void setPhone(String phone) {
48         this.phone = phone;
49     }
50
51     public String getKeyword() {
52         return keyword;
53     }
54
55     public void setKeyword(String keyword) {
56         this.keyword = keyword;
57     }
58
59     public String getStatus() {
60         return status;
61     }
62
63     public void setStatus(String status) {
64         this.status = status;
65     }
66
67     @Override
68     public String toString() {
69         return "PillFactory{" +
70             "factoryId=" + factoryId +
71             ", factoryName='" + factoryName + '\'' +
72             ", factoryCode='" + factoryCode + '\'' +
73             ", contact='" + contact + '\'' +
74             ", phone='" + phone + '\'' +
75             ", keyword='" + keyword + '\'' +
76             ", status='" + status + '\'' +
77             '}';
78     }
79 }

```

步骤2：数据访问层

接口代码

```

1 public interface PillFactoryMapper {
2     public List<PillFactory> selectPillFactoryList(PillFactory factory);
3 }

```

XML代码

```

1 <resultMap type="PillFactory" id="PillFactoryResult">
2     <id property="factoryId" column="factory_id"/>
3     <result property="factoryName" column="factory_name"/>
4     <result property="factoryCode" column="factory_code"/>
5     <result property="contact" column="contact"/>
6     <result property="phone" column="phone"/>
7     <result property="keyword" column="keyword"/>
8     <result property="status" column="status"/>
9     <result property="createBy" column="create_by"/>
10    <result property="createTime" column="create_time"/>
11    <result property="updateBy" column="update_by"/>
12    <result property="updateTime" column="update_time"/>
13    <result property="remark" column="remark"/>
14 </resultMap>
15
16 <sql id="selectFactoryVo">
17     select factory_id, factory_name, factory_code, contact, phone, keyword, status,
18     create_by,
19     create_time, update_by, update_time, remark
20     from pill_factory
21 </sql>
22
23 <select id="selectPillFactoryList" parameterType="PillFactory"
24 resultMap="PillFactoryResult">
25     <include refid="selectFactoryVo"/>
26     <where>
27         <if test="factoryName != null and factoryName != ''">
28             AND factory_name like concat('%', #{factoryName}, '%')
29         </if>
30         <if test="keyword != null and keyword != ''">
31             AND keyword like concat('%', #{keyword}, '%')
32         </if>
33         <if test="phone != null and phone != ''">
34             AND phone like concat('%', #{phone}, '%')
35         </if>
36         <if test="status != null and status != ''">
37             AND status = #{status}
38         </if>
39     </where>
40 </select>

```

步骤3: 服务层代码

接口代码

```
1 public List<PillFactory> selectPillFactoryList(PillFactory factory);
```

实现类代码

```
1 @Service
2 public class PillFactoryServiceImpl implements IPillFactoryService {
3     @Autowired
4     private PillFactoryMapper pillFactoryMapper;
5
6     @Override
7     public List<PillFactory> selectPillFactoryList(PillFactory factory) {
8         return pillFactoryMapper.selectPillFactoryList(factory);
9     }
10 }
```

步骤4：测试类测试



pom依赖

```
1 <dependency>
2     <groupId>org.springframework.boot</groupId>
3     <artifactId>spring-boot-starter-test</artifactId>
4 </dependency>
```

PssTest代码

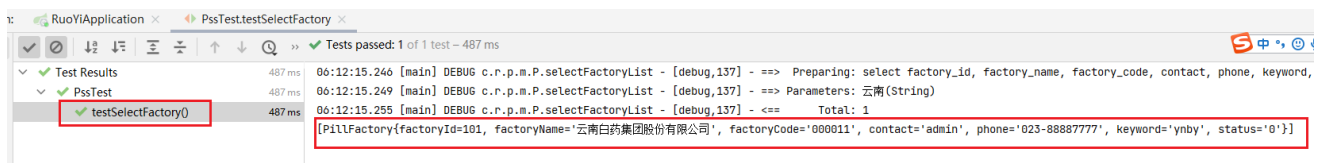
```
1 package com.ruoyi.web;
2
3 import com.ruoyi.pill.domain.PillFactory;
4 import com.ruoyi.pill.service.IPillFactoryService;
5 import org.junit.jupiter.api.Test;
6 import org.springframework.beans.factory.annotation.Autowired;
7 import org.springframework.boot.test.context.SpringBootTest;
```

```

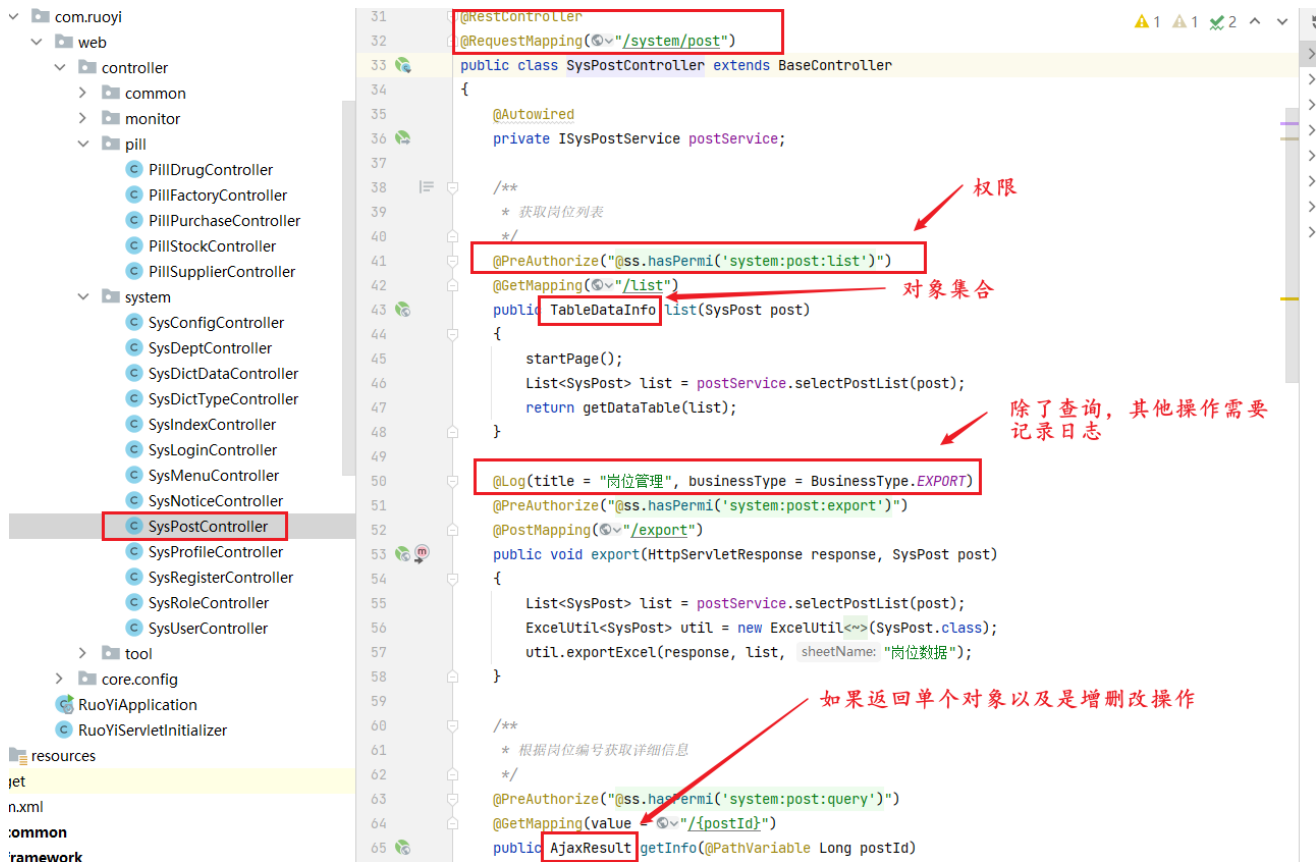
8
9 import java.util.List;
10
11 @SpringBootTest
12 public class PssTest {
13     @Autowired
14     private IPillFactoryService pillFactoryService;
15
16     @Test
17     void contextLoads() {
18     }
19
20     @Test
21     public void testSelectFactory(){
22         PillFactory pillFactory = new PillFactory();
23         pillFactory.setFactoryName("云南");
24         List<PillFactory> pillFactories =
25         pillFactoryService.selectPillFactoryList(pillFactory);
26         System.out.println(pillFactories);
27     }
28 }

```

运行结果



3.1.4 常规开发-controller代码



创建controller层

pom代码

```

1 <!-- 加入ruoyi-pill依赖-->
2 <dependency>
3     <groupId>com.ruoyi</groupId>
4     <artifactId>ruoyi-pill</artifactId>
5     <version>3.8.4</version>
6 </dependency>

```

控制层代码

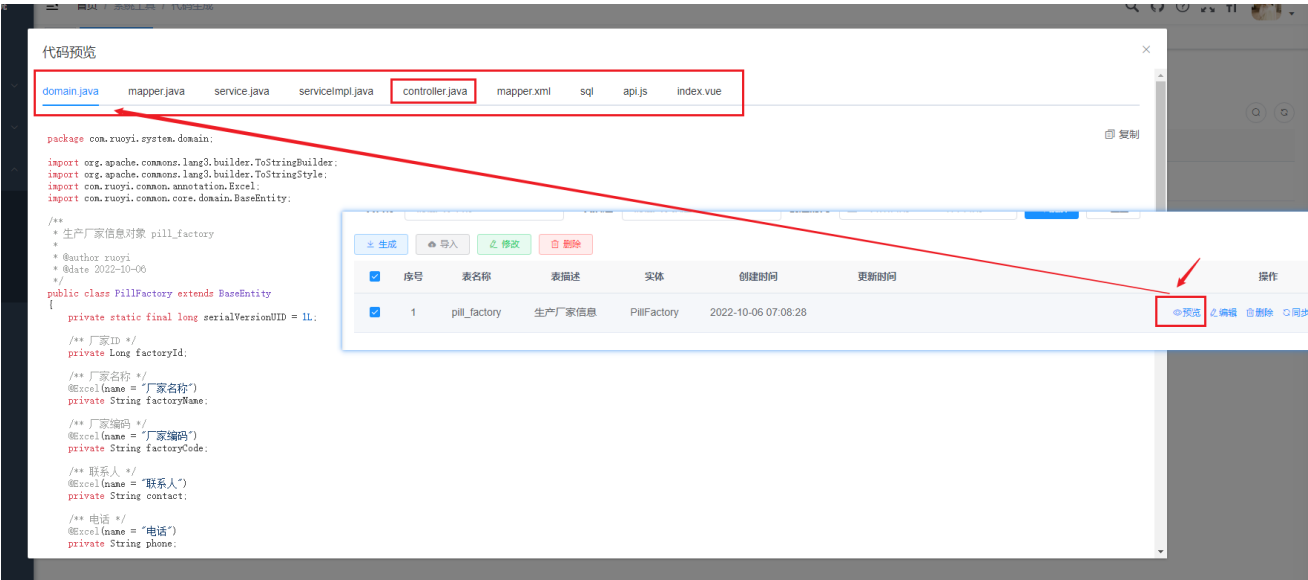
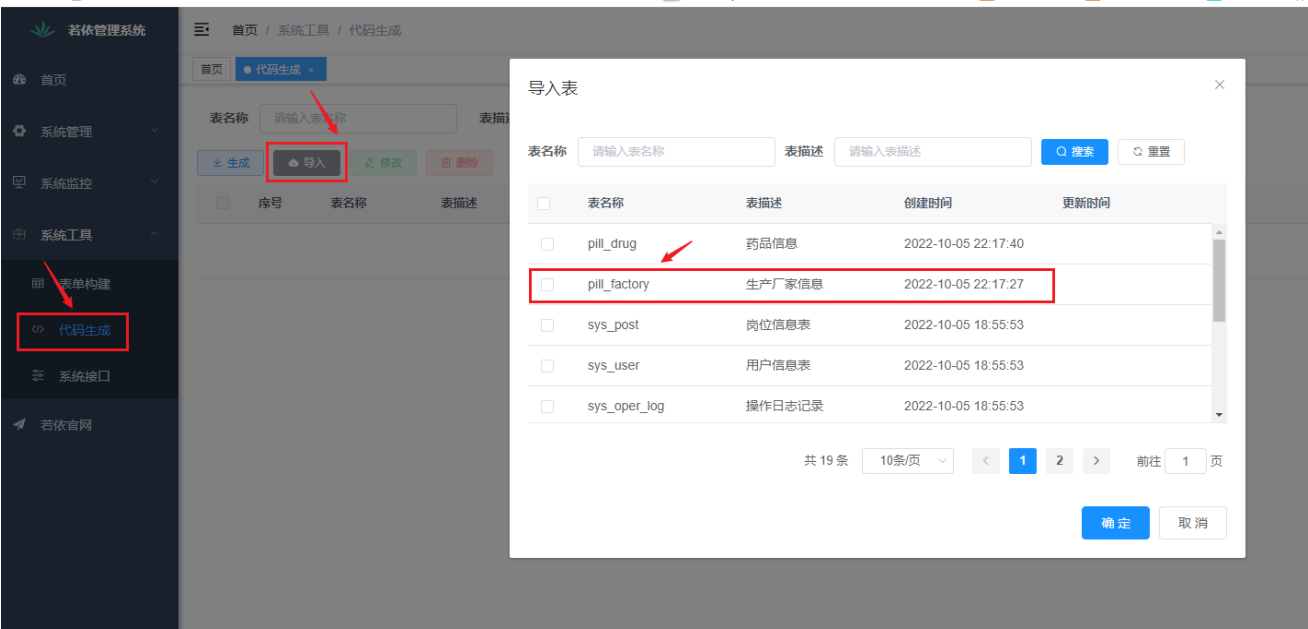
```

1 @RestController
2 @RequestMapping("/pill/factory")
3 public class PillFactoryController extends BaseController {
4     @Autowired
5     private IPillFactoryService pillFactoryService;
6
7     /**
8      * 获取生产厂家列表
9      */
10    @PreAuthorize("@ss.hasPermi('pill:factory:list')")
11    @GetMapping("/list")
12    public TableDataInfo list(PillFactory factory) {
13        startPage();
14        List<PillFactory> list = pillFactoryService.selectPillFactoryList(factory);
15        return getDataTable(list);

```

```
16     }
17 }
```

3.1.4 框架代码生成模式开发



注意事项

代码预览

domain.java mapper.java service.java serviceImpl.java controller.java mapper.xml sql api.js index.vue

```
package com.ruoyi.system.service;

import java.util.List;
import com.ruoyi.system.domain.PillFactory;

/**
 * 生产厂家信息服务接口
 *
 * @author ruoyi
 * @date 2022-10-06
 */
public interface IPillFactoryService {

    /**
     * 查询生产厂家信息
     *
     * @param factoryId 生产厂家信息主键
     * @return 生产厂家信息
     */
    public PillFactory selectPillFactoryByFactoryId(Long factoryId);

    /**
     * 查询生产厂家信息列表
     *
     * @param pillFactory 生产厂家信息
     * @return 生产厂家信息集合
     */
    public List<PillFactory> selectPillFactoryList(PillFactory pillFactory);

    /**
     * 新增生产厂家信息
     *
     * @param pillFactory 生产厂家信息
     * @return 结果
     */
}
```

修改配置

The screenshot shows an IDE with the following components:

- Project Structure (Left):** A tree view showing the project hierarchy. The `generator.yml` file is highlighted in the `src/main/resources` directory.
- Configuration File (Right):** The `generator.yml` file is open in the editor. The configuration is as follows:

```
1 # 代码生成
2 gen:
3   # 作者
4   author: ruoyi
5   # 默认生成包路径 system 需改成自己的模块名称 如 system monitor tool
6   packageName: com.ruoyi.system
7   # 自动去除表前缀，默认是false
8   autoRemovePre: false
9   # 表前缀（生成类名不会包含表前缀，多个用逗号分隔）
10  tablePrefix: sys_
```
- Annotations:** Red boxes and arrows highlight specific configuration values and their corresponding project paths:
 - `author: ruoyi` is highlighted with a red box and an arrow pointing to the text `bbj`.
 - `packageName: com.ruoyi.system` is highlighted with a red box and an arrow pointing to the text `com.ruoyi.pill`.
 - `tablePrefix: sys_` is highlighted with a red box and an arrow pointing to the text `pill_`.

重启服务器，需要把原始的记录删除后，在导入，这时候生成的就可以看到我们的配置信息了

代码预览

domain.java mapper.java service.java serviceImpl.java controller.java mapper.xml sql api.js index.vue

```
package com.ruoyi.pill.service;

import java.util.List;
import com.ruoyi.pill.domain.PillFactory;

/**
 * 生产厂家信息.Service接口
 */
* @author bbi
* @date 2022-10-06
*/
public interface IPillFactoryService
{
    /**
     * 查询生产厂家信息
     *
     * @param factoryId 生产厂家信息主键
     * @return 生产厂家信息
     */
    public PillFactory selectPillFactoryByFactoryId(Long factoryId);

    /**
     * 查询生产厂家信息列表
     *
     * @param pillFactory 生产厂家信息
     * @return 生产厂家信息集合
     */
    public List<PillFactory> selectPillFactoryList(PillFactory pillFactory);

    /**
     * 新增生产厂家信息
     *
     * @param pillFactory 生产厂家信息
     * @return 结果
     */
    public int insertPillFactory(PillFactory pillFactory);
}
```

生成

导入

修改

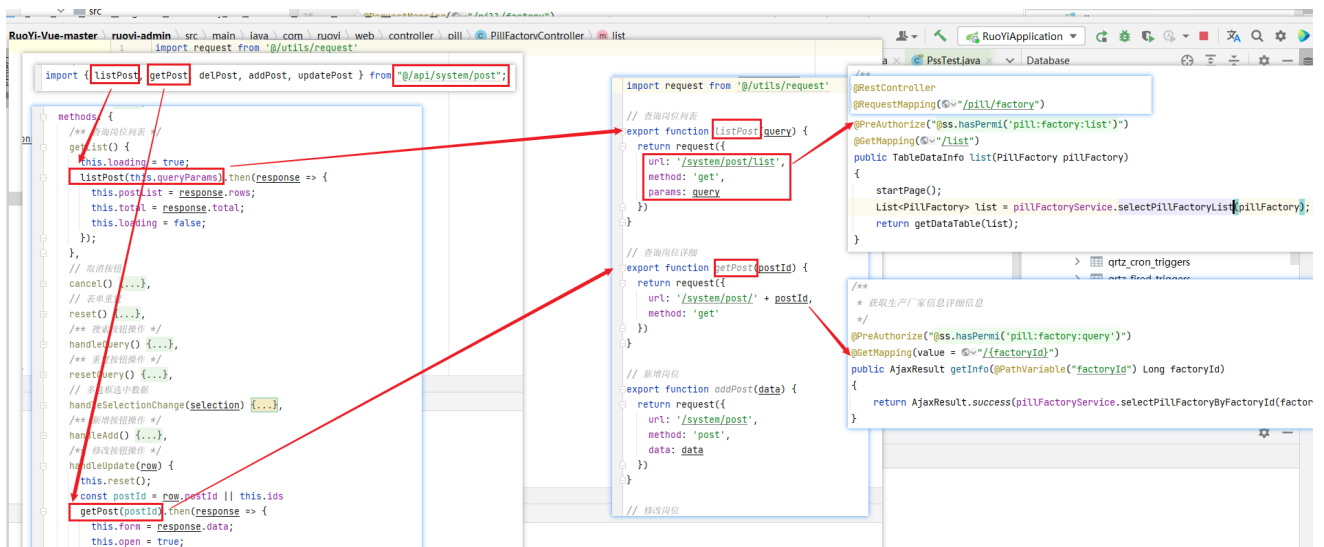
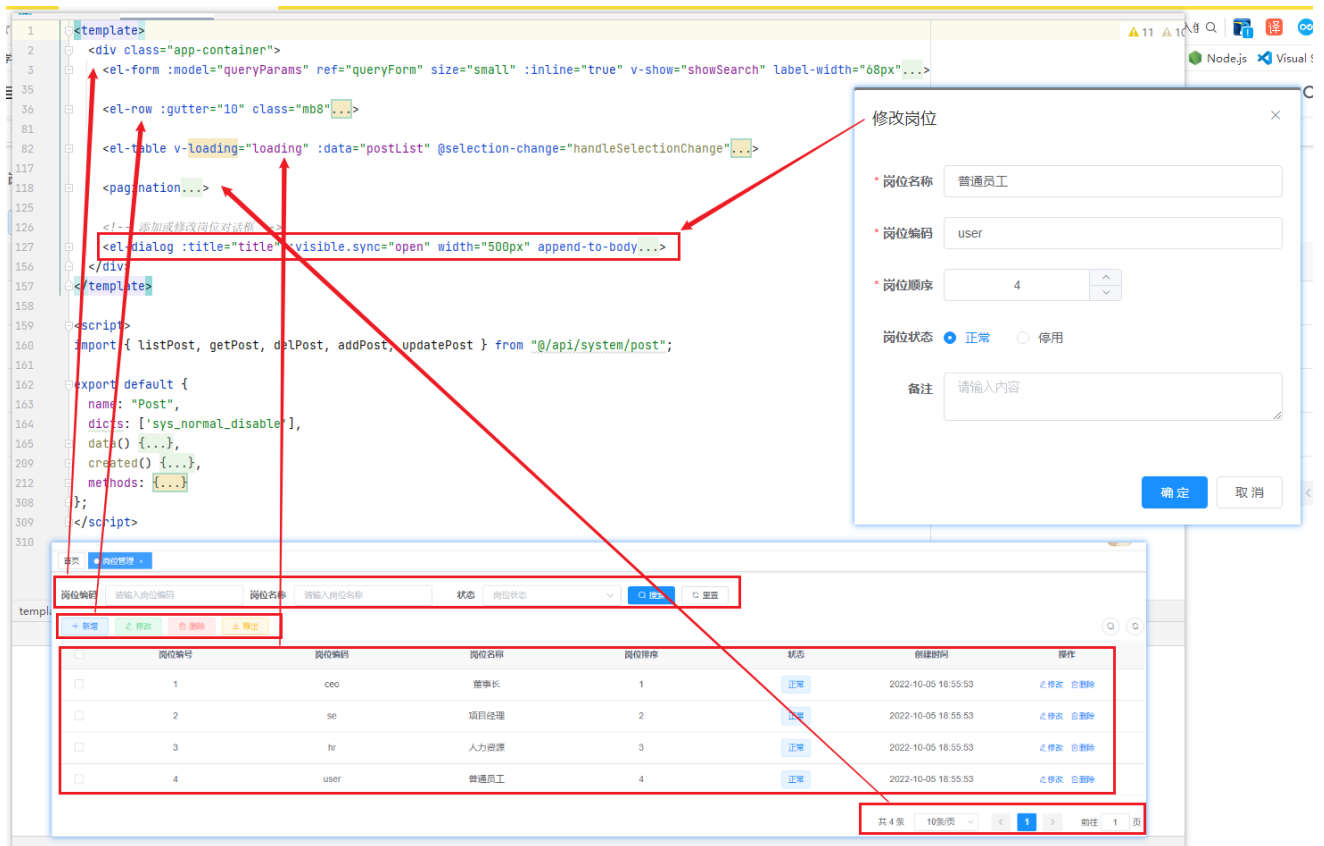
删除

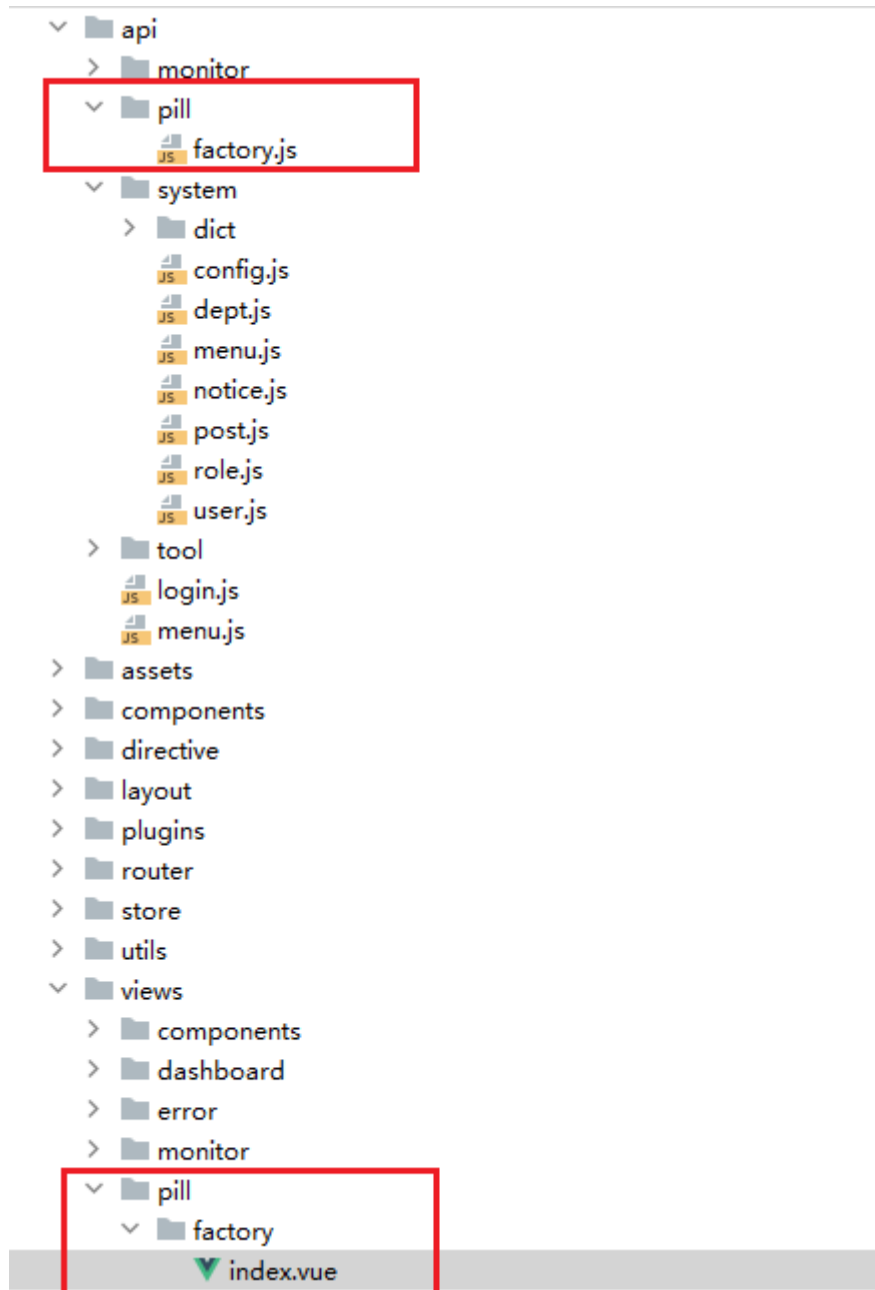
<input type="checkbox"/>	序号	表名称	表描述	实体	创建时间	更新时间	操作
<input type="checkbox"/>	1	pill_factory	生产厂家信息	PillFactory	2022-10-06 07:32:46		预览 编辑 删除 同步 生成代码

3.2 前端开发

3.2.1 常规开发-创建组件与脚本

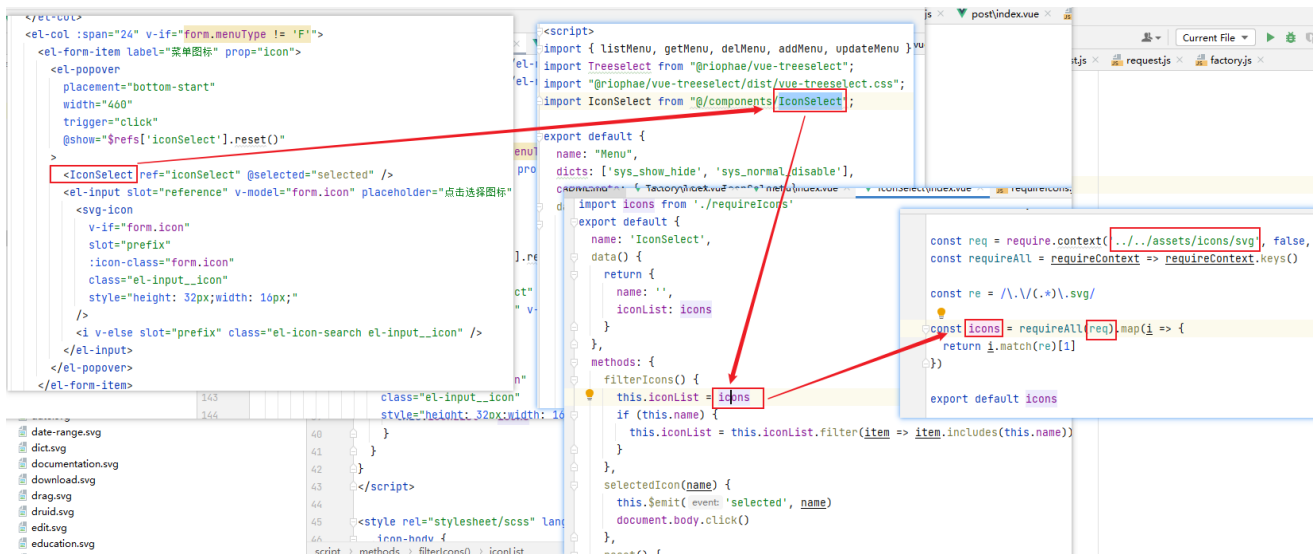
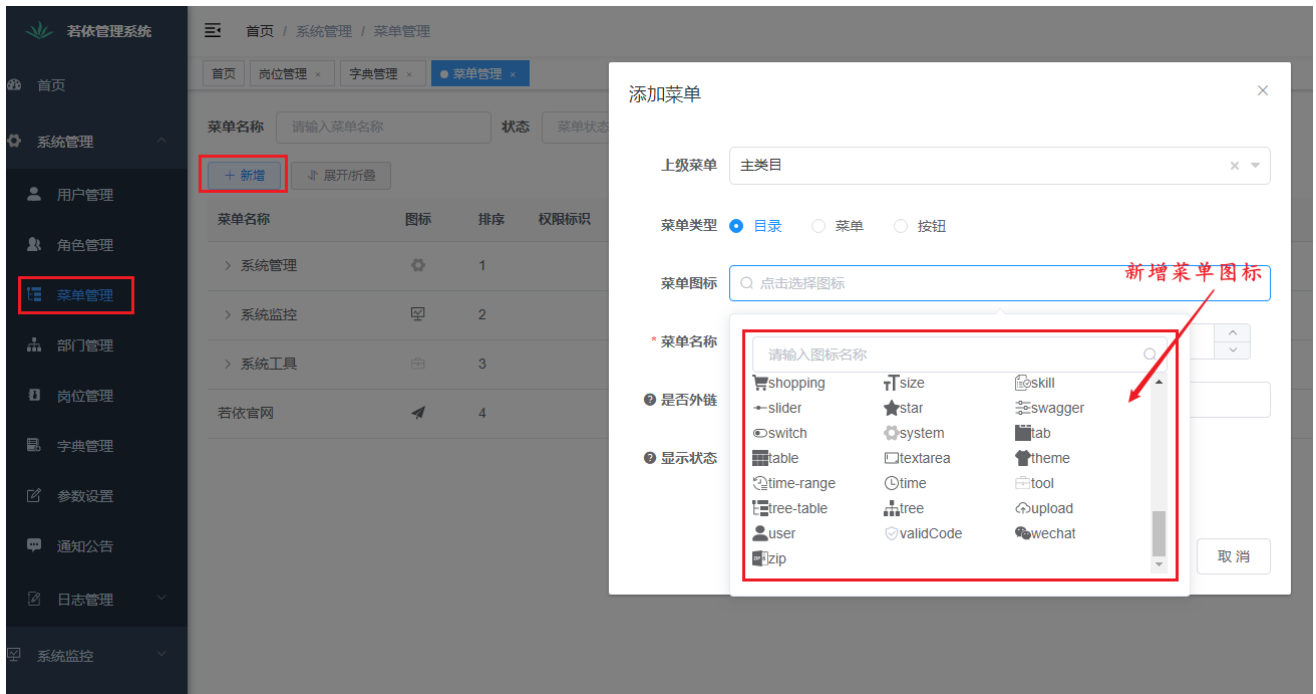
组件各部分详解

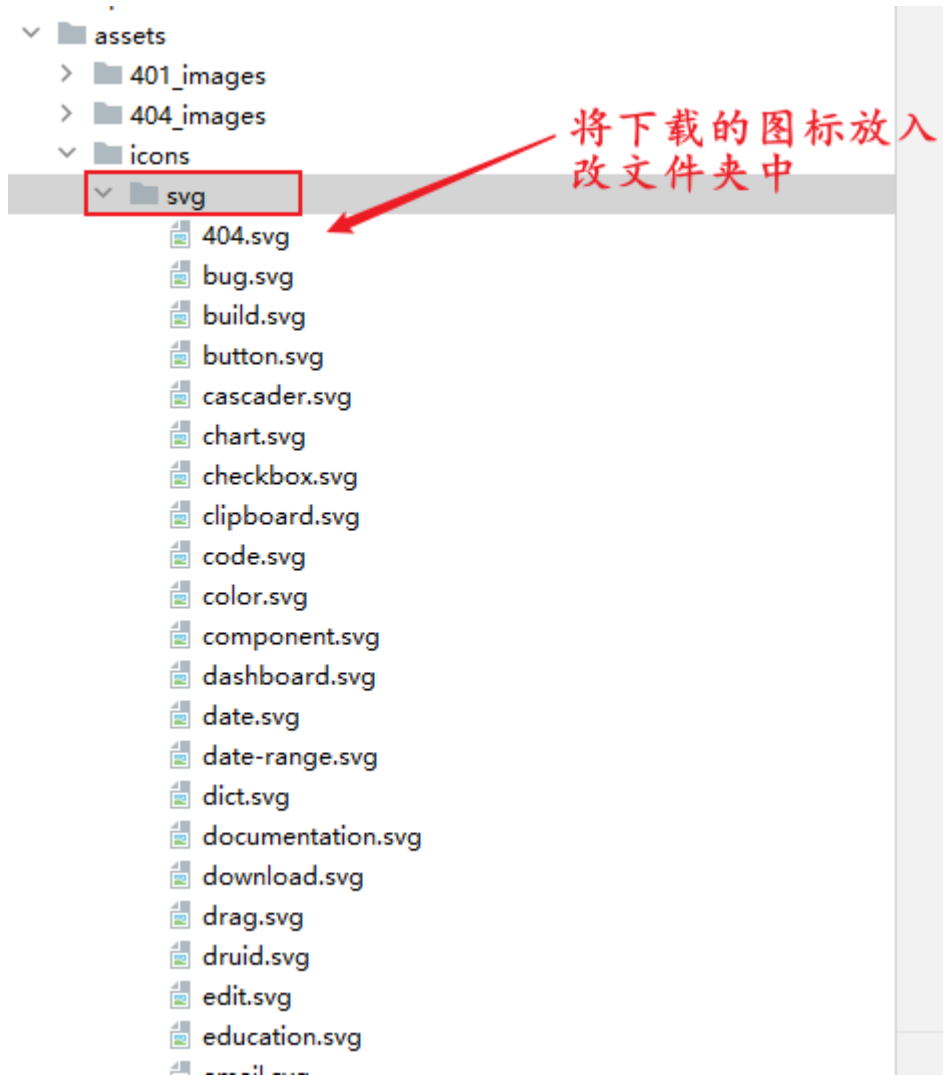




3.2.2 菜单创建-新增自定义图标

自定义图标





添加菜单

上级菜单 主类目

菜单类型 ☒ 目录 ☐ 菜单 ☐ 按钮

菜单图标 Q 点击选择图标

* 菜单名称

是否外链

显示状态

请输入图标名称

link	list	lock
log	loginfor	message
money	monitor	nested
number	online	password
pdf	people	peoples
phone	pill	post
qq	question	radio
rate	redis-list	redis

3.2.3 菜单创建-菜单栏目

删除广告菜单

若依管理系统

首页

系统管理

系统监控

系统工具

若依官网

首页

菜单管理

领取阿里云通用云产品1888优惠券
<https://www.aliyun.com/minisite/goods?userCode=brki8iof>
领取腾讯云通用云产品2860优惠券
https://cloud.tencent.com/redirect.php?redirect=1025&cps_key=198c8df2ed259157187173bc
阿里云服务器折扣区 > 点我进入 腾讯云服务器秒杀区 > 点我进入
云产品通用红包，可叠加官网常规优惠使用。(仅限新用户)

删除广告菜单

若依后台管理框架

一直想做一款后台管理系统，看了很多优秀的开源项目但是发现没有合适自己的。于是利用空闲休息时间开始若依管理系统，她可以用于所有的Web应用程序，如网站管理后台，网站会员中心，CMS，CRM，OA等等，制，以做出更强系统。所有前后台代码封装过后十分精简易上手，出错概率低。同时支持移动客户端访问。

当前版本: v3.8.4

若依管理系统

首页 / 系统管理 / 菜单管理

菜单管理

菜单名称: 请输入菜单名称 状态: 菜单状态 搜索 重置

+ 新增 展开所有

菜单名称	图标	排序	权限标识	组件路径	状态	创建时间	操作
> 系统管理		1			正常	2022-10-05 18:55:53	修改 新增 删除
> 系统监控		2			正常	2022-10-05 18:55:53	修改 新增 删除
> 系统工具		3			正常	2022-10-05 18:55:53	修改 新增 删除
若依官网		4			正常	2022-10-05 18:55:53	修改 新增 删除

localhost:3306

2020manufacture

bookstore

bookstore2

db_pss

表

gen_table

gen_table_column

pill_drug

pill_factory

qrtz_blob_triggers

qrtz_calendars

qrtz_cron_triggers

qrtz_fired_triggers

qrtz_job_details

qrtz_locks

qrtz_paused_trigger_grps

qrtz_scheduler_state

qrtz_simple_triggers

qrtz_simprop_triggers

qrtz_triggers

sys_config

sys_dept

sys_dict_data

sys_dict_type

sys_job

sys_job_log

sys_logininfor

sys_menu

sys_notice

sys_oper_log

sys_post

sys_role

sys_role_dept

sys_role_menu

sys_user

sys_user_post

sys_user_role

对象 sys_post @db_pss (localhos... sys_user @db_pss (localhos... pill_factory @db_pss (localh... pill_drug @db_pss (localhos... sys_menu @db_pss (localho...

开始事务 备注 筛选 排序 导入 导出

menu_id	menu_name	parent_id	order_num	path	component	query	is_frame	is_cache	menu_type	visible	status	perms
1	系统管理	0	1	system	(Null)		1	0	M	0	0	
2	系统监控	0	2	monitor	(Null)		1	0	M	0	0	
3	系统工具	0	3	tool	(Null)		1	0	M	0	0	
4	若依官网	0	4	http://ruoyi.vip	(Null)		0	0	M	0	0	
100	用户管理	1	1	user	system/user/index		1	0	C	0	0	system:user
101	角色管理	1	2	role	system/role/index		1	0	C	0	0	system:role
102	菜单管理	1	3	menu	system/menu/index		1	0	C	0	0	system:menu
103	部门管理	1	4	dept	system/dept/index		1	0	C	0	0	system:dept
104	岗位管理	1	5	post	system/post/index		1	0	C	0	0	system:post
105	字典管理	1	6	dict	system/dict/index		1	0	C	0	0	system:dict
106	参数设置	1	7	config	system/config/index		1	0	C	0	0	system:config
107	通知公告	1	8	notice	system/notice/index		1	0	C	0	0	system:notice
108	日志管理	1	9	log			1	0	M	0	0	
109	在线用户	2	1	online	monitor/online/index		1	0	C	0	0	monitor:online
110	定时任务	2	2	job	monitor/job/index		1	0	C	0	0	monitor:job
111	数据监控	2	3	druid	monitor/druid/index		1	0	C	0	0	monitor:druid
112	服务器监控	2	4	server	monitor/server/index		1	0	C	0	0	monitor:server
113	缓存监控	2	5	cache	monitor/cache/index		1	0	C	0	0	monitor:cache
114	缓存列表	2	6	cacheList	monitor/cache/list		1	0	C	0	0	monitor:cacheList
115	表单构建	3	1	build	tool/build/index		1	0	C	0	0	tool:build
116	代码生成	3	2	gen	tool/gen/index		1	0	C	0	0	tool:gen
117	系统接口	3	3	swagger	tool/swagger/index		1	0	C	0	0	tool:swagger
500	操作日志	108	1	operlog	monitor/operlog/index		1	0	C	0	0	monitor:operlog
501	登录日志	108	2	logininfor	monitor/logininfor/index		1	0	C	0	0	monitor:logininfor
1000	用户查询	100	1				1	0	F	0	0	system:user
1001	用户新增	100	2				1	0	F	0	0	system:user
1002	用户修改	100	3				1	0	F	0	0	system:user
1003	用户删除	100	4				1	0	F	0	0	system:user

添加菜单

×

上级菜单

主类目

×

▼

菜单类型

☒ 目录

☐ 菜单

☐ 按钮

菜单图标

 pill

* 菜单名称

药品进销存

* 显示排序

4

^

▼

🔍 是否外链

☐ 是

☒ 否

* 🔍 路由地址

pill

🔍 显示状态

☒ 显示

☐ 隐藏

🔍 菜单状态

☒ 正常

☐ 停用

确定

取消

添加菜单

×

上级菜单

药品进销存

×

▼

菜单类型

☐ 目录

☒ 菜单

☐ 按钮

菜单图标

🏠

build

* 菜单名称

生产厂家

* 显示排序

1

^

▼

❓ 是否外链

☐ 是

☒ 否

* ❓ 路由地址

factory

❓ 组件路径

pill/factory/index

❓ 权限字符

pill:factory:list

❓ 路由参数

请输入路由参数

❓ 是否缓存

☒ 缓存

☐ 不缓存

❓ 显示状态

☒ 显示

☐ 隐藏

❓ 菜单状态

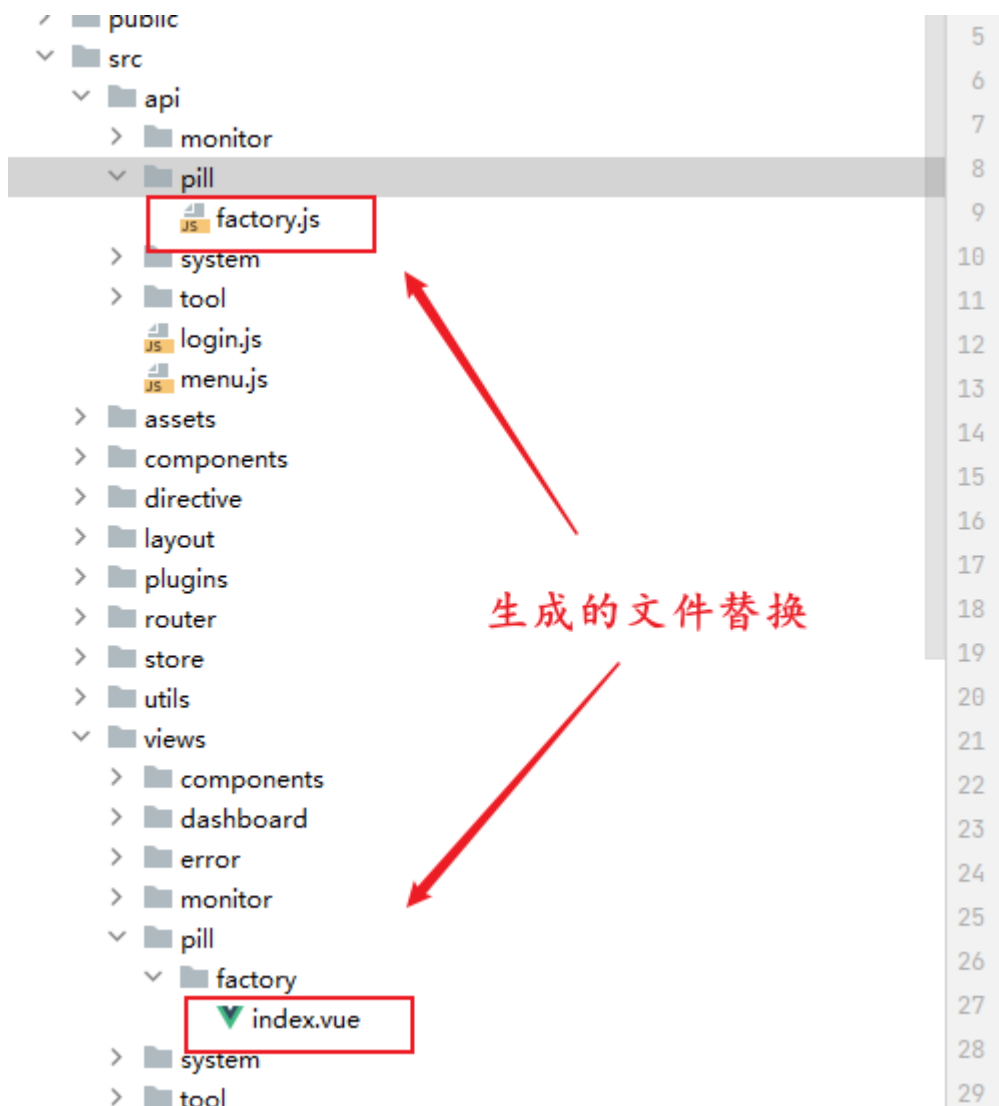
☒ 正常

☐ 停用

确定

取消

3.2.4 自动代码生成模式开发



调整细节页面

```
<el-table v-loading="loading" :data="factoryList" @selection-change="handleSelectionChange">
  <el-table-column type="selection" width="55" align="center" />
  <el-table-column label="厂家ID" align="center" prop="factoryId" width="80" />
  <el-table-column label="厂家名称" align="center" prop="factoryName" min-width="200" />
  <el-table-column label="厂家编码" align="center" prop="factoryCode" />
  <el-table-column label="联系人" align="center" prop="contact" />
  <el-table-column label="电话" align="center" prop="phone" />
  <el-table-column label="关键字" align="center" prop="keyword" />
  <el-table-column label="状态" align="center" prop="status" />
  <el-table-column label="创建时间" align="center" prop="createTime" min-width="150" />
  <el-table-column label="操作" align="center" class-name="small-padding fixed-width">
    <template slot-scope="scope">
      <el-button
        size="mini"
        type="text"
        icon="el-icon-edit"
        @click="handleUpdate(scope.row)"
      />
    </template>
  </el-table-column>
</el-table>
```

```

        placeholder="请输入岗位名称"
        clearable
        @keyup.enter.native="handleQuery"
      />
    </el-form-item>
    <el-form-item label="状态" prop="status">
      <el-select v-model="queryParams.status" placeholder="岗位状态" clearable>
        <el-option
          v-for="dict in dict.type.sys_normal_disable"
          :key="dict.value"
          :label="dict.label"
          :value="dict.value"
        />
      </el-select>
    </el-form-item>
    <el-form-item>
      <el-button type="primary" icon="el-icon-search" size="mini" @click="handleQuery">搜索</el-button>
      <el-button icon="el-icon-refresh" size="mini" @click="resetQuery">重置</el-button>
    </el-form-item>
  </el-form>

```

ost\index.vue

```

    <el-table-column label="岗位编码" align="center" prop="postCode" />
    <el-table-column label="岗位名称" align="center" prop="postName" />
    <el-table-column label="岗位排序" align="center" prop="postSort" />
    <el-table-column label="状态" align="center" prop="status">
      <template slot-scope="scope">
        <dict-tag :options="dict.type.sys_normal_disable" :value="scope.row.status"/>
      </template>
    </el-table-column>
    <el-table-column label="创建时间" align="center" prop="createTime" width="180">
      <template slot-scope="scope">
        <span>{{ parseTime(scope.row.createTime) }}</span>
      </template>
    </el-table-column>
    <el-table-column label="操作" align="center" class-name="small-padding fixed-width">
  </el-table>

  <pagination...>

  <!-- 添加或修改岗位对话框 -->
  <el-dialog :title="title" :visible.sync="open" width="500px" append-to-body...>
  </div>
</template>

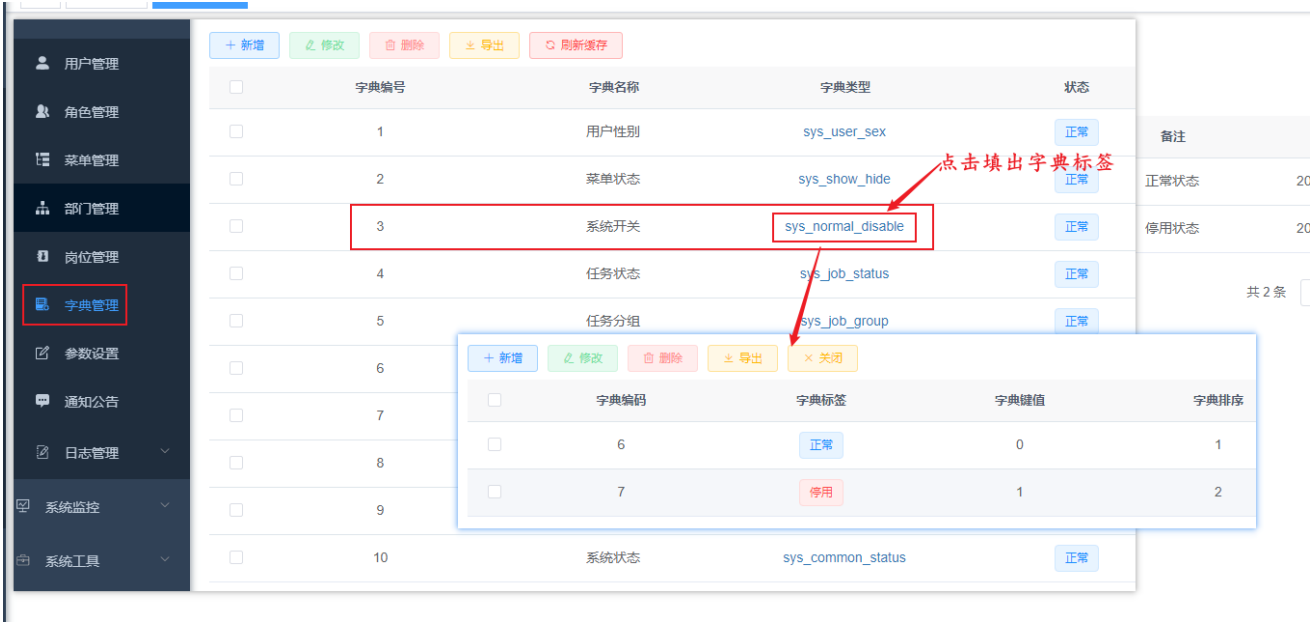
<script>
import { listPost, getPost, delPost, addPost, updatePost } from '@api/system/post';

export default {
  name: "Post",
  dicts: ['sys_normal_disable'],

```

相关代码

```
1 | dicts: ['sys_normal_disable'],
```



```
1 | <el-form-item label="状态" prop="status">
2 |   <el-select v-model="queryParams.status" placeholder="岗位状态" clearable>
3 |     <el-option
4 |       v-for="dict in dict.type.sys_normal_disable"
5 |       :key="dict.value"
6 |       :label="dict.label"
7 |       :value="dict.value"
8 |     />
9 |   </el-select>
10 | </el-form-item>
```

```
1 | <el-table-column label="状态" align="center" prop="status">
2 |   <template slot-scope="scope">
3 |     <dict-tag :options="dict.type.sys_normal_disable" :value="scope.row.status"/>
4 |   </template>
5 | </el-table-column>
```

3.2.5 增删改查功能调试

查询功能

```

<select id="selectPillFactoryList" parameterType="PillFactory" resultMap="PillFactoryResult">
  <include refid="selectPillFactoryVo"/>
  <where>
    <if test="factoryName != null and factoryName != ''"> and factory_name like concat('%', #{factoryName}, '%')</if>
    <if test="factoryCode != null and factoryCode != ''"> and factory_code = #{factoryCode}</if>
    <if test="contact != null and contact != ''"> and contact = #{contact}</if>
    <if test="phone != null and phone != ''"> and phone = #{phone}</if>
    <if test="keyword != null and keyword != ''"> and keyword = #{keyword}</if>
    <if test="status != null and status != ''"> and status = #{status}</if>
  </where>
</select>

```

并不是模糊查询

后端代码修改

```

1  <select id="selectPillFactoryList" parameterType="PillFactory"
2    resultMap="PillFactoryResult">
3    <include refid="selectPillFactoryVo"/>
4    <where>
5      <if test="factoryName != null and factoryName != ''"> and factory_name like
6        concat('%', #{factoryName}, '%')</if>
7      <if test="factoryCode != null and factoryCode != ''"> and factory_code = #
8        {factoryCode}</if>
9      <if test="contact != null and contact != ''"> and contact = #{contact}</if>
10     <if test="phone != null and phone != ''"> and phone like concat('%', #{phone}, '%')
11     </if>
12     <if test="keyword != null and keyword != ''"> and keyword like concat('%', #
13     {keyword}, '%')</if>
14     <if test="status != null and status != ''"> and status = #{status}</if>
15   </where>
16 </select>

```

新增功能

添加生产厂家信息

必选项验证

厂家名称

请输入厂家名称

厂家编码

请输入厂家编码

联系人

请输入联系人

电话

请输入电话

关键字

请输入关键字

缺一个状态单选

备注

请输入内容

确定

取消

相关代码

验证代码

```
1 // 表单校验
2 rules: {
3   factoryName: [
4     { required: true, message: "生产厂家名称不能为空", trigger: "blur" }
5   ],
6   factoryCode: [
7     { required: true, message: "生产厂家编码不能为空", trigger: "blur" }
8   ]
9 }
```

状态单选

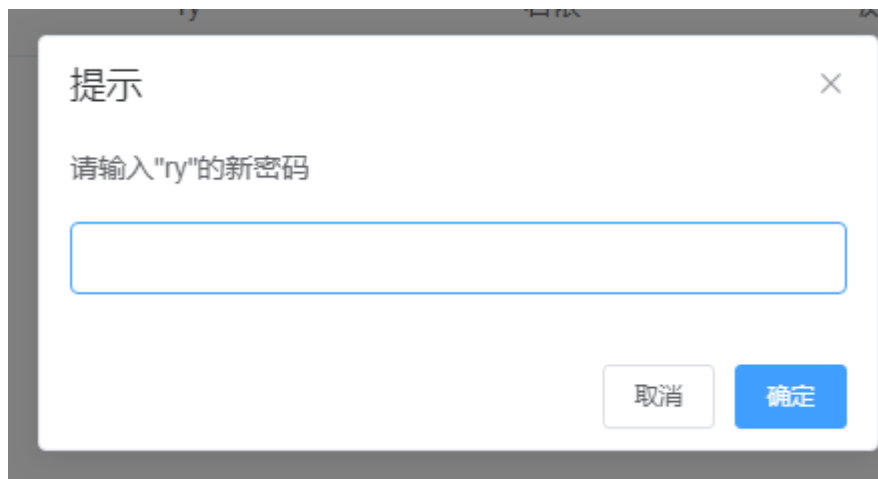
```
1 <el-form-item label="生产厂家状态" prop="status">
2   <el-radio-group v-model="form.status">
3     <el-radio
4       v-for="dict in dict.type.sys_normal_disable"
5       :key="dict.value"
6       :label="dict.value"
7     >{{dict.label}}</el-radio>
8   </el-radio-group>
9 </el-form-item>
```

3.2.6 功能权限分配

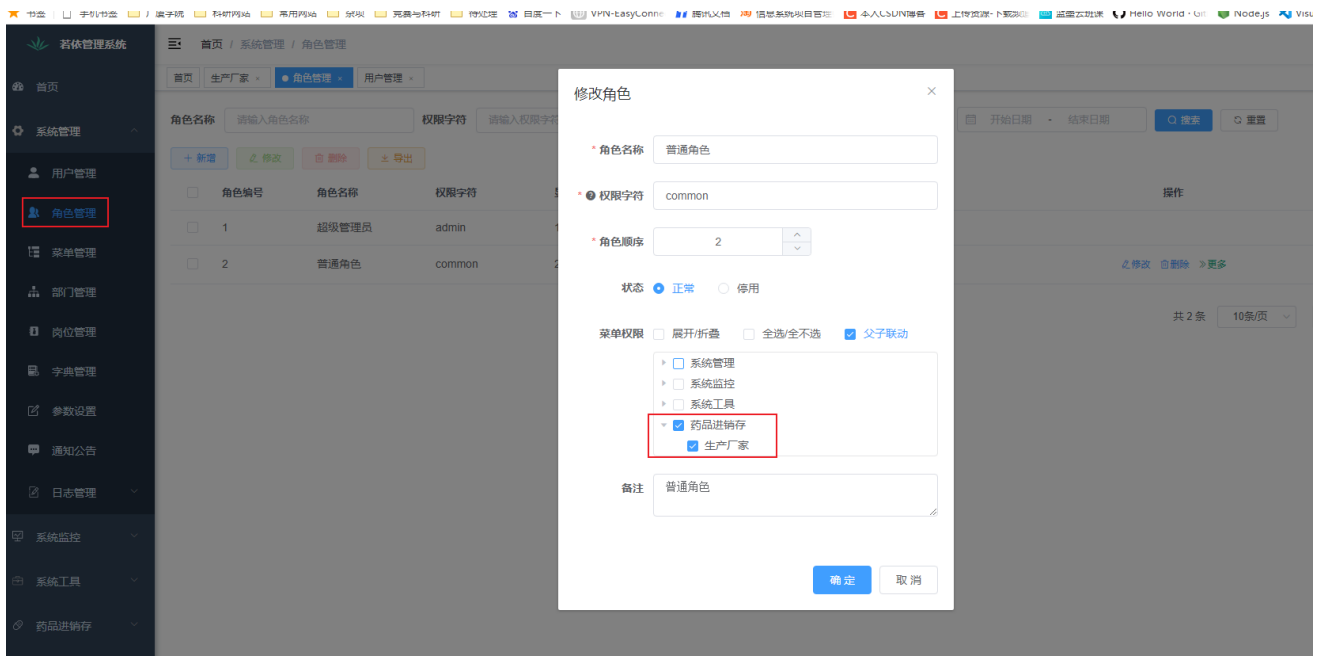
用户管理



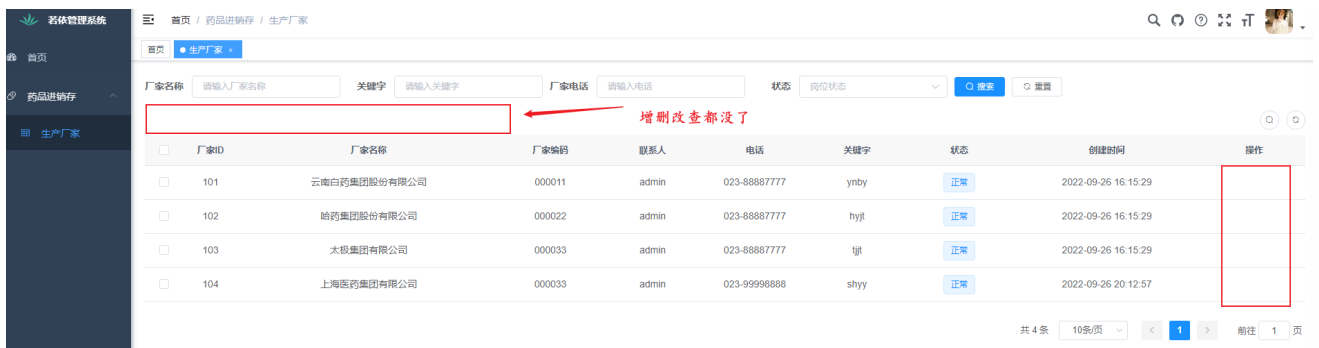
重置密码



角色管理



换成新用户登录

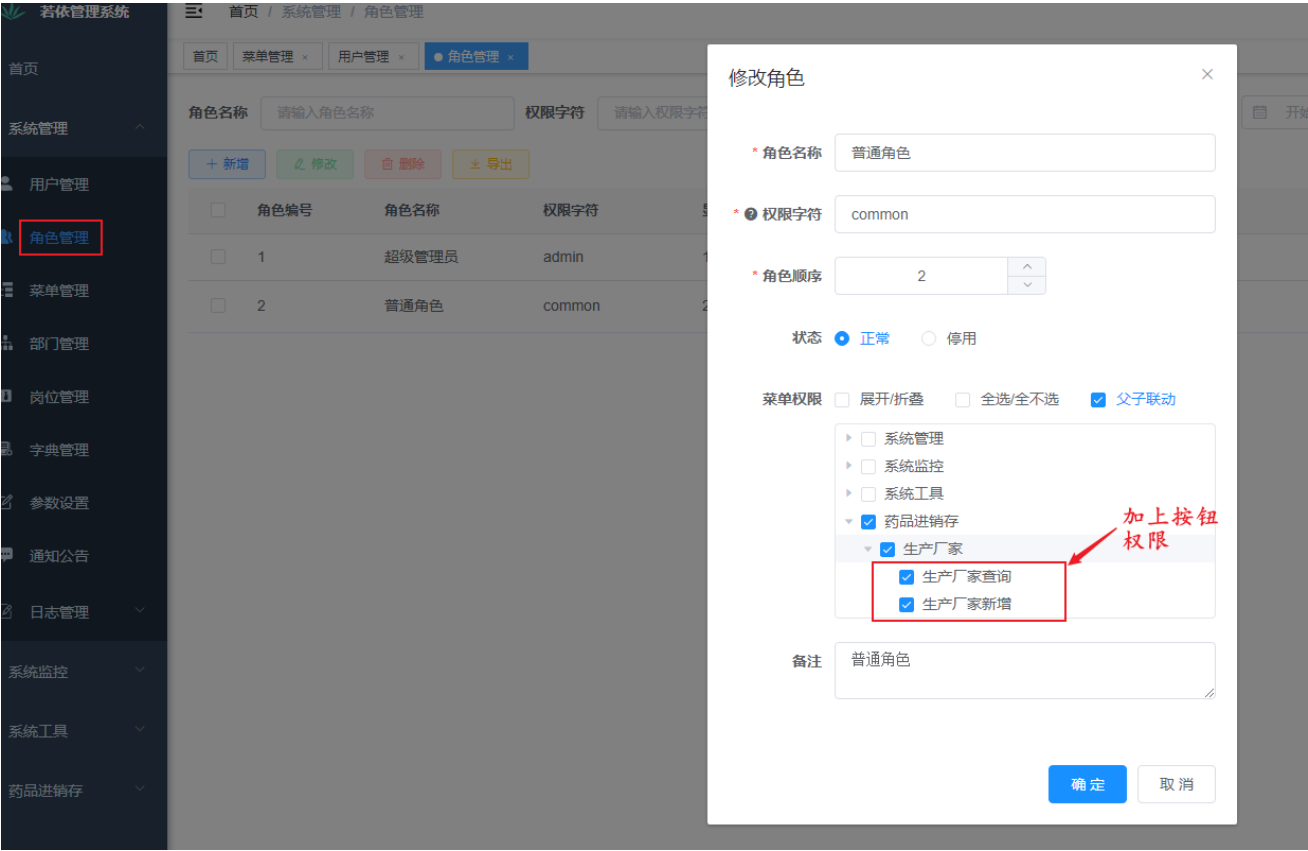


参考岗位权限设置

岗位管理	5	system.post.list	system/post/index	正常	2022-10-05 18:55:53	修改 新增 删除
岗位查询	1	system.post.query		正常	2022-10-05 18:55:53	修改 新增 删除
岗位新增	2	system.post.add		正常	2022-10-05 18:55:53	修改 新增 删除
岗位修改	3	system.post.edit		正常	2022-10-05 18:55:53	修改 新增 删除
岗位删除	4	system.post.remove		正常	2022-10-05 18:55:53	修改 新增 删除
岗位导出	5	system.post.export		正常	2022-10-05 18:55:53	修改 新增 删除

设置生产厂家按钮权限

药品进销存	4			正常	2022-10-06 09:50:36	修改 新增 删除
生产厂家	1	pill.factory.list	pill/factory/index	正常	2022-10-06 09:57:28	修改 新增 删除
生产厂家查询	1	pill.factory.query		正常	2022-10-06 15:14:03	修改 新增 删除
生产厂家新增	2	pill.factory.add		正常	2022-10-06 15:14:49	修改 新增 删除

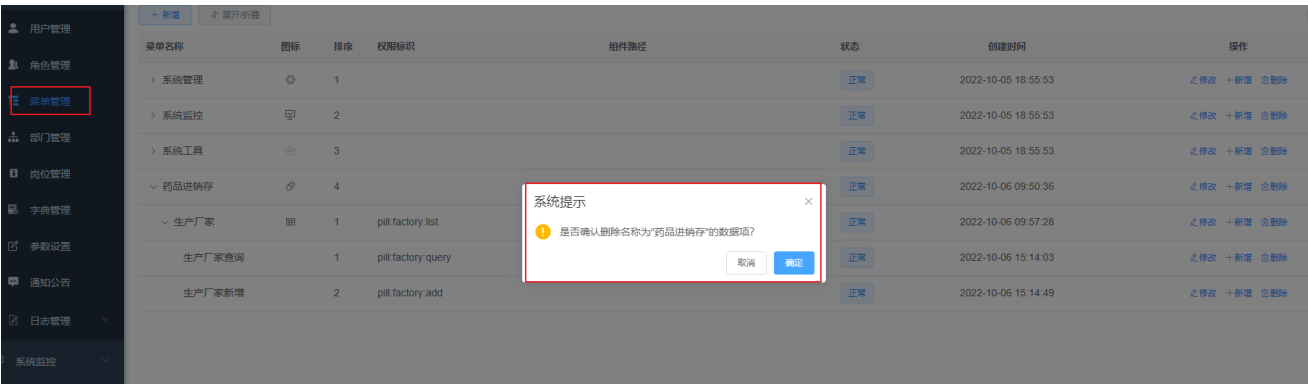


换账号登录

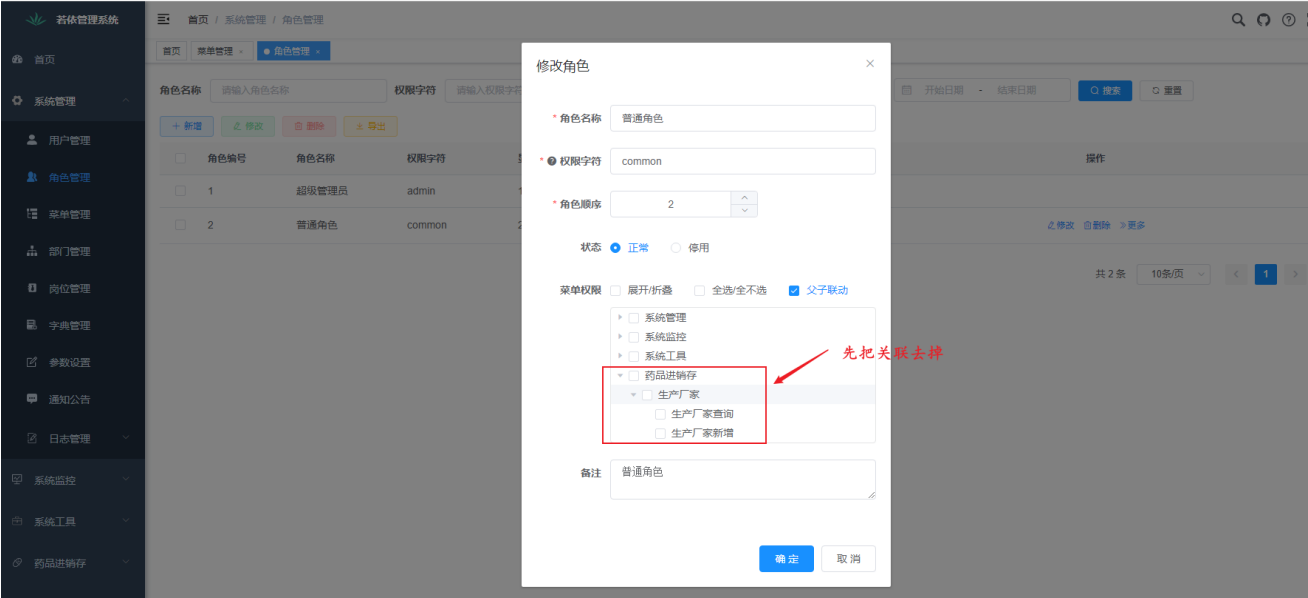


3.2.7 脚本实现权限分配

先删除原先的菜单



删除失败，先删除关联



名称	修改日期	类型
main	2022/10/6 7:35	文件夹
vue	2022/10/6 7:35	文件夹
factoryMenu.sql	2022/10/6 7:34	SQL 文件

运行脚本后

系统工具	📁	3			正常	2022-10-05 18:55:53	🔗修改 +新增 🗑删除
表单构建	📄	1	tool/build:list	tool/build/index	正常	2022-10-05 18:55:53	🔗修改 +新增 🗑删除
生产厂商信息		1	pill/factory:list	pill/factory/index	正常	2022-10-06 15:25:40	🔗修改 +新增 🗑删除
代码生成	🔗	2	tool/gen:list	tool/gen/index	正常	2022-10-05 18:55:53	🔗修改 +新增 🗑删除
系统接口	📄	3	tool/swagger:list	tool/swagger/index	正常	2022-10-05 18:55:53	🔗修改 +新增 🗑删除
药品进销存	📄	4			正常	2022-10-06 09:50:36	🔗修改 +新增 🗑删除

调整菜单

药品进销存		4			正常	2022-10-06 09:50:36	修改 新增 删除
生产厂家		1	pill:factory.list	pill:factory/index	正常	2022-10-06 15:25:40	修改 新增 删除
生产厂家信...		1	pill:factory.query		正常	2022-10-06 15:25:40	修改 新增 删除
生产厂家信...		2	pill:factory.add		正常	2022-10-06 15:25:40	修改 新增 删除
生产厂家信...		3	pill:factory.edit		正常	2022-10-06 15:25:40	修改 新增 删除
生产厂家信...		4	pill:factory.remove		正常	2022-10-06 15:25:40	修改 新增 删除
生产厂家信...		5	pill:factory.export		正常	2022-10-06 15:25:40	修改 新增 删除

修改角色权限

若依管理系统

首页

系统管理

角色管理

用户管理

菜单管理

部门管理

岗位管理

字典管理

参数设置

通知公告

日志管理

系统监控

系统工具

药品进销存

首页 / 系统管理 / 角色管理

角色名称 请输入角色名称 权限字符 请输入权限字符

[+ 新增](#) [修改](#) [删除](#) [导出](#)

<input type="checkbox"/>	角色编号	角色名称	权限字符
<input type="checkbox"/>	1	超级管理员	admin
<input type="checkbox"/>	2	普通角色	common

修改角色

* 角色名称 普通角色

* 权限字符 common

* 角色顺序 2

状态 ☒ 正常 ☐ 停用

菜单权限 ☐ 展开/折叠 ☐ 全选/全不选 ☒ 父子联动

☐ 系统管理

☐ 系统监控

☐ 系统工具

☒ 药品进销存

☒ 生产厂家

☒ 生产厂家信息查询

☒ 生产厂家信息新增

☒ 生产厂家信息修改

☒ 生产厂家信息删除

☒ 生产厂家信息导出

备注 普通角色

[确定](#) [取消](#)

切换角色

若依管理系统

首页

药品进销存

生产厂家

首页 / 药品进销存 / 生产厂家

生产厂家 请输入厂家名称 关键字 请输入关键字 厂家电话 请输入电话 状态 岗位状态 [搜索](#) [重置](#)

[+ 新增](#) [修改](#) [删除](#) [导出](#)

<input type="checkbox"/>	厂家ID	厂家名称	厂家编码	联系人	电话	关键字	状态	创建时间	操作
<input type="checkbox"/>	101	云南白药集团股份有限公司	000011	admin	023-88887777	ynby	正常	2022-09-26 16:15:29	修改 删除
<input type="checkbox"/>	102	哈药集团股份有限公司	000022	admin	023-88887777	hyjt	正常	2022-09-26 16:15:29	修改 删除
<input type="checkbox"/>	103	太极集团有限公司	000033	admin	023-88887777	tj	正常	2022-09-26 16:15:29	修改 删除
<input type="checkbox"/>	104	上海医药集团有限公司	000033	admin	023-99998888	shyy	正常	2022-09-26 20:12:57	修改 删除

共 4 条 10条/页 [1](#) 前往 1 页

4-模块开发-药品信息

4.1 字典管理

分析药品信息中 各类型状态

首页 / 药品进销存管理 / 药品信息维护

生产厂家维护 药品信息维护

药品名称 请输入药品名称 药品类型 请输入药品类型 生产厂家 请输入生产厂家 处方类型 请输入处方类型 状态 药品状态

搜索 重置

+ 新增 修改 删除 导出

<input type="checkbox"/>	药品ID	药品名称	药品编码	生产厂家	药品类型	处方类型	单位	价格	库存量	预置值	换算量	状态	操作
<input type="checkbox"/>	101	白果	zcy0001	云南白药集团股份有限公司	中草药	中药处方	g	2	200	100	1	正常	修改 删除
<input type="checkbox"/>	102	白英	zcy0002	云南白药集团股份有限公司	中草药	中药处方	g	2	200	100	1	正常	修改 删除

共 2 条 10条/页 < 1 > 前往 1 页

新增字典类型

修改字典类型

* 字典名称

药品类型

* 字典类型

pill_drug_dtype

状态

正常

停用

备注

药品类型列表

确定

取消

添加字典数据



字典类型

pill_drug_dtype

* 数据标签

中草药

* 数据键值

0

样式属性

请输入样式属性

* 显示排序

1



回显样式

默认(default)



状态



正常



停用

备注

请输入内容

确定

取消

添加字典数据



字典类型 pill_drug_dtype

* 数据标签 西药

* 数据键值 1

样式属性 请输入样式属性

* 显示排序 2

回显样式 默认(default)

状态 ☒ 正常 ☐ 停用

备注 请输入内容

确定

取消

添加字典类型



* 字典名称

处方类型

* 字典类型

pill_drug_ptype

状态 ☒ 正常 ☐ 停用

备注

处方类型列表

确定

取消

添加字典数据

×

字典类型

pill_drug_ptype

* 数据标签

中药处方

* 数据键值

0

样式属性

请输入样式属性

* 显示排序

1

回显样式

默认(default)

状态



正常



停用

备注

请输入内容

确定

取消

4.2 代码生成器详细属性设置

根据原型设计页面

基本信息

字段信息

生成信息

序号	字段列名	字段描述	物理类型	Java类型	java属性	插入	编辑	列表	查询	查询方式	必填	显示类型	字典类型
1	drug_id	药品ID	bigint(20)	Long	drugId	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	=	<input type="checkbox"/>	文本框	请选择
2	drug_name	药品名称	varchar(50)	String	drugName	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	LIKE	<input checked="" type="checkbox"/>	文本框	请选择
3	drug_code	药品编码	varchar(50)	String	drugCode	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	=	<input checked="" type="checkbox"/>	文本框	请选择
4	factory_id	生产厂家	bigint(20)	Long	factoryId	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	LIKE	<input type="checkbox"/>	文本框	请选择
5	drug_type	药品类型	char(1)	String	drugType	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	=	<input type="checkbox"/>	下拉框	药品类型
6	prescription_type	处方类型	char(1)	String	prescriptionTyp	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	=	<input type="checkbox"/>	下拉框	处方类型
7	unit	单位	varchar(10)	String	unit	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	=	<input type="checkbox"/>	文本框	请选择
8	price	价格	int(10)	Integer	price	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	=	<input type="checkbox"/>	文本框	请选择
9	num	库存量	int(10)	Integer	num	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	=	<input type="checkbox"/>	文本框	请选择
10	warn_value	预警值	int(10)	Integer	warnValue	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	=	<input type="checkbox"/>	文本框	请选择
11	conversion	换算量	int(10)	Integer	conversion	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	=	<input type="checkbox"/>	文本框	请选择

提交

返回

代码生成

修改[pill_drug]生成配置

基本信息

字段信息

生成信息

* 生成模板

单表 (增删改查)

* 生成模块名

pill

* 生成功能名

药品信息

* 生成包路径

com.ruoyi.pill

* 生成业务名

drug

上级菜单

药品进销存

生成代码方式

☒ zip压缩包 ☐ 自定义路径

提交

返回

4.3 全智能模块开发

生产代码

替换后端 前端，运行菜单SQL脚本

4.4 查询调整-多表链接

后端代码修改

PillDrug


```

1 private PillFactory pillFactory;
2
3 public PillFactory getPillFactory() {
4     return pillFactory;
5 }
6
7 public void setPillFactory(PillFactory pillFactory) {
8     this.pillFactory = pillFactory;
9 }
10
11 ...
12 .append("pillFactory", getPillFactory())

```

PillDrugMapper.xml

```

1 <resultMap type="PillDrug" id="PillDrugResult">
2     <result property="drugId" column="drug_id" />
3     <result property="drugName" column="drug_name" />
4     <result property="drugCode" column="drug_code" />
5     <result property="factoryId" column="factory_id" />
6     <result property="drugType" column="drug_type" />
7     <result property="prescriptionType" column="prescription_type" />
8     <result property="unit" column="unit" />
9     <result property="price" column="price" />
10    <result property="num" column="num" />
11    <result property="warnValue" column="warn_value" />
12    <result property="conversion" column="conversion" />
13    <result property="status" column="status" />
14    <result property="createBy" column="create_by" />
15    <result property="createTime" column="create_time" />
16    <result property="updateBy" column="update_by" />
17    <result property="updateTime" column="update_time" />
18    <result property="remark" column="remark" />
19    <association property="pillFactory" javaType="pillFactory"
resultMap="com.ruoyi.pill.mapper.PillFactoryMapper.PillFactoryResult"></association>
20 </resultMap>
21
22 <sql id="selectPillDrugVo">
23     select pd.*,pf.* from pill_drug pd left join pill_factory pf on pd.factory_id =
pf.factory_id
24 </sql>
25
26 <select id="selectPillDrugList" parameterType="PillDrug" resultMap="PillDrugResult">
27     <include refid="selectPillDrugVo"/>
28     <where>
29         <if test="drugName != null and drugName != ''"> and pd.drug_name like
concat('%', #{drugName}, '%')</if>
30         <if test="factoryId != null "> and pd.factory_id like concat('%', #{factoryId},
'%')</if>
31         <if test="pillFactory != null and pillFactory.factoryName != null and

```

```

pillFactory.factoryName != ''">
32         and pf.factory_name like concat('%', #{pillFactory.factoryName}, '%')</if>
33         <if test="drugType != null and drugType != ''"> and pd.drug_type = #{drugType}
</if>
34         <if test="prescriptionType != null and prescriptionType != ''"> and
pd.prescription_type = #{prescriptionType}</if>
35         <if test="status != null and status != ''"> and pd.status = #{status}</if>
36     </where>
37 </select>
38
39 <select id="selectPillDrugByDrugId" parameterType="Long" resultMap="PillDrugResult">
40     <include refid="selectPillDrugVo"/>
41     where pd.drug_id = #{drugId}
42 </select>

```

测试代码

```

1  @Autowired
2  private IPillDrugService pillDrugService;
3
4  @Test
5  public void testSelectDrug(){
6      PillDrug pillDrug = new PillDrug();
7      PillFactory pillFactory = new PillFactory();
8      pillFactory.setFactoryName("哈药");
9      pillDrug.setPillFactory(pillFactory);
10     List<PillDrug> pillDrugs = pillDrugService.selectPillDrugList(pillDrug);
11     System.out.println(pillDrugs);
12 }

```

前端代码修改

queryParams

```

1  // 查询参数
2  queryParams: {
3      pageNum: 1,
4      pageSize: 10,
5      drugName: null,
6      factoryId: null,
7      'pillFactory.factoryName': null,
8      drugType: null,
9      prescriptionType: null,
10     status: null,
11 },

```

生产厂家文本框

```

1 <el-form-item label="生产厂家" >
2   <el-input
3     v-model="queryParams['pillFactory.factoryName']"
4     placeholder="请输入生产厂家"
5     clearable
6     @keyup.enter.native="handleQuery"
7   />
8 </el-form-item>

```

数据列表

```

1 <el-table v-loading="loading" :data="drugList" @selection-change="handleSelectionChange">
2   <el-table-column type="selection" width="55" align="center" />
3   <el-table-column label="药品ID" align="center" prop="drugId" />
4   <el-table-column label="药品名称" align="center" prop="drugName" />
5   <el-table-column label="药品编码" align="center" prop="drugCode" />
6   <el-table-column label="生产厂家" align="center" prop="pillFactory.factoryName" />
7   <el-table-column label="厂家编码" align="center" prop="pillFactory.factoryCode" />

```

页面列表调整

```

<el-table-column type="selection" width="55" align="center" />
<el-table-column label="药品ID" align="center" prop="drugId" width="80" />
<el-table-column label="药品名称" align="center" prop="drugName" min-width="150" />
<el-table-column label="药品编码" align="center" prop="drugCode" />
<el-table-column label="生产厂家" align="center" prop="pillFactory.factoryName" min-width="200" />
<el-table-column label="厂家编码" align="center" prop="pillFactory.factoryCode" />
<el-table-column label="药品类型" align="center" prop="drugType">
  <template slot-scope="scope">
    <dict-tag :options="dict.type.pill_drug_dtype" :value="scope.row.drugType"/>
  </template>
</el-table-column>
<el-table-column label="处方类型" align="center" prop="prescriptionType">
  <template slot-scope="scope">
    <dict-tag :options="dict.type.pill_drug_ptype" :value="scope.row.prescriptionType"/>
  </template>
</el-table-column>
<el-table-column label="单位" align="center" prop="unit" />
<el-table-column label="价格" align="center" prop="price" />
<el-table-column label="库存量" align="center" prop="num" />
<el-table-column label="预警值" align="center" prop="warnValue" />
<el-table-column label="换算量" align="center" prop="conversion" />
<el-table-column label="状态" align="center" prop="status">
  <template slot-scope="scope">
    <dict-tag :options="dict.type.sys_normal_disable" :value="scope.row.status"/>
  </template>
</el-table-column>
<el-table-column label="创建时间" align="center" prop="createTime" width="180">
  <template slot-scope="scope">
    <span>{{ parseTime(scope.row.createTime, '{y}-{m}-{d}') }}</span>
  </template>
</el-table-column>
<el-table-column label="操作" align="center" class-name="small-padding fixed-width" min-width="100">
  <template slot-scope="scope">
    <el-button

```

4.5 生产厂家下拉框实现

代码

```
1 <el-form-item label="生产厂家" prop="factoryId">
2   <el-select v-model="form.factoryId" placeholder="请输入生产厂家" clearable>
3     <el-option
4       v-for="item in factoryList"
5       :key="item.factoryId"
6       :value="item.factoryId"
7       :label="item.factoryName"
8     />
9   </el-select>
10 </el-form-item>
```

```
1 // 生产厂家数据
2 factoryList: [],
```

```
1 import {listFactory} from "@/api/pill/factory";
```

```
1 getFactoryList(){
2   listFactory().then(response => {
3     this.factoryList = response.rows;
4   });
5 },
```

```
1 created() {
2   this.getList();
3   this.getFactoryList();
4 },
```