吉运Java开发平台3.0使用说明

# 1. 平台说明

1.1 声明

本平台基于开源项目若依后台管理系统改造而来。http://doc.ruoyi.vip

1.2 开发工具

采用Idea作为开发工具。

# 2. 技术选型

2.1 核心框架

1. 核心框架：Spring Boot 2.1.3
2. 安全框架：Apache Shiro、JWT
3. 模板引擎：Thymeleaf
4. 持久层框架：MyBatis、MyBatisPlus
5. 定时任务:Quartz
6. 数据库连接池：Druid
7. 缓存框架：Redis

2.2 第三方组件

1. 工具类：Hutool
2. 图表控件：echarts
3. 网络通讯：websocket
4. PDF操作：itextpdf
5. Office文件处理：easypoi
6. 打印插件：lodop
7. 微信开发插件：weixin-java-mp

* 1. 前端框架

1. Layui
2. weui+前端框架
3. toastr
4. select2

# 3. 主要功能说明

代码结构：  
com.ruoyi  
├── common // 工具类  
│ └── annotation // 自定义注解  
│ └── config // 全局配置  
│ └── constant // 通用常量  
│ └── enums // 通用枚举  
│ └── exception // 通用异常  
│ └── json // JSON数据处理  
│ └── support // 字符集处理  
│ └── utils // 通用类处理  
│ └── xss // XSS过滤处理  
├── framework // 框架核心  
│ └── aspectj // 注解实现  
│ └── config // 系统配置  
│ └── datasource // 数据权限  
│ └── manager // 异步处理  
│ └── shiro // 权限控制  
│ └── util // 通用工具  
│ └── web // 前端控制  
├── ruoyi-generator // 代码生成  
├── ruoyi-quartz // 定时任务  
├── ruoyi-system // 系统代码  
├── ruoyi-admin // 后台服务  
├── ruoyi-xxxxxx // 其他模块

不能轻易覆盖的文件：  
static/ajax/libs/layui/plugin/formSelects/formSelects-v4.js  
static/ajax/libs/layer/layer.min.js  
  
注意：  
 pad、screen、api、wap 这几个关键字已做特殊路径排除在shiro监控范围之外，新建requestmapping不要使用这几个关键词  
 PAD页面地址： http://localhost/pad  
 手机端地址： http://localhost/wap  
 大屏幕地址： http://localhost/screen  
 平台私有JS ruoyi-admin/static/js/my.js  
 平台私有CSS ruoyi-admin/static/css/my.css（由于框架采用iframe方式打开页面，所以需要在index页面引用）  
 项目私有JS ruoyi-web/static/js/private.js  
 项目私有CSS ruoyi-web/static/css/private.css  
  
API接口说明：  
@RequestMapping("/api/\*") 为api开头的请求方法不会进行shiro验证  
接口代码实现目录 com.ruoyi.api  
接口管理访问地址 http://localhost:81/swagger-ui.html  
接口验证，采用JWT实现。com/ruoyi/framework/aspectj/AuthenticationAspect.java  
  
登录页面修改说明：  
templates/login.html  
默认内置集成了3套不同风格的登录页面  
登录页面相关静态资源存放规则 static/ruoyi/login/1/\* （1文件下存放一种风格的资源，如果新加请延续）  
  
系统名称：  
配置KEY，sys.config.systemName  
在系统管理->参数管理 中进行在线修改  
  
模板路径：  
static/docs/\* 存放各类下载、文档生成等模板文件  
  
redis:  
redis使用Jedis做为底层(注意：hutool中的redis未使用)  
application-druid.yml 文件中prefix对应值就是本项目的前缀（每个项目请务必修为本项目名称）  
com/ruoyi/common/utils/JedisUtils.java  
key命名规则，XXXX:XXXX（例如：token:userid）  
  
wesocket:  
com/ruoyi/web/websocket/SocketServer.java  
static/js/websocket.js  
参考demo: templates/index.html  
  
jwt:  
自定义注解：@ValidateRequest  
请求头部加入key: x-access-token  
验证实现切面：ruoyi-web\src\main\java\com\ruoyi\framework\aspectj\AuthenticationAspect.java  
验证逻辑：  
1.在需要去验证的接口添加@ValidateRequest  
2.通过 /api/auth/login 接口 获得token  
3.将token写入请求头部进行请求  
  
微信：  
核心实现 ruoyi-wx  
demo实现 com/ruoyi/web/controller/wap/WechatController.java  
入口方法 /wap/wechat/access?appid=wx2b911f070115b89a  
  
SQL打印：  
ruoyi-framework/src/main/java/com/ruoyi/framework/config/MybatisPlusConfig.java

# 4. 部署系统

4.1 war部署方式

4.1.1 修改pom.xml文件。将jar修改为war



如果是分模块需要修改ruoyi-admin



4.1.2 在spring-boot-starter依赖中移除tomcat模块



<exclusions>

<exclusion>

<artifactId>spring-boot-starter-tomcat</artifactId>

<groupId>org.springframework.boot</groupId>

</exclusion>

</exclusions>

4.1.3 部署到tomcat的webapps目录下面

默认为ruoyi\_admin.war



4.1.4 启动及验证

运行startup.bat 出现如下图即部署成功

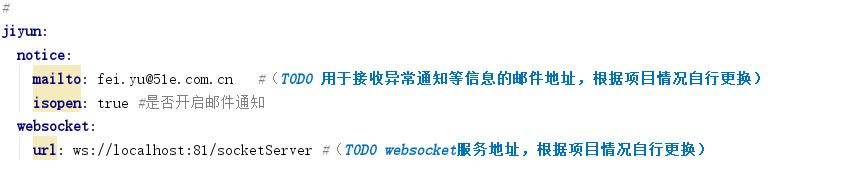
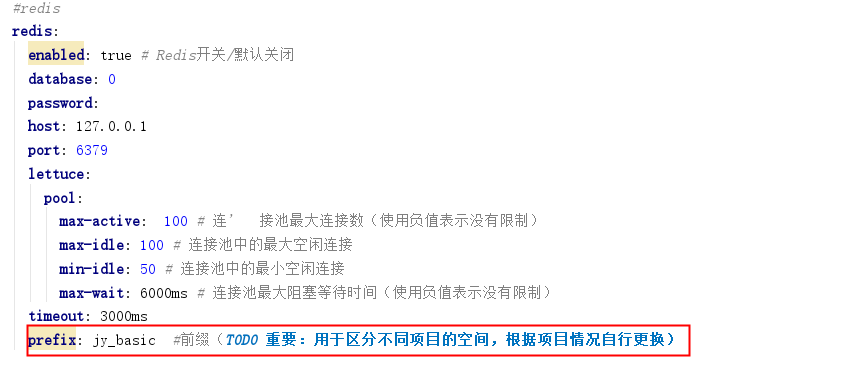


4.2 Jar方式部署

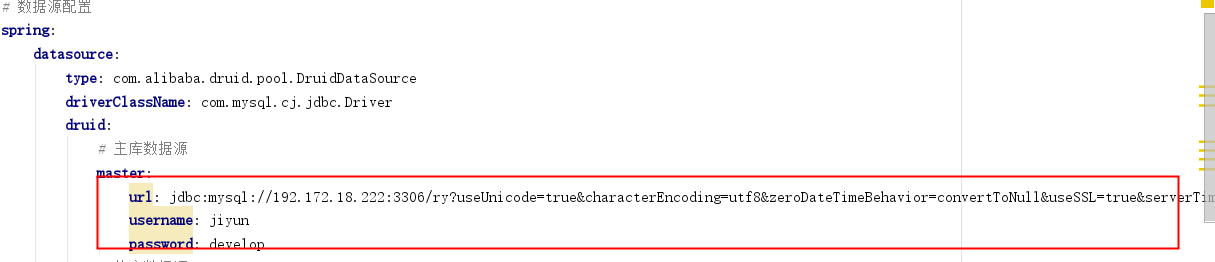
执行命令：java –jar ruoyi\_admin.jar

脚本执行：ry.sh start 启动stop 停止

**重要：每个项目必须修改的配置，请根据项目实际情况修改。  
1. application.yml**



2. application-druid.yml



3. logback.xml

