****

**Ruoyi-Flex开发编译手册**

数据小王子

2024

[Ruoyi-Flex开发编译手册 1](#_Toc12119)

[1、搭建Java开发环境 3](#_Toc30569)

[1.1、高度注意事项 3](#_Toc9892)

[1.2、本地开发环境 3](#_Toc29777)

[1.2.1、环境要求 3](#_Toc25631)

[1.2.2、JDK 3](#_Toc2402)

[1.2.3、Maven 6](#_Toc32255)

[1.2.4、IDEA 8](#_Toc27557)

[1.2.5、Redis 8](#_Toc5677)

[1.2.6、数据库 8](#_Toc31321)

[1.2.7、数据库图形管理工具 8](#_Toc10237)

[1.2.8、MinIO 9](#_Toc16457)

[1.3、配置Maven 9](#_Toc17581)

[1.3.1、配置华为Maven镜像服务器 9](#_Toc31959)

[1.3.2、配置Maven本地仓库 10](#_Toc15567)

[2、安装数据库 10](#_Toc23092)

[2.1、启动本机免安装版PostgreSQL（或者MySQL） 10](#_Toc26120)

[2.2、创建数据库ruoyi-flex 10](#_Toc25261)

[2.3、导入执行数据库脚本 11](#_Toc26576)

[2.4、升级数据库结构 13](#_Toc5352)

[3、准备依赖项目 14](#_Toc3852)

[3.1、启动Redis 14](#_Toc14218)

[3.2、启动MinIO 14](#_Toc26565)

[4、运行java主项目 15](#_Toc20132)

[4.1、配置JDK版本，下载依赖 15](#_Toc5447)

[4.2、配置参数 19](#_Toc27084)

[4.2.1、选择配置文件 19](#_Toc25698)

[4.2.2、开发环境配置文件 19](#_Toc30682)

[4.2.3、监控中心配置文件 23](#_Toc12467)

[4.2.4、调度中心配置文件 23](#_Toc30103)

[4.3、配置IDEA 24](#_Toc15633)

[4.3.1、配置项目编码为UTF-8 24](#_Toc14051)

[4.3.2、安装IDEA插件 25](#_Toc24729)

[4.3.3、打开“服务”面板 26](#_Toc17956)

[4.4、使用默认Mybatis-Flex配置参数 27](#_Toc13552)

[4.5、maven编译 28](#_Toc29682)

[4.6、调试运行 29](#_Toc6063)

[4.6.1、启动顺序 29](#_Toc16732)

[4.6.2、单独启动RuoYiApplication 31](#_Toc8216)

[5、运行前端 32](#_Toc24903)

[5.1、开发环境准备 32](#_Toc29571)

[5.1.1、安装Git 32](#_Toc12710)

[5.1.2、安装NodeJS 32](#_Toc30245)

[5.1.3、设置npm镜像源仓库 33](#_Toc21413)

[5.1.4、安装Vite 33](#_Toc797)

[5.1.5、安装IDE 33](#_Toc509)

[5.2、运行主项目前端 34](#_Toc28147)

[5.2.1、下载前端项目 34](#_Toc32517)

[5.2.1、打开项目 34](#_Toc24800)

[5.2.2、安装依赖 34](#_Toc69)

[5.2.3、运行项目 35](#_Toc1701)

[5.2.4、服务监控 37](#_Toc27532)

[5.2.5、任务调度 37](#_Toc9138)

[6、项目结构 38](#_Toc6623)

[7、演示模块 39](#_Toc8684)

[8、代码生成 40](#_Toc14011)

本文档以在本地Win10电脑上编译部署Ruoyi-Flex单体版为例进行归纳总结：

# 1、搭建Java开发环境

## 1.1、高度注意事项

源码、JDK、MySQL、Redis等存放路径禁止包含中文、空格、特殊字符等，否则易出现问题 ，敬请高度注意！！！！！！

## 1.2、本地开发环境

### 1.2.1、环境要求

|  |  |  |
| --- | --- | --- |
| **项目** | **推荐版本** | **说明** |
| JDK | 21.0.1+ | JAVA环境依赖(需配置环境变量) |
| Spring Boot | 3.2.1+ | Spring环境 |
| Maven | 3.9.3及以上版本 | 项目构建(需配置环境变量) |
| IDEA | IDEA2023.3及以上版本 | 集成开发工具 |
| Redis | V5及以上版本 | 缓存 |
| MySQL | 8.0.x+ | 数据库任选一(Ruoyi-Flex V4.1.X默认) |
| PostgreSQL | 15+ | 数据库任选一(Ruoyi-Flex V4.2.X默认) |

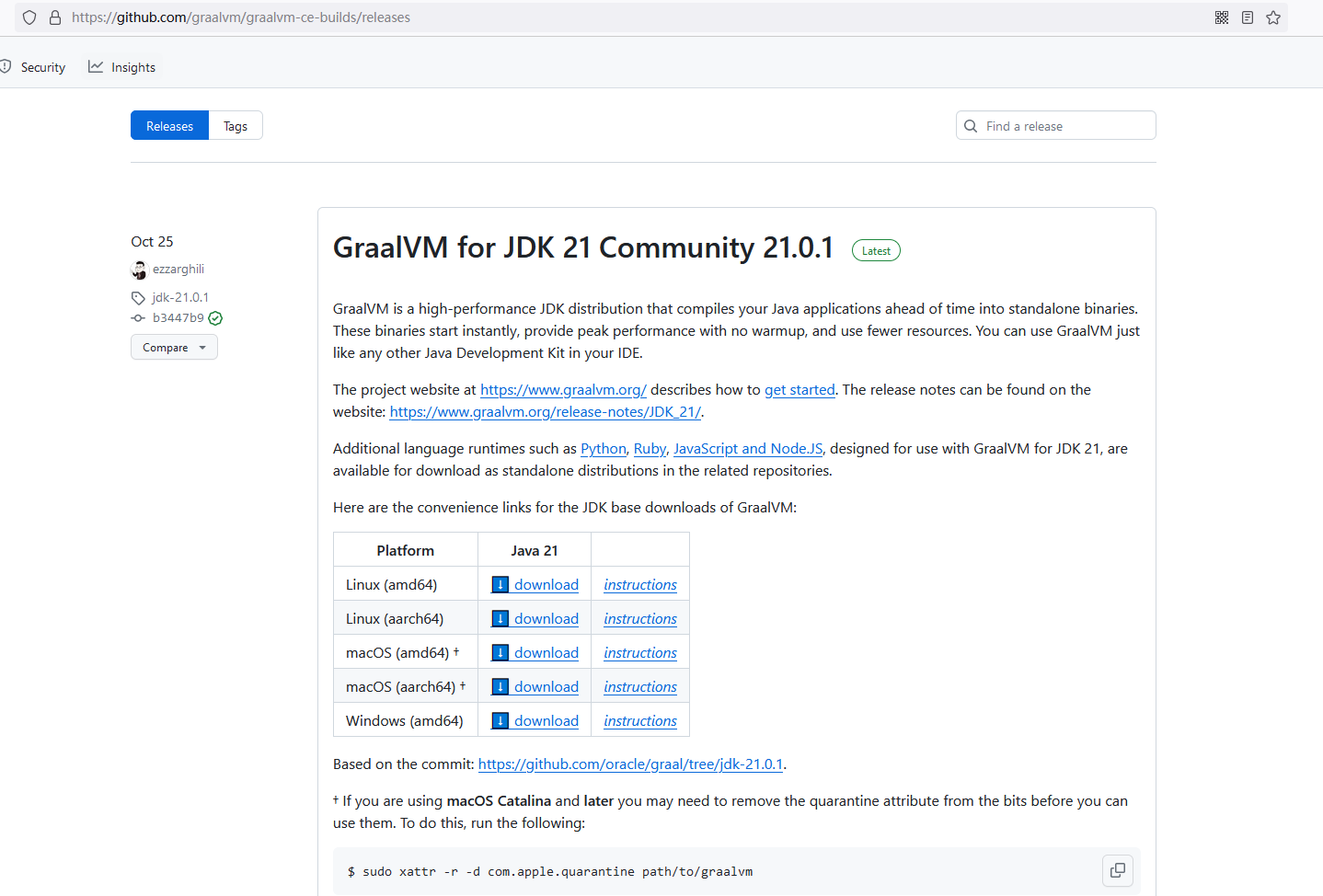
### 1.2.2、JDK

Ruoyi\_Flex的V4.1.X需要jdk17，Ruoyi\_Flex的V4.2.X+需要JDK21。

JDK版本不低于 21.0.1 版本，具体是openjdk-21 或 graalvm-community-openjdk-21.x版本。需要使用openjdk或者graalvm 运行Ruoyi-Flex，从原理上讲国内厂商的Alibaba Dragonwell 21 、 BiShengJDK 21等JDK也是适用的。

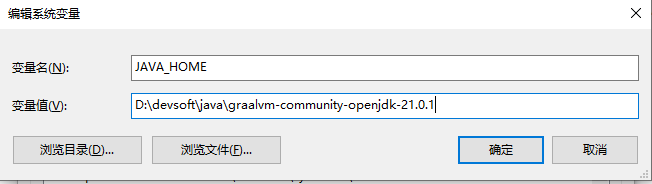
禁止使用 OracleJdk，原因：一是由于spring的bug导致打包运行会报错)；二是Oracle公司对jdk收费了；三是国产信创安全的大势所趋。

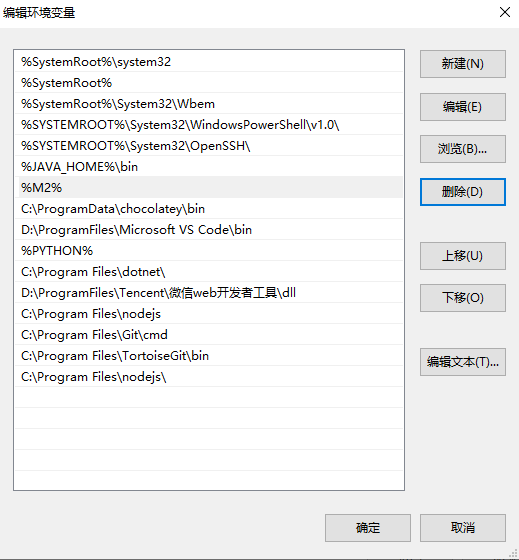
graalvm的下载地址：<https://github.com/graalvm/graalvm-ce-builds/releases，>



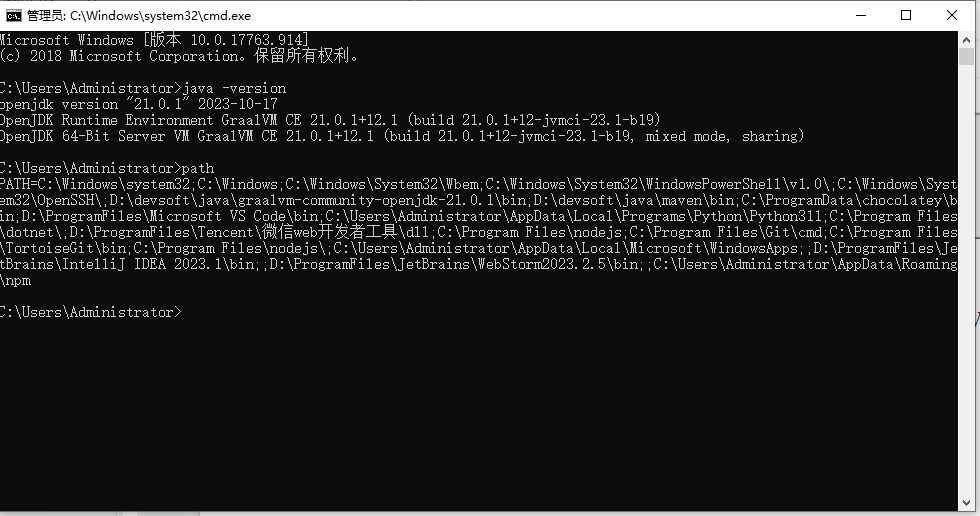
注意下载GraalVM for JDK 21 Community 21.X.Y+版本。下载下来解压安装到本地没有中文、空格、特殊字符的文件夹中。

我的微机上安装的是graalvm-community-openjdk-21.0.1，设置环境变量JAVA\_HOME指向安装目录：D:\devsoft\java\graalvm-community-openjdk-21.0.1；将下面的bin加入系统环境变量Path中：





打开“命令提示符”，输入java -version及path两个命令，查看安装是否成功：



### 1.2.3、Maven

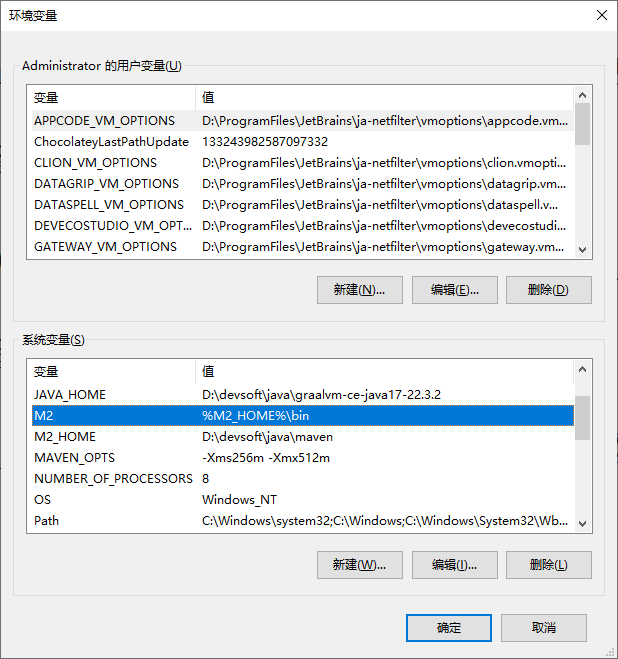
Maven建议安装“3.9.3及以上版本”，我的微机上安装的是3.9.3绿色版Maven，后期可以根据项目需求随时切换maven版本。

建立三个系统变量：

“M2\_HOME”指向安装路径“D:\devsoft\java\maven”

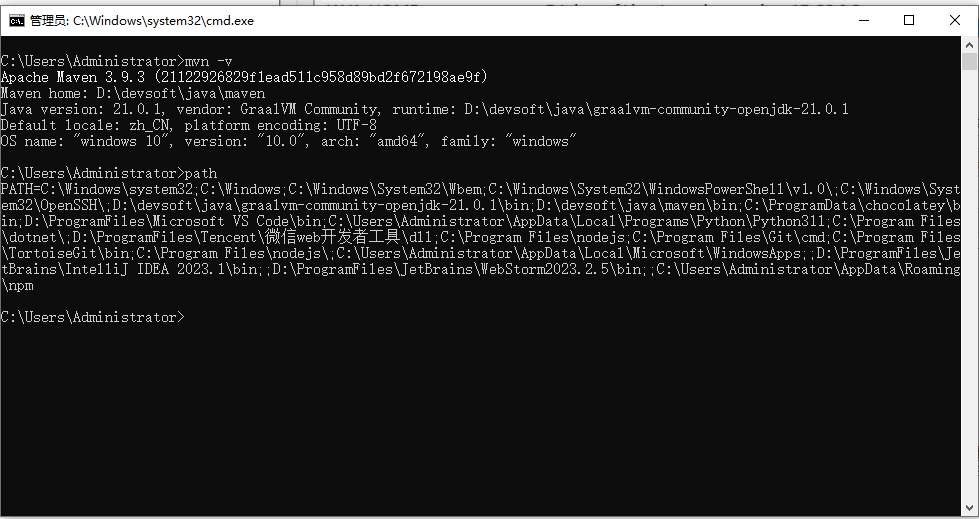
“M2”指向bin文件夹

“MAVEN\_OPTS”为运行参数“-Xms256m -Xmx512m”

如图：

然后将%M2%加入Path中。

打开“命令提示符”，输入mvn -v及path两个命令，查看安装是否成功：



### 1.2.4、IDEA

IDEA建议安装“2023.3及以上版本”，IDEA 2023.3完全支持JDK21，具备良好的开发体验。

### 1.2.5、Redis

由于Ruoyi-Flex大量使用了redis特性，Redis版本必须大于 5.X，我的微机安装的是“Redis 7.0.11”，win版下载地址：<https://github.com/zkteco-home/redis-windows/releases>。

### 1.2.6、数据库

Ruoyi-Flex计划支持2种开源数据库：MySQL（V8以上）、PostgreSQL（V15以上）。我的微机上安装的是绿色免安装版的“MySQL V8.0.27”、“PostgreSQL V16.1”。国产信创数据库产品大多基于这两个数据库开发，使用语法格式是兼容的。

### 1.2.7、数据库图形管理工具

可以使用自己熟悉的任何一款数据库图形管理工具，我微机上的数据库图形管理工具是“HeidiSQL V12.4”,免费开源的绿色免安装版。

### 1.2.8、MinIO

如果在本地存放头像、文档文件，则需要安装开源免费的MinIO分布式存储系统。如果把文件存储到七牛、阿里、腾讯等云存储服务器上，就没有必要安装MinIO了。

MinIO的官方网站下载地址：https://min.io。

## 1.3、配置Maven

### 1.3.1、配置华为Maven镜像服务器

如果不能从maven服务器下载依赖文件，就配置华为Maven镜像：

打开maven安装路径conf下配置文件settings.xml，添加：

<mirror>

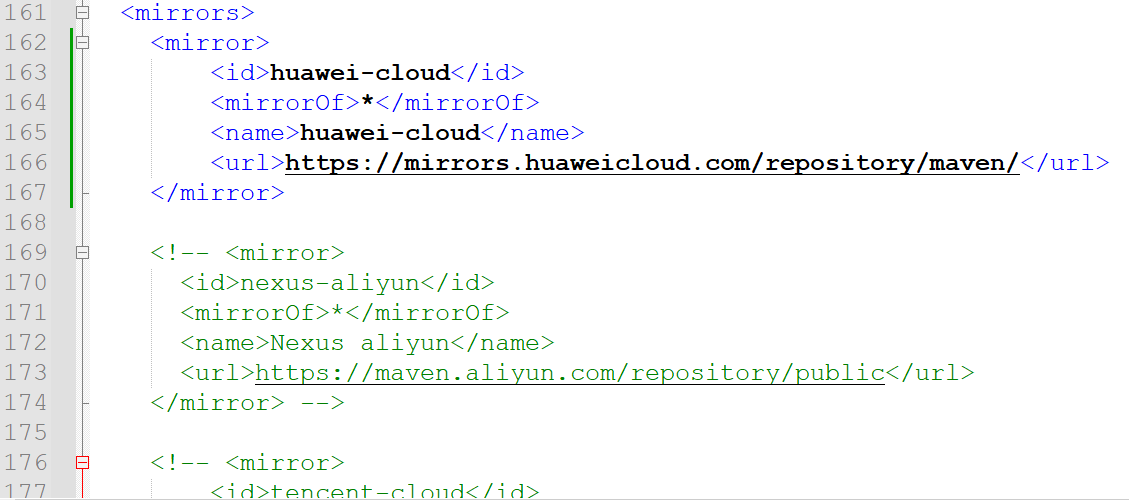
<id>huawei-cloud</id>

<mirrorOf>\*</mirrorOf>

<name>huawei-cloud</name> <url>https://mirrors.huaweicloud.com/repository/maven/</url>

</mirror>

同时将其它镜像注释取消或者将华为镜像服务器移到上面一个（因为maven默认第一个镜像起作用），如图：



### 1.3.2、配置Maven本地仓库

为了让C盘瘦身，将Maven本地仓库设置到其它盘符中。

我的微机是设置为：D:\devsoft\java\maven\repository文件夹，配置settings.xml文件：

<localRepository>D:\devsoft\java\maven\repository</localRepository>

# 2、安装数据库

下面以本地数据库为例来说明安装过程：

## 2.1、启动本机免安装版PostgreSQL（或者MySQL）

免安装版数据库管理系统PostgreSQL V16.1（或者MySQL V8.0.27）在本机硬盘上，先启动起来。

## 2.2、创建数据库ruoyi-flex

为了开发方便，将“平台数据库”与“调度数据库”放到一个数据库ruoyi-flex中，相关命令参考如下：

PostgreSQL格式的建库脚本命令：

CREATE DATABASE "ruoyi-flex"

WITH

OWNER = postgres

ENCODING = 'UTF8'

LOCALE\_PROVIDER = 'libc'

CONNECTION LIMIT = -1

IS\_TEMPLATE = False;

MySQL 格式的建库脚本命令：

CREATE DATABASE IF NOT EXISTS `ruoyi-flex` /\*!40100 DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4\_bin \*/ /\*!80016 DEFAULT ENCRYPTION='N' \*/;

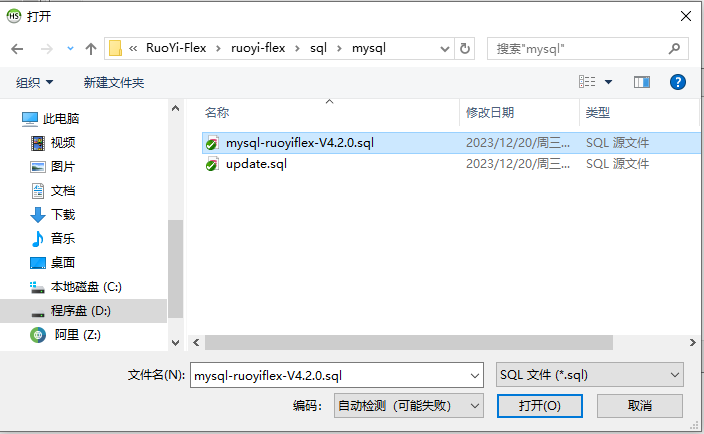
每次大版本升级，数据库脚本文件（相应数据库-ruoyiflex-V版本号.sql）会全部更新一次，小版本更新会放到update.sql文件中。

## 2.3、导入执行数据库脚本

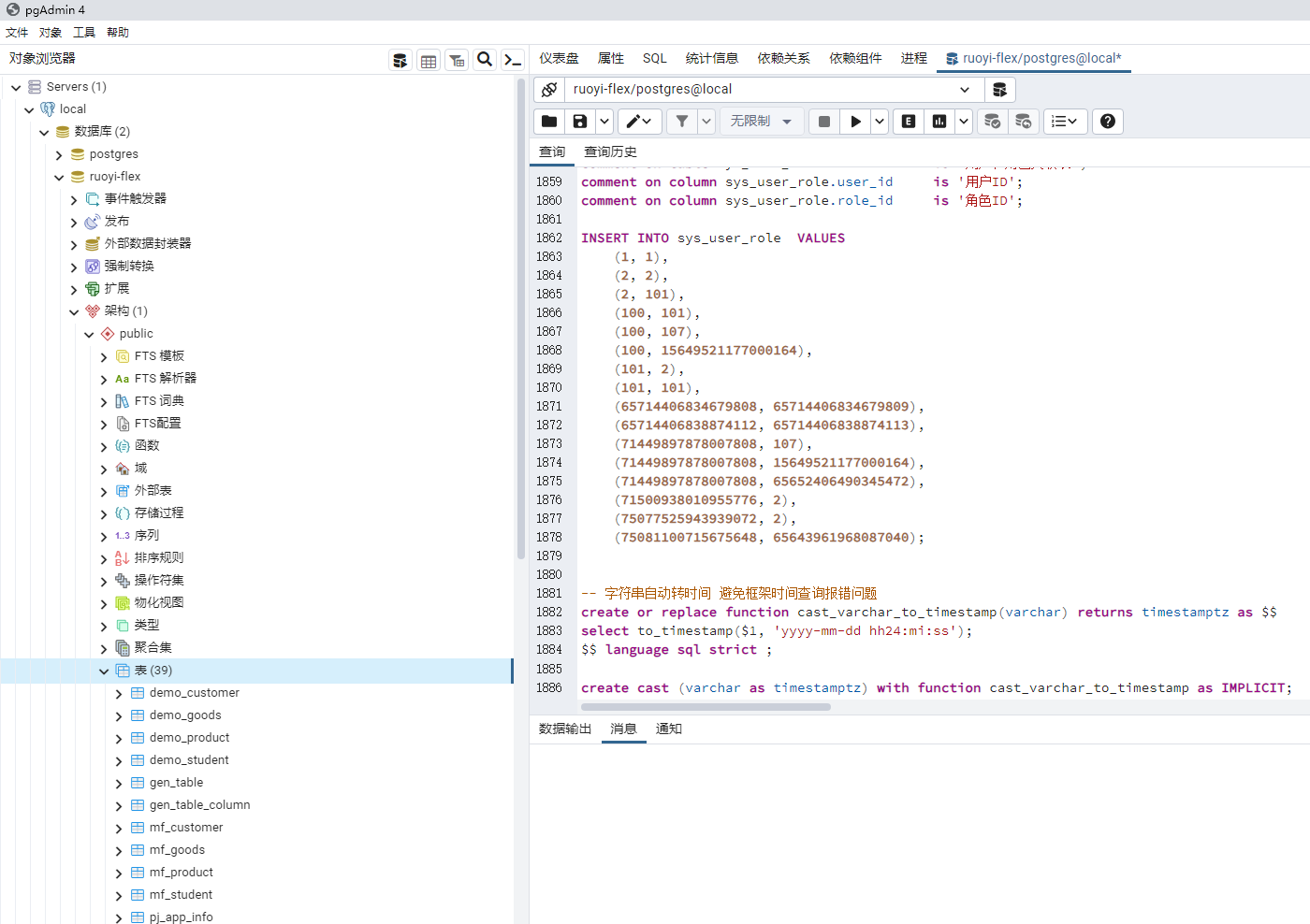
从V5.2.0开始，系统引入了另一个任务调度框架EasyRetry，两个任务调度框架（EasyRetry、PowerJob）的SQL各自位于自己的单独文件中。

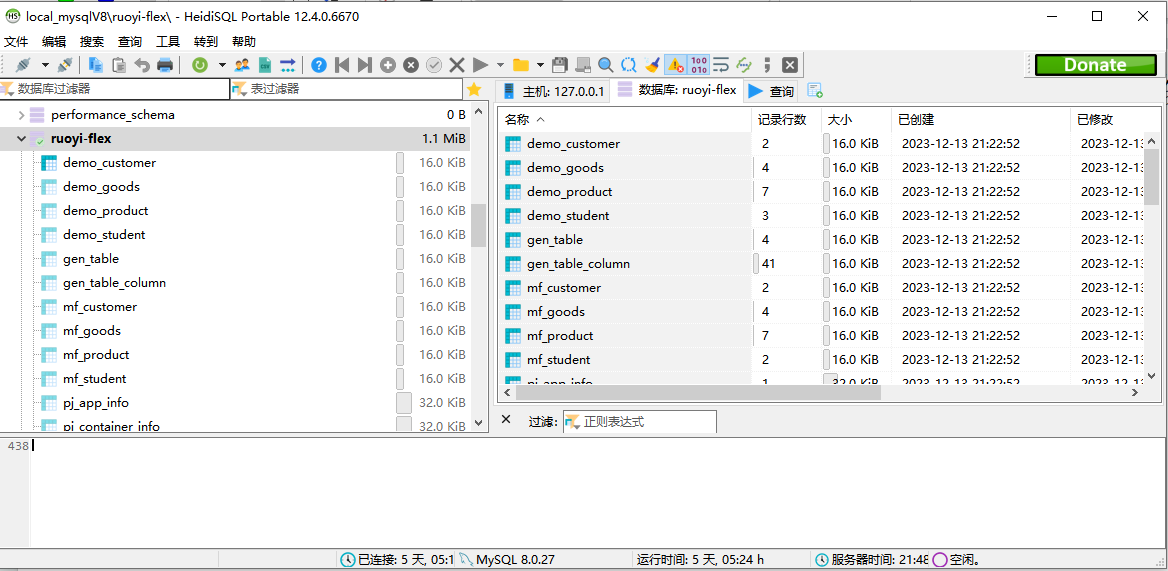
**1、创建ruoyiflex库表**

使用任何一款数据库图形管理工具（HeidiSQL）连接上后台数据库，通过“文件-》运行SQL”，选择最新版本，以V4.2.0版本为例进行描述：\ruoyi-flex\sql\postgresql\postgresql-ruoyiflex-V4.2.0.sql (或者mysql\mysql-ruoyiflex-V4.2.0.sql)文件，打开：



建立表结构定义，插入系统默认数据，成功执行后，刷新ruoyi-flex数据库共有40张表：





**2、创建任务调度框架表**

根据选择的任务调度框架EasyRetry（PowerJob），可以创建单独的数据库或者导入ruoyiflex库**：**请执行ruoyi-flex\sql\mysql(或者postgresql)\easy\_retry（或者powerjob）.sql文件。

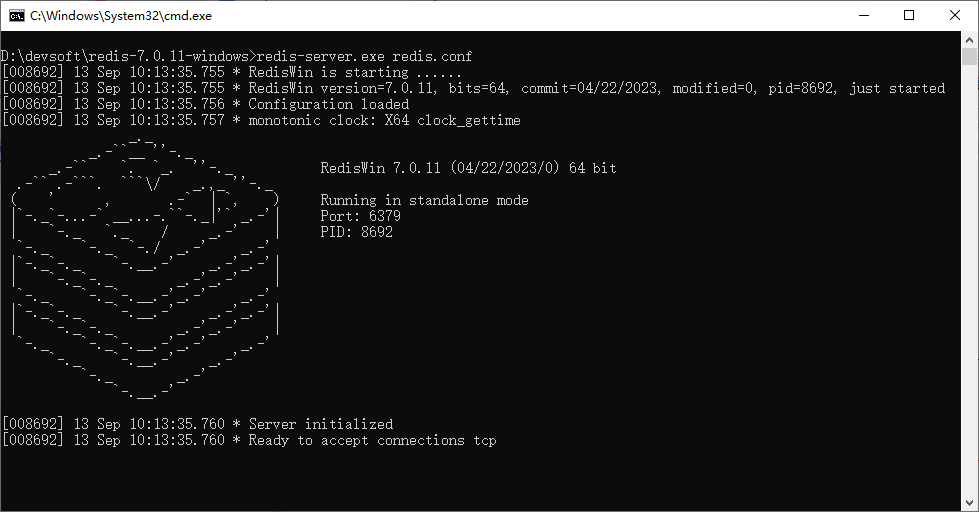
## 2.4、升级数据库结构

每次小版本升级，请执行ruoyi-flex\sql\mysql(或者postgresql)\update.sql文件:使用任何一款数据库图形管理工具（HeidiSQL）连接上后台数据库，通过“文件-》运行SQL”，选择该文件，打开，执行即可。

# 3、准备依赖项目

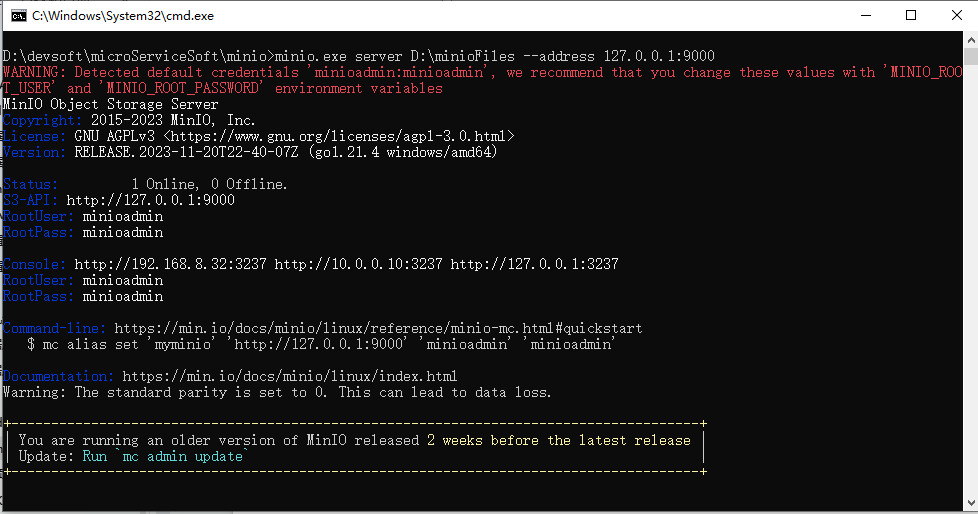
## 3.1、启动Redis

我的微机安装的是“Redis 7.0.11”，在Redis安装文件夹使用redis-server.exe redis.windows.conf命令启动Redis：



## 3.2、启动MinIO

启动本地的MinIO程序（如果不需要，就跳过本步骤）：



第一次使用，请登录MinIO控制台(http://127.0.0.1:5072)来创建桶ruoyi-flex、access\_key与secret\_key,桶ruoyi-flex的access policy为public。

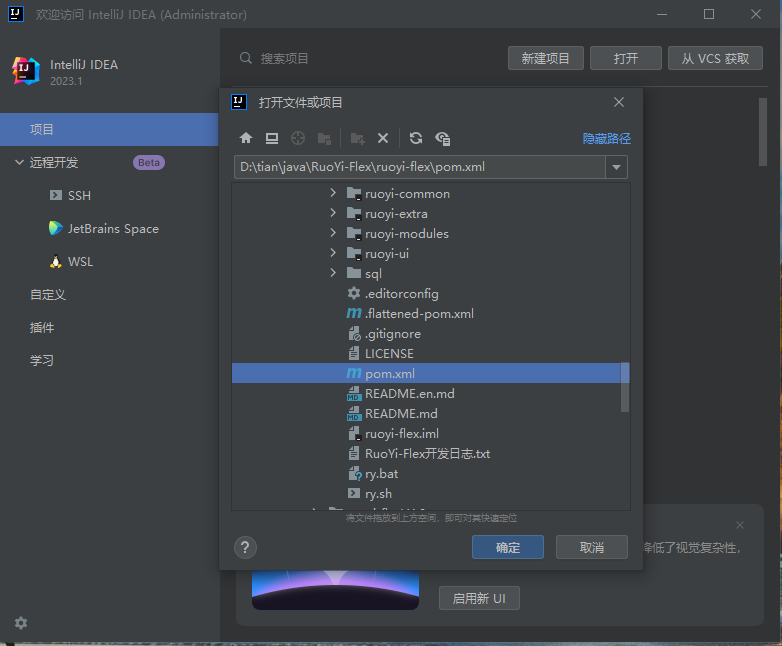
# 4、运行java主项目

## 4.1、配置JDK版本，下载依赖

下载ruoyi-flex源码到本地文件夹：D:\tian\java\RuoYi-Flex\ruoyi-flex。

第一步：使用IDEA打开ruoyi-flex项目：

打开D:\tian\java\RuoYi-Flex\ruoyi-flex\pom.xml文件：



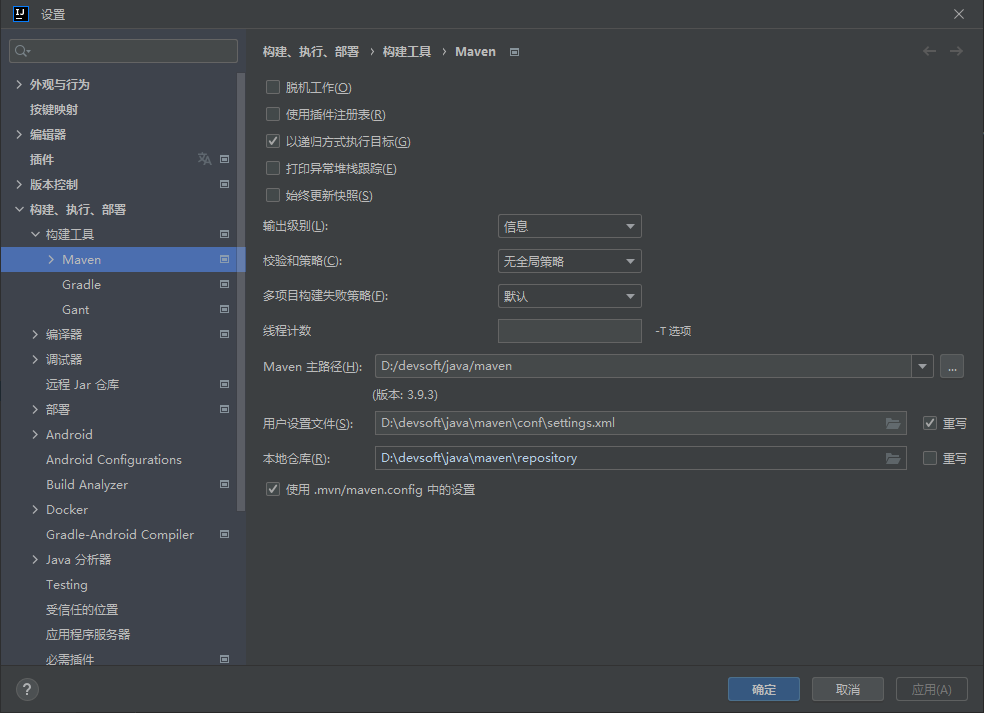
第二步：修改本项目使用本机定制安装的maven，不使用idea内置的maven：

文件->设置：搜索maven：

Maven主路径：D:/devsoft/java/maven

修改用户设置文件：D:\devsoft\java\maven\conf\settings.xml

本地仓库：D:\devsoft\java\maven\repository

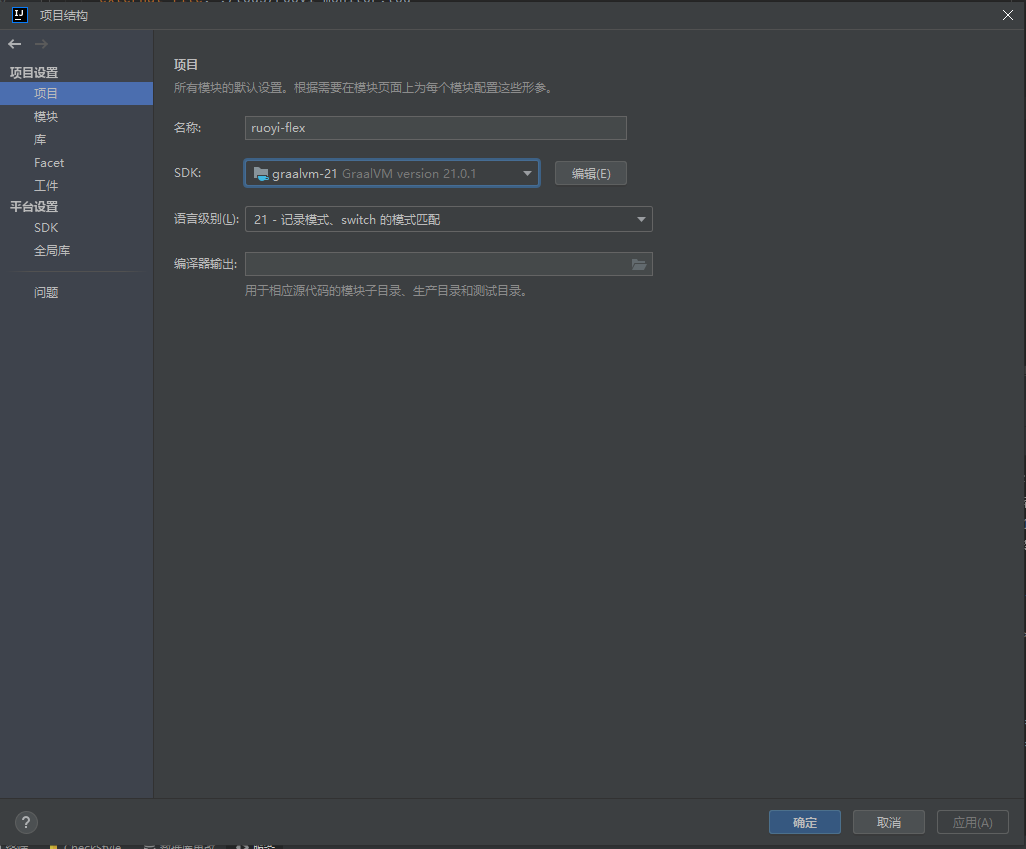


第三步：修改项目结构，使用JDK21

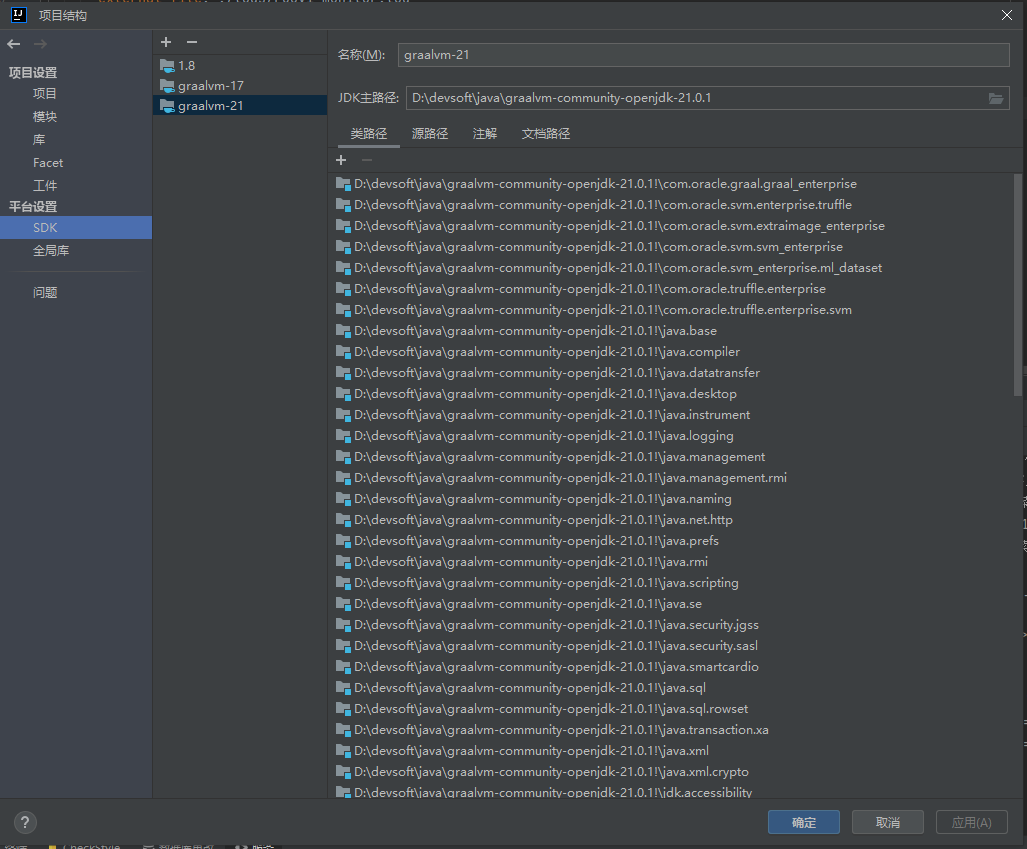
文件->项目结构：

首次进入需要在“项目设置”中的“SDK”框中“添加JDK”，选择JDK21所在的文件夹（D:\devsoft\java\graalvm-community-openjdk-21.0.1）即可。

项目设置：SDK选择“graalvm-21”，语言级别选择“21-记录模式、switch的模式匹配”



平台设置：SDK选择“graalvm-21”

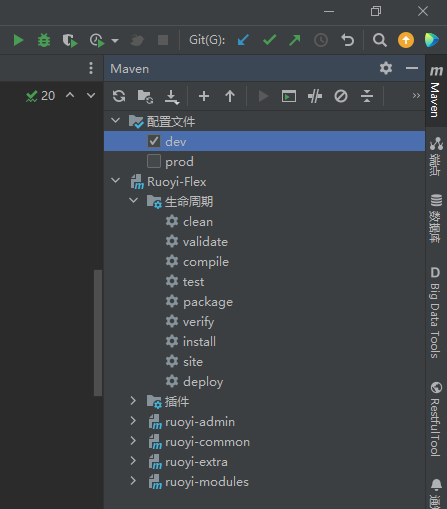


第四步，打开IDEA右侧的maven面板，点击第一个按钮“重新加载maven项目”按钮，静等maven下载各个依赖到本地仓库，如果下载完依赖后IDEA还标红，那是因为使用了Mybatis-Flex的APT（Annotation Processing Tool）技术，在项目编译的时候，会自动根据 Entity 类定义的字段生成一些辅助类（类似SYS-CONFIG之类的），通过执行maven编译命令: mvn clean package 可以自动生成辅助类，IDEA会找到这些文件，不再提示错误。还不到编译阶段，此时先不用管这些错误提示。

## 4.2、配置参数

### 4.2.1、选择配置文件

默认为dev开发环境，可以根据需要勾选相应环境：



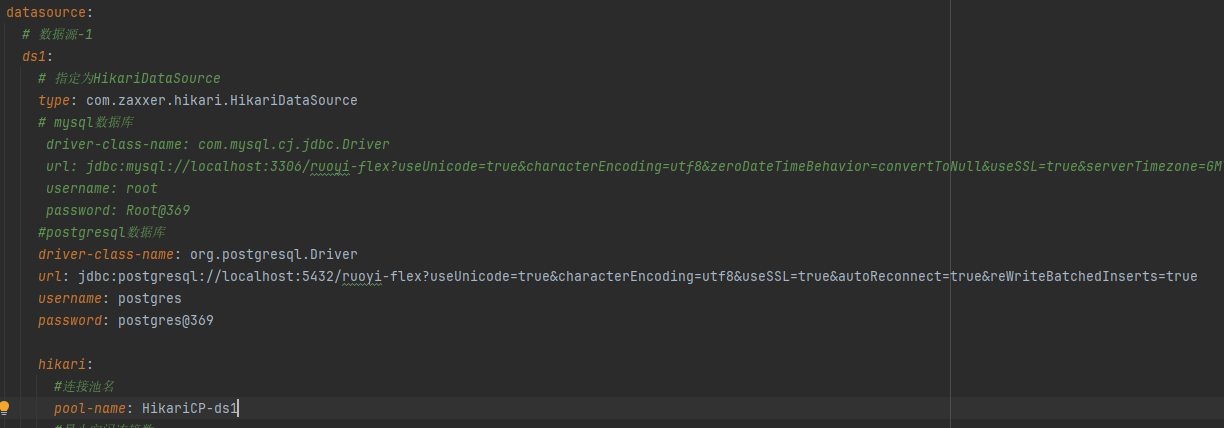
如果项目正式上线运行，请将dev勾选为pro生产环境。

### 4.2.2、开发环境配置文件

打开 \ruoyi-flex\ruoyi-admin\src\main\resources\application-dev.yml：

根据实际开发情况，修改如下参数：

#### （1）数据库配置：



在单数据源的状态下，根据实际情况，选择是连接PostgreSQL还是MySQL数据库，上图演示的是使用PostgreSQL数据库，如果想使用数据库MySQL，就把上图的PostgreSQL数据库部分4行内容注释掉，启用MySQL数据库的4行内容。

请根据实际情况，修改下列默认配置参数内容：

数据库服务器IP地址: localhost

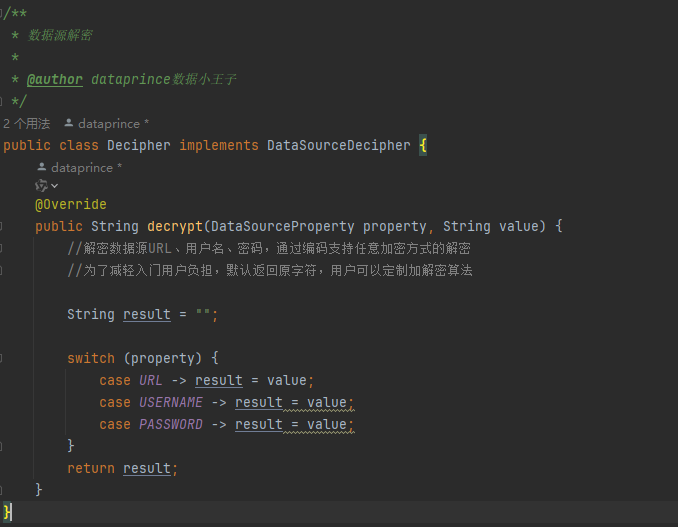
端口: 5432

数据库：ruoyi-flex

用户名: postgres

密码: postgres@369

为了方便入门，数据库登录用户、密码不再加密，有加解密需求的，请修改本文件及 ruoyi-flex\ruoyi-common\ruoyi-common-orm\src\main\java\com\ruoyi\common\orm\decipher\Decipher.java文件：



#### （2）Redis配置

host: localhost

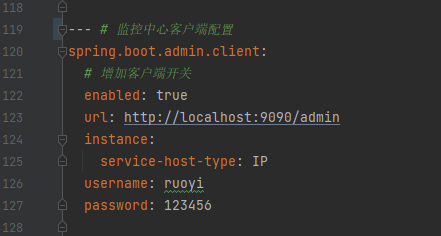
port: 6379

database: 0

#password: # 密码不为空时，请将本行注释启用

timeout: 10s

#### （3）监控中心客户端配置



如果在开发时为了节约内存，不想启动监控中心Sping Boot Admin服务端程序，那相应的spring.boot.admin.client.enabled应该设置为false。

#### （4）easy retry客户端配置



如果在开发时为了节约内存，不想启动EasyRetry服务端程序，那相应的easy-retry.enabled应该设置为false。

#### （5）powerjob客户端配置



如果在开发时为了节约内存，不想启动PowerJob服务端程序，那相应的powerjob.worker.enabled应该设置为false。

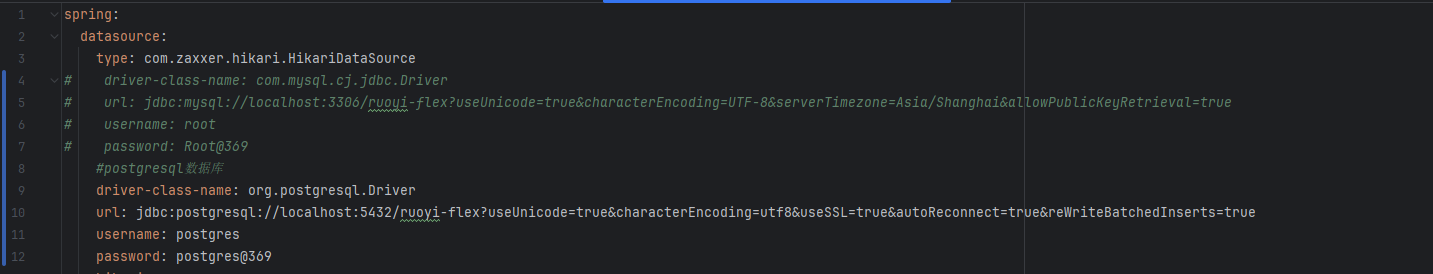
### 4.2.3、监控中心配置文件

Spring Boot Admin监控中心配置文件位于ruoyi-flex\ruoyi-extra\ruoyi-monitor\src\main\resources\application.yml，可以根据实际需要修改其内容。

### 4.2.4、调度中心配置文件

根据实际需要，运行一个任务调度中心即可：

（1）EasyRetry任务调度中心开发环境配置文件位于ruoyi-flex\ruoyi-extra\ruoyi-easyretry-server\src\main\resources\application-dev.yml，可以根据实际需要修改其内容。



根据实际情况，选择是连接PostgreSQL还是MySQL数据库，默认是使用PostgreSQL数据库。根据实际需要，配置jdbc-url、username、password参数值。

（2）PowerJob任务调度中心开发环境配置文件位于ruoyi-flex\ruoyi-extra\ruoyi-powerjob-server\src\main\resources\application-dev.properties，可以根据实际需要修改其内容。

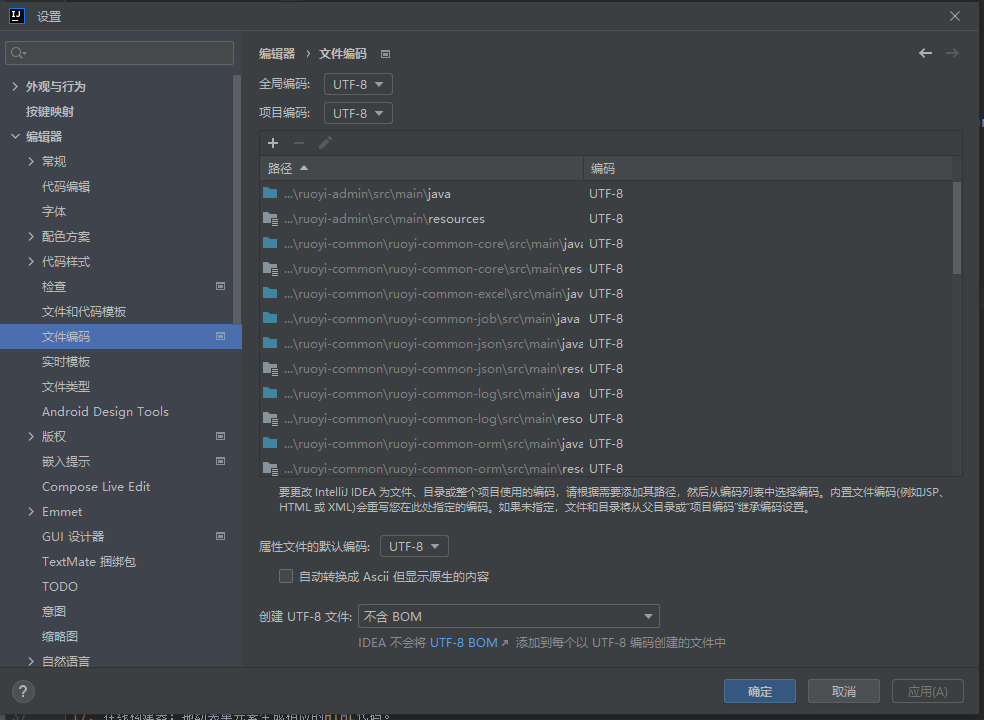


根据实际情况，选择是连接PostgreSQL还是MySQL数据库，默认是使用PostgreSQL数据库。根据实际需要，配置jdbc-url、username、password参数值。

## 4.3、配置IDEA

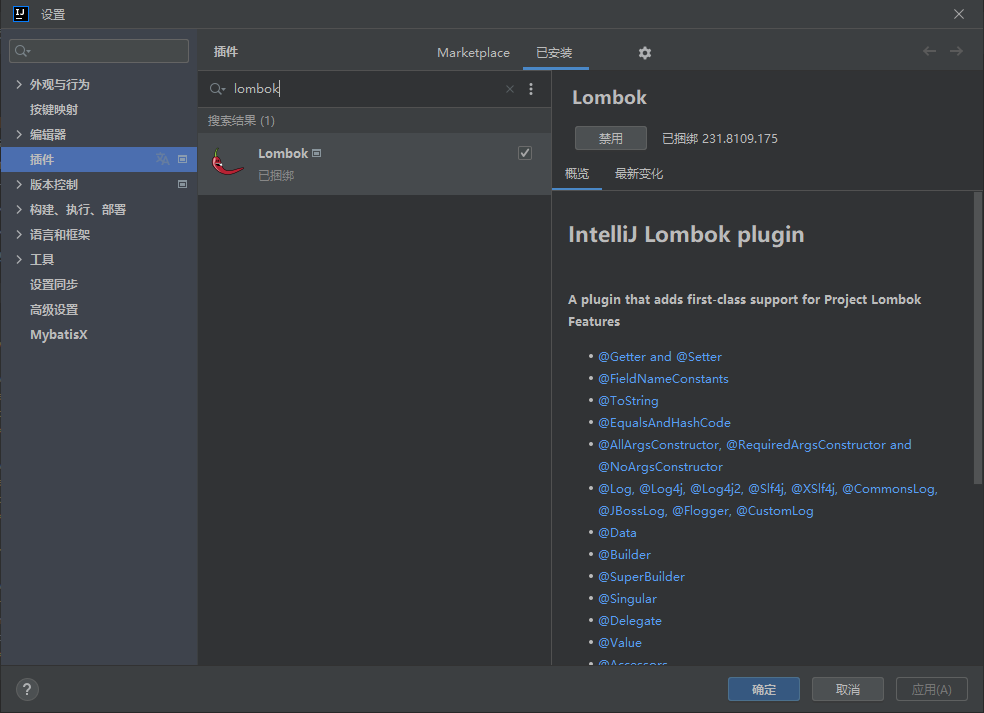
### 4.3.1、配置项目编码为UTF-8

在IDEA的文件（File）->设置（Settings）界面：找到“编辑器”下面的“文件编码”：



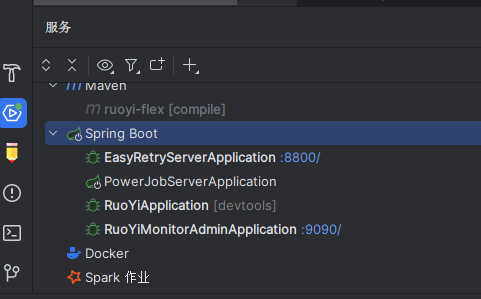
### 4.3.2、安装IDEA插件

在IDEA的文件（File）->设置（Settings）->插件（Plugins）中安装Lombok、Mybatis-Flex-Helper、Chines(Simplified) Language中文语言包、Maven Project Version、Alibaba Java Coding Guidelines、MybatisX、Bito、CodeGeex等插件，其中“Lombok”、“Maven Project Version”是必须安装的。



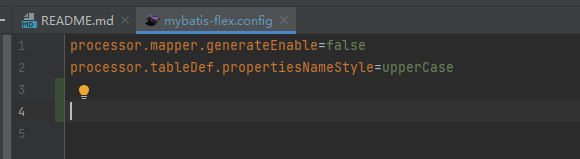
### 4.3.3、打开“服务”面板

使用Alt+8快捷键打开服窗口，或者通过IDEA的视图（View）->工具窗口（Tool Windows）->服务（Services）：



## 4.4、使用默认Mybatis-Flex配置参数

在项目主目录下有mybatis-flex.config文件，其内容如下：

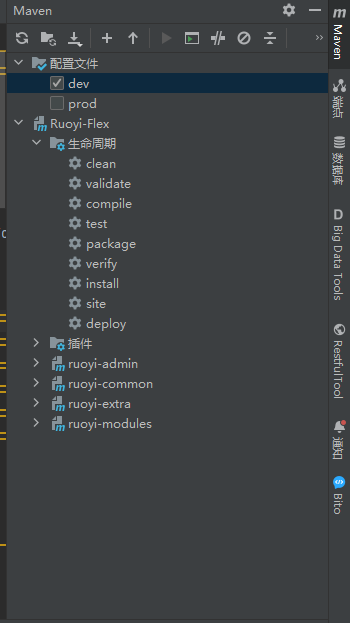


里面的各个配置参数是当前Mybatis-Flex的默认值，为了预防未来的版本这些默认值会发生变化（可能微乎及微）来，所以通过本文件来统一设置一下。

所以，这个文件不是让大家来自定义修改的！这个文件不是让大家来自定义修改的！这个文件不是让大家来自定义修改的！

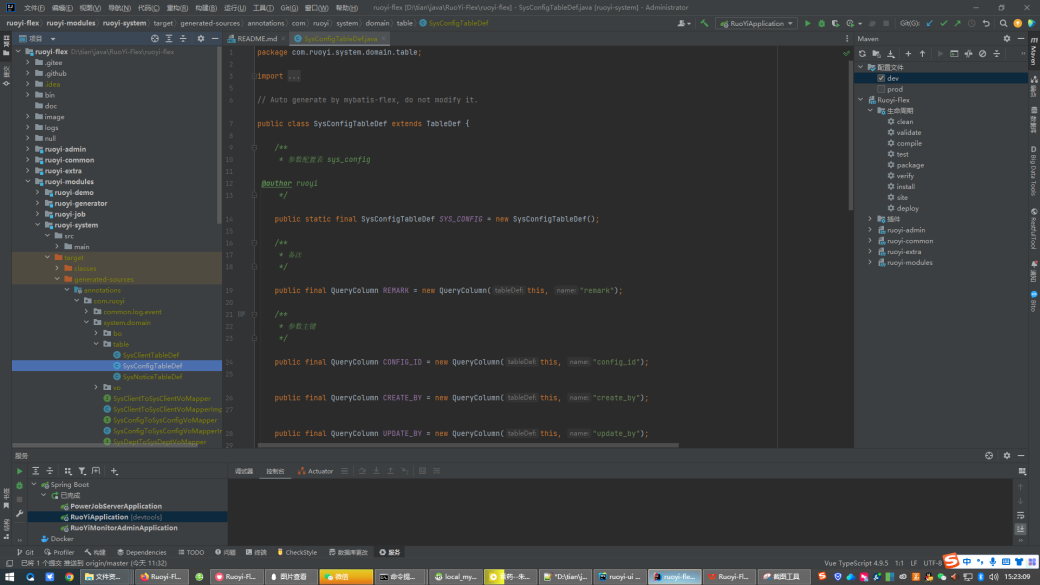
如果你修改了upperCase，那本项目中用到的所有辅助类名称就得重写一遍，工作量可就大了！参数processor.tableDef.propertiesNameStyle的值需要在项目之初就确定下来，中途不能随便更改！

## 4.5、maven编译

打开IDEA右侧的maven面板：

依次点击“Ruoyi-Flex”生命周期的clean、validate、compile、test、package、verify、install，此时就具备运行条件了。

Compile操作后，Mybatis-Flex会在\target\generated-sources\annotations\com\ruoyi\system\domain\table\下面生成\*TableDef.java辅助类，例如：



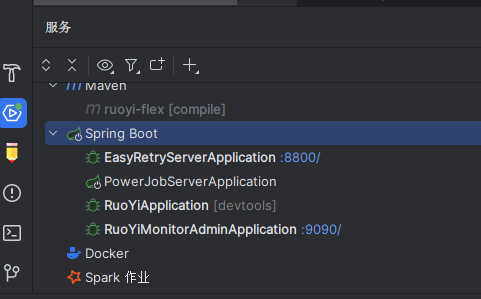
里面就有类似SYS\_CONFIG的定义，本程序中会大量地使用。此时IDEA就不会再报错了！

如果此时偶尔IDEA还报错，一般使用“构建”菜单的“重新构建项目”、“**重新加载所有maven项目**”就可以了；如果还不行，重启电脑，重新打开IDEA就好了。

## 4.6、调试运行

### 4.6.1、启动顺序

目前Ruoyi-Flex内置了4个应用EasyRetryServerApplication、PowerJobServerApplication、RuoYiApplication、RuoYiMonitorAdminApplication：



EasyRetryServerApplication：：EasyRetry调度中心服务器端，非必须启动项目。

PowerJobServerApplication：PowerJob调度中心服务器端，非必须启动项目。

RuoYiMonitorAdminApplication：监控中心Spring Boot Admin的服务器端，非必须启动项目。

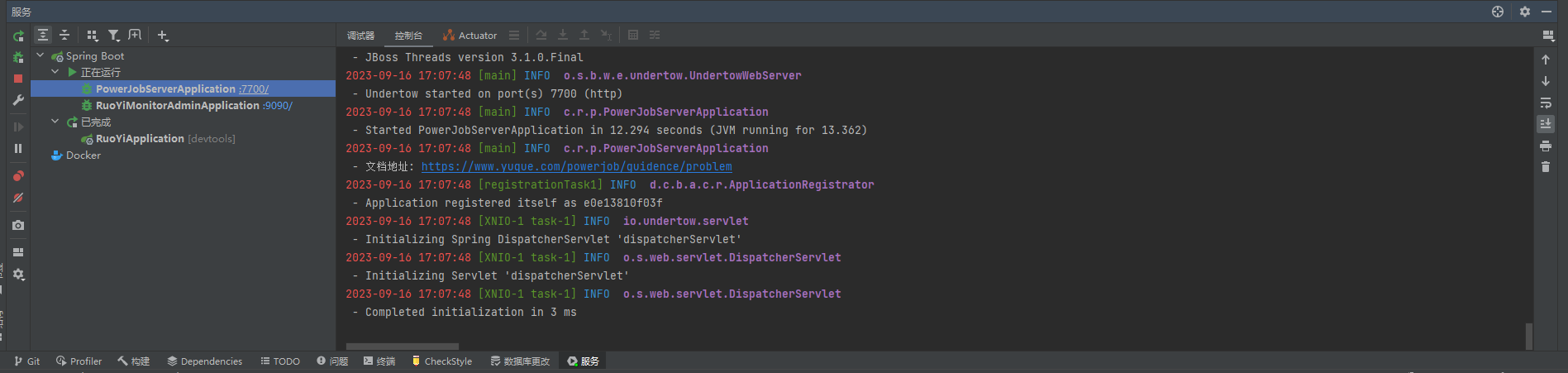
RuoYiApplication为主应用，必须启动项目。

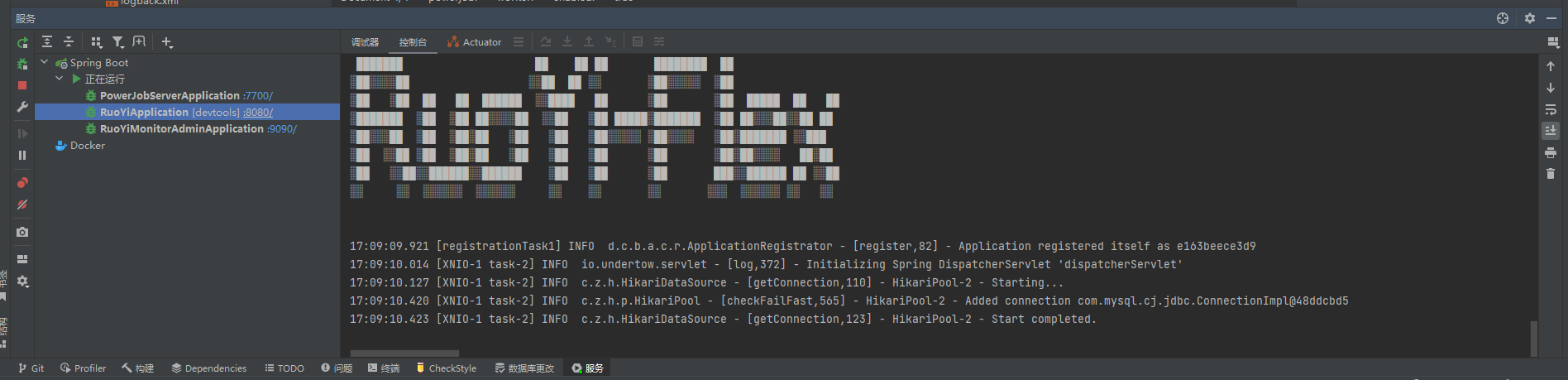
这三个应用如果全部启动，启动顺序依次为：

1. RuoYiMonitorAdminApplication
2. EasyRetryServerApplication（或者PowerJobServerApplication）
3. RuoYiApplication

在IDEA“服务”窗口中，单击相应的应用服务名称，单击“调试”或者“运行”按钮，启动应用。







### 4.6.2、单独启动RuoYiApplication

（1）不启用监控中心客户端

请参照4.2.2节第（3）部分的内容，spring.boot.admin.client.enabled设为false。

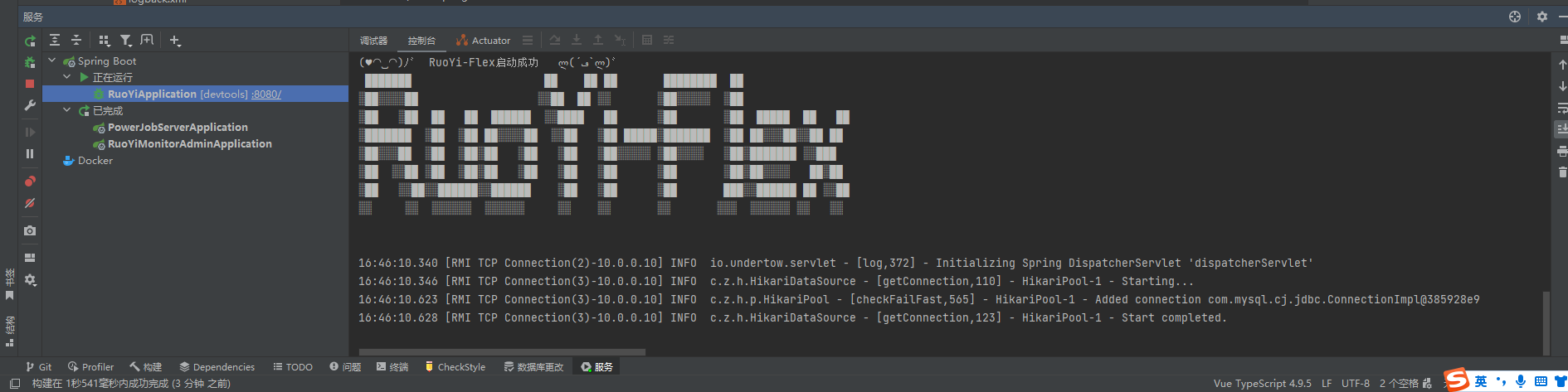
（2）不启用调度中心EasyRetry客户端

请参照4.2.2节第（4）部分的内容，easy-retry.enabled设为false。

（3）不启用调度中心PowerJob客户端

请参照4.2.2节第（5）部分的内容，powerjob.worker.enabled设为false。

（4）单独启动RuoYiApplication即可。



# 5、运行前端

## 5.1、开发环境准备

我以windows10为例，来演示前端windows开发环境的搭建过程：

### 5.1.1、安装Git

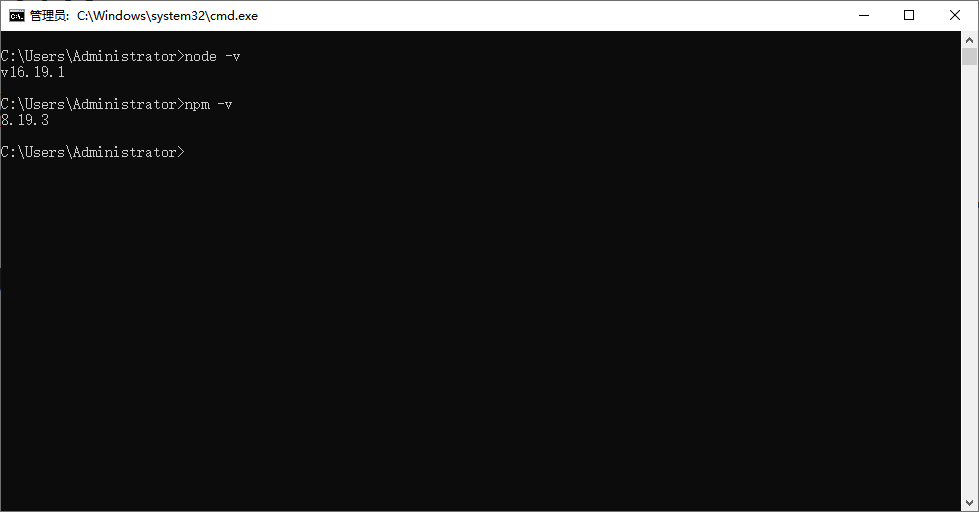
请到https://git-scm.com/网站 ,下载Git安装包，按提示安装即可。

### 5.1.2、安装NodeJS

注意：NodeJS不要使用太低、太高的版本，建议使用V16+！！！

请打开 https://nodejs.org/en/ ，下载V16的LTS最新版本，安装。

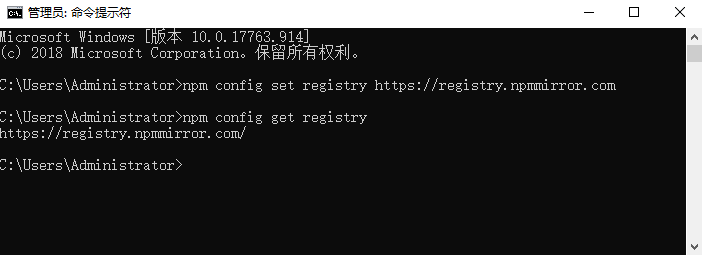
之后打开命令提示符，依次输入如下命令查看安装信息：



### 5.1.3、设置npm镜像源仓库

设置淘宝镜像源：npm config set registry <https://registry.npmmirror.com>

查看镜像源使用状态：npm config get registry



### 5.1.4、安装Vite

打开命令提示符，输入安装Vite的命令：

npm install -g vite

### 5.1.5、安装IDE

根据公司规定、个人喜好，安装前端开发IDE：WebStorm或者VSCode，我的微机安装的是WebStorm V2023.1版本。

## 5.2、运行主项目前端

### 5.2.1、下载前端项目

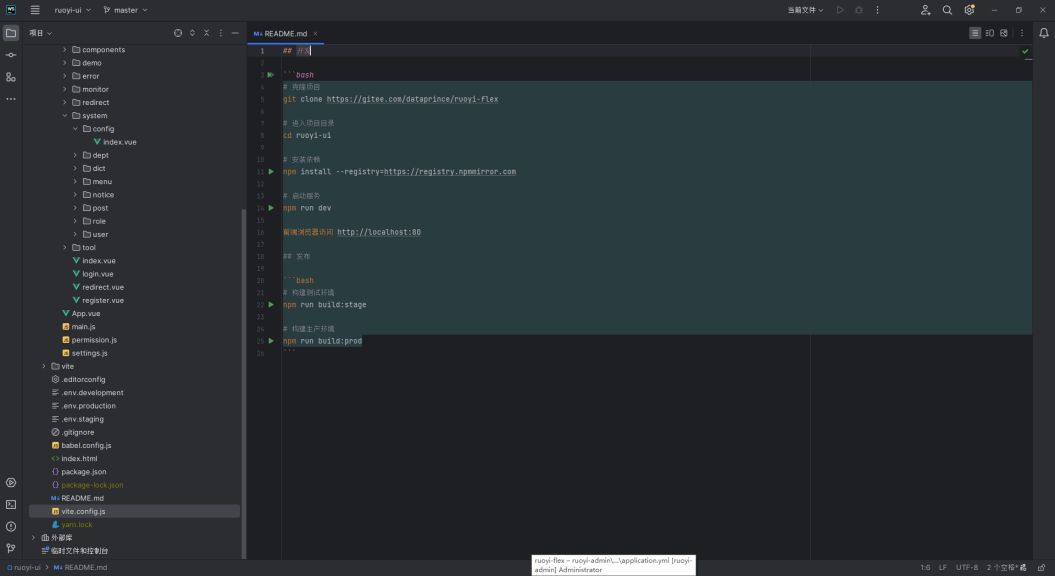
从下列网址https://gitee.com/dataprince/flex-elementplus-ui下载zip文件或者克隆本项目到本地文件夹中：

git clone https://gitee.com/dataprince/flex-elementplus-ui.git

如果是下载的zip文件，请解压到本地flex-elementplus-ui文件夹中。

### 5.2.1、打开项目

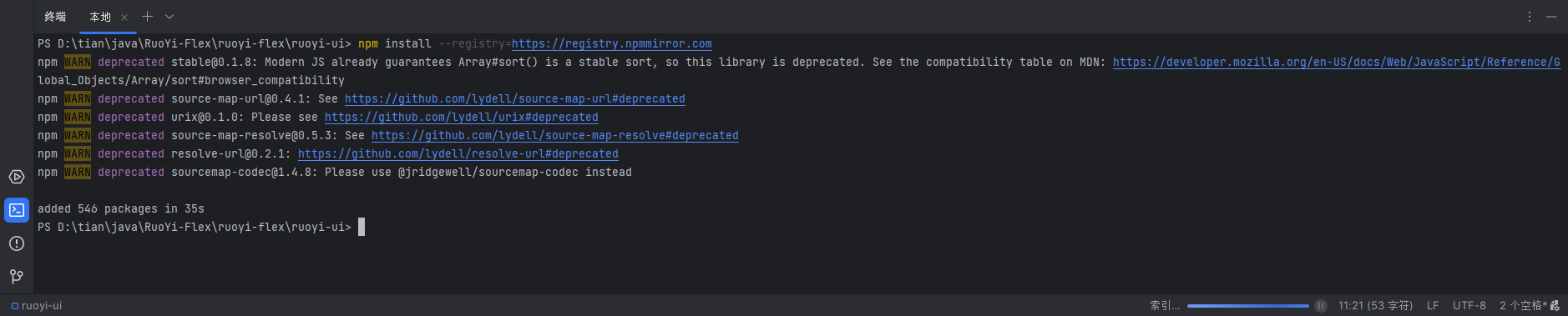
使用WebStorm，选择菜单：文件 -> 打开文件夹 ，选中 flex-elementplus-ui文件夹



### 5.2.2、安装依赖

在WebStorm终端窗口中执行以下命令，安装项目所需依赖：

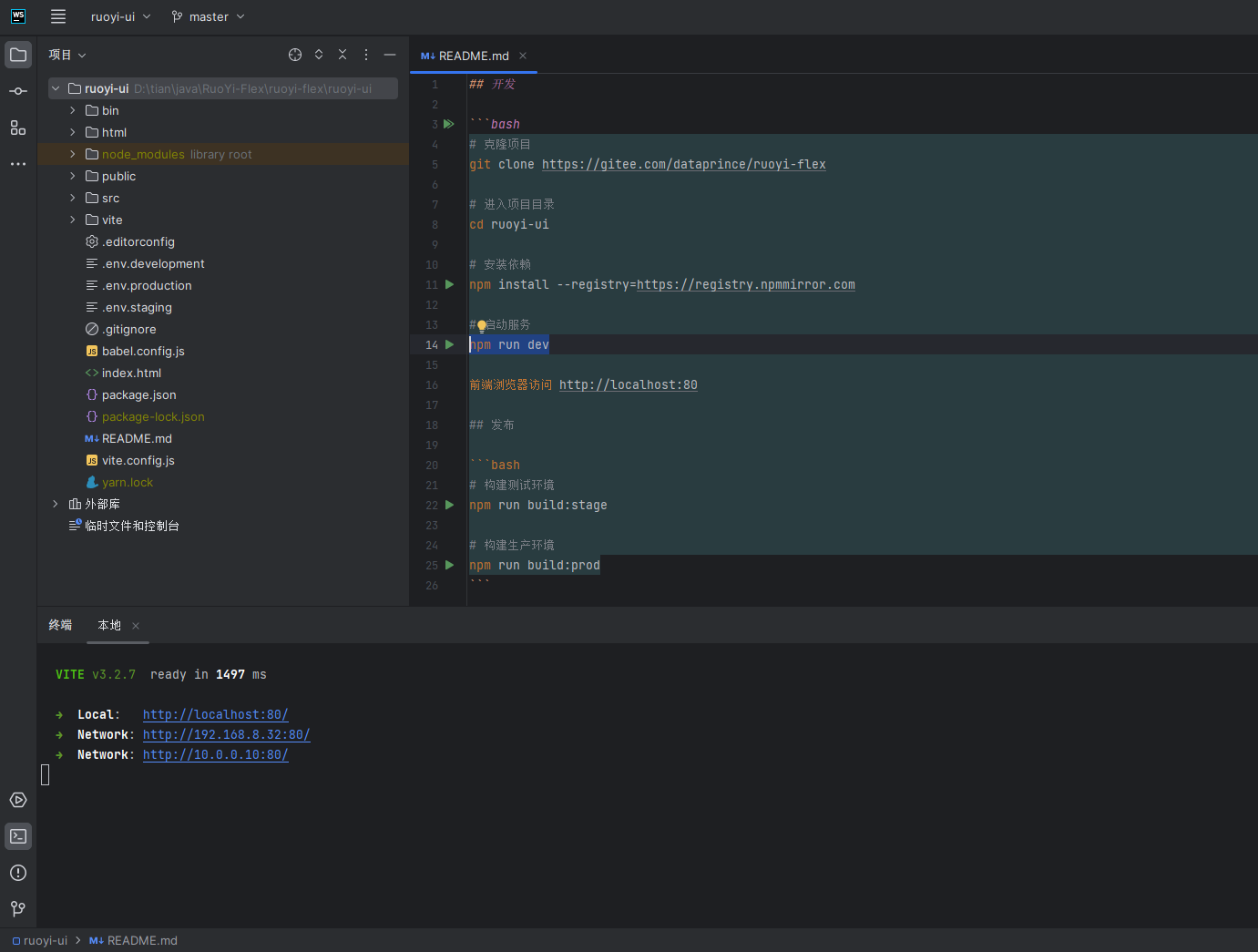
npm install --registry http://registry.npmmirror.com



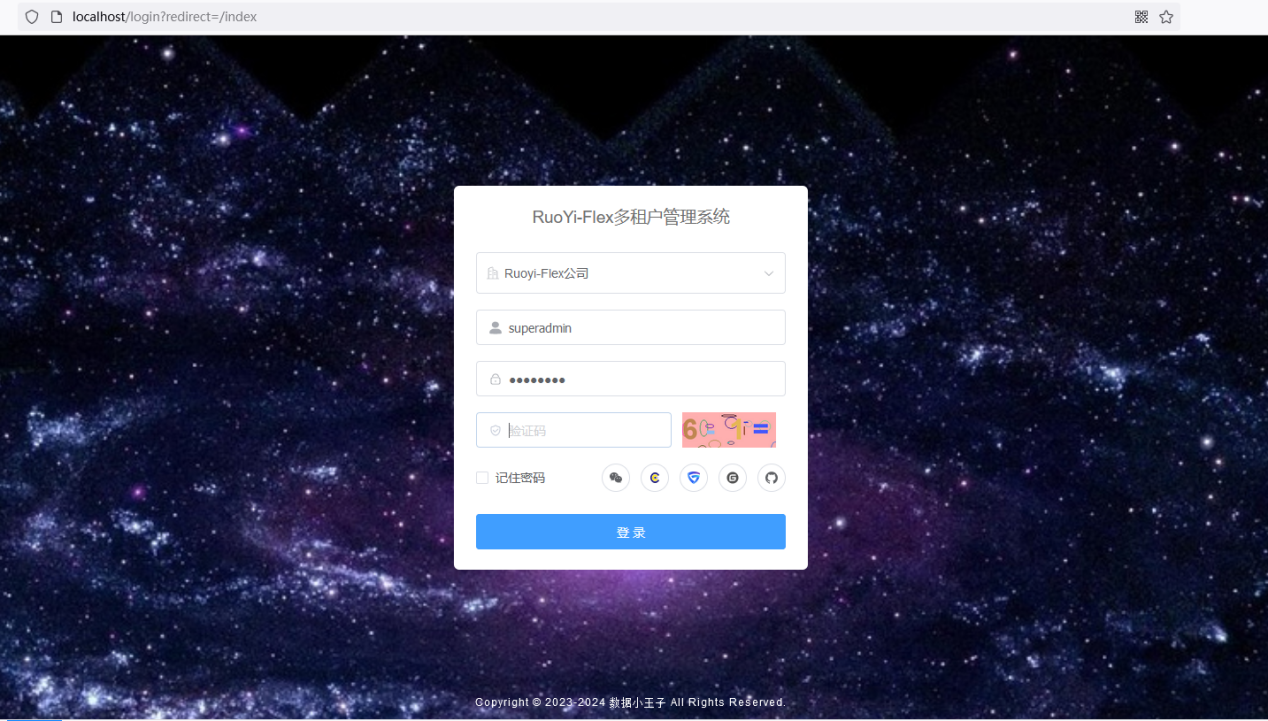
### 5.2.3、运行项目

安装完依赖后，执行以下命令来运行前端项目：

npm run dev



运行成功后，会自动打开浏览器项目登录界面，如下图所示：



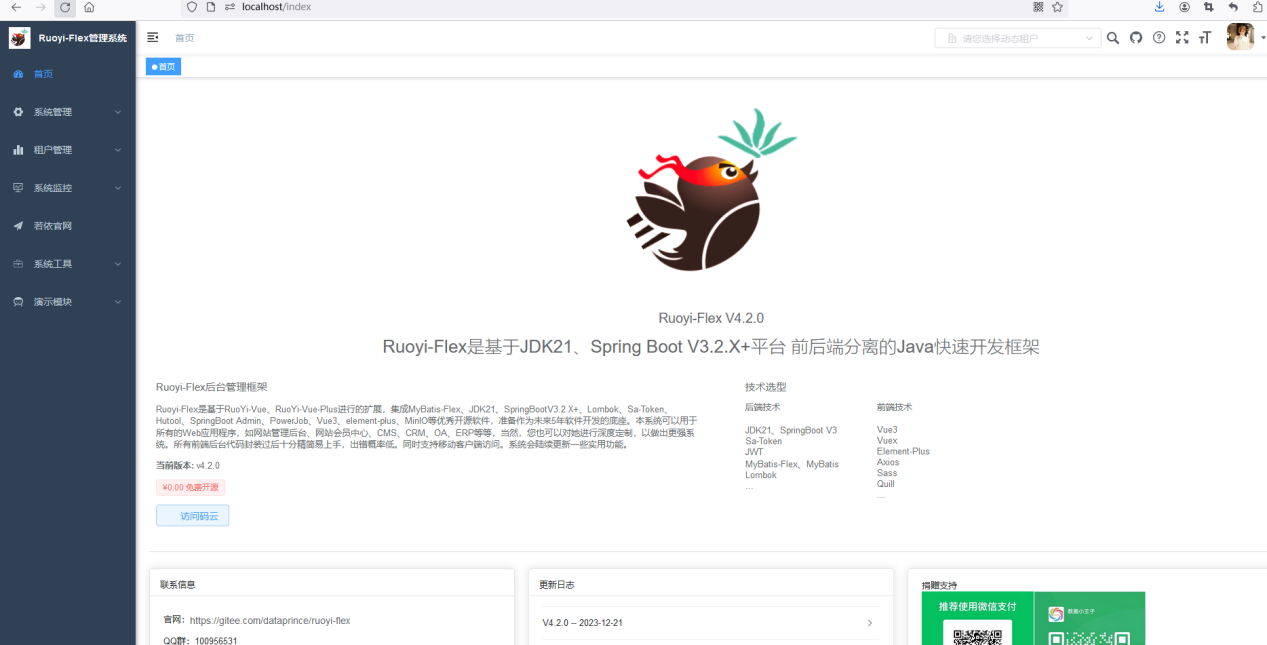
第一个输入框为租户选择框，默认值为“Ruoyi-Flex公司”；

第二个输入框为用户名，默认值为superadmin；

第三个输入框为用户名，默认值为admin123；

第四个输入框为验证码：

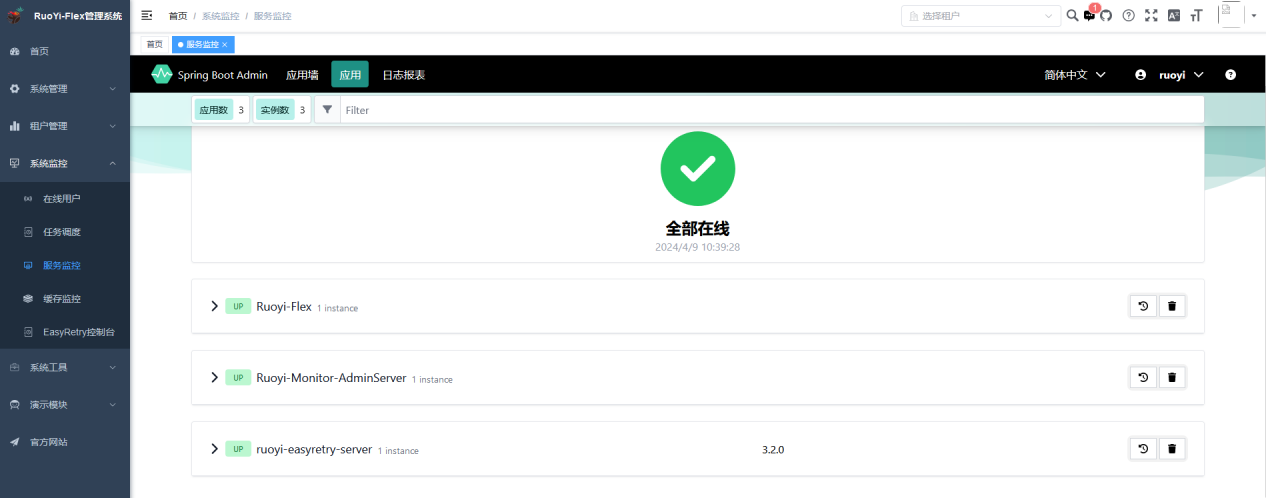
输入验证码，点击“登录”，即可进入系统：



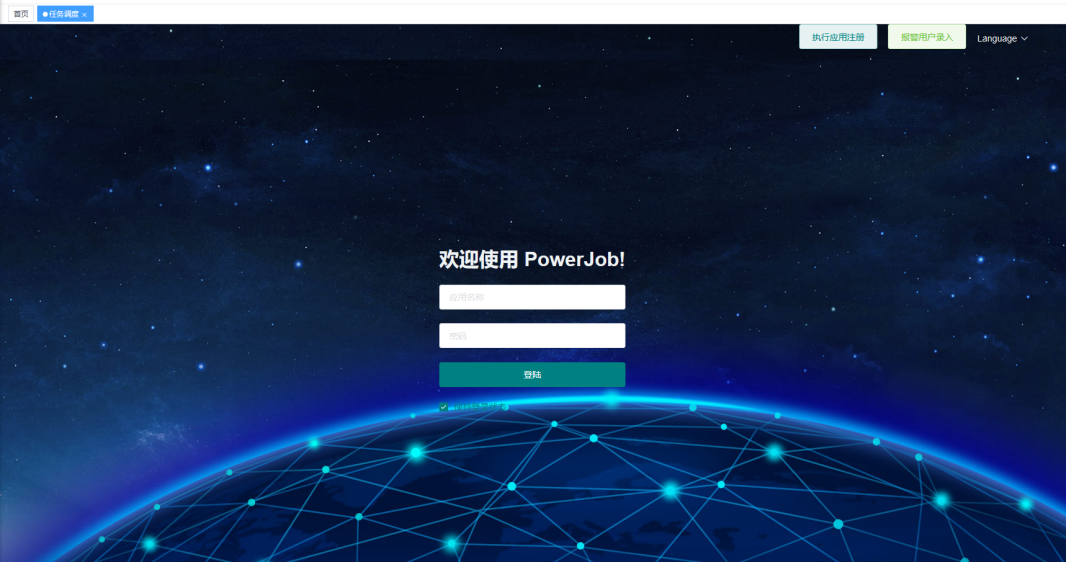
### 5.2.4、服务监控



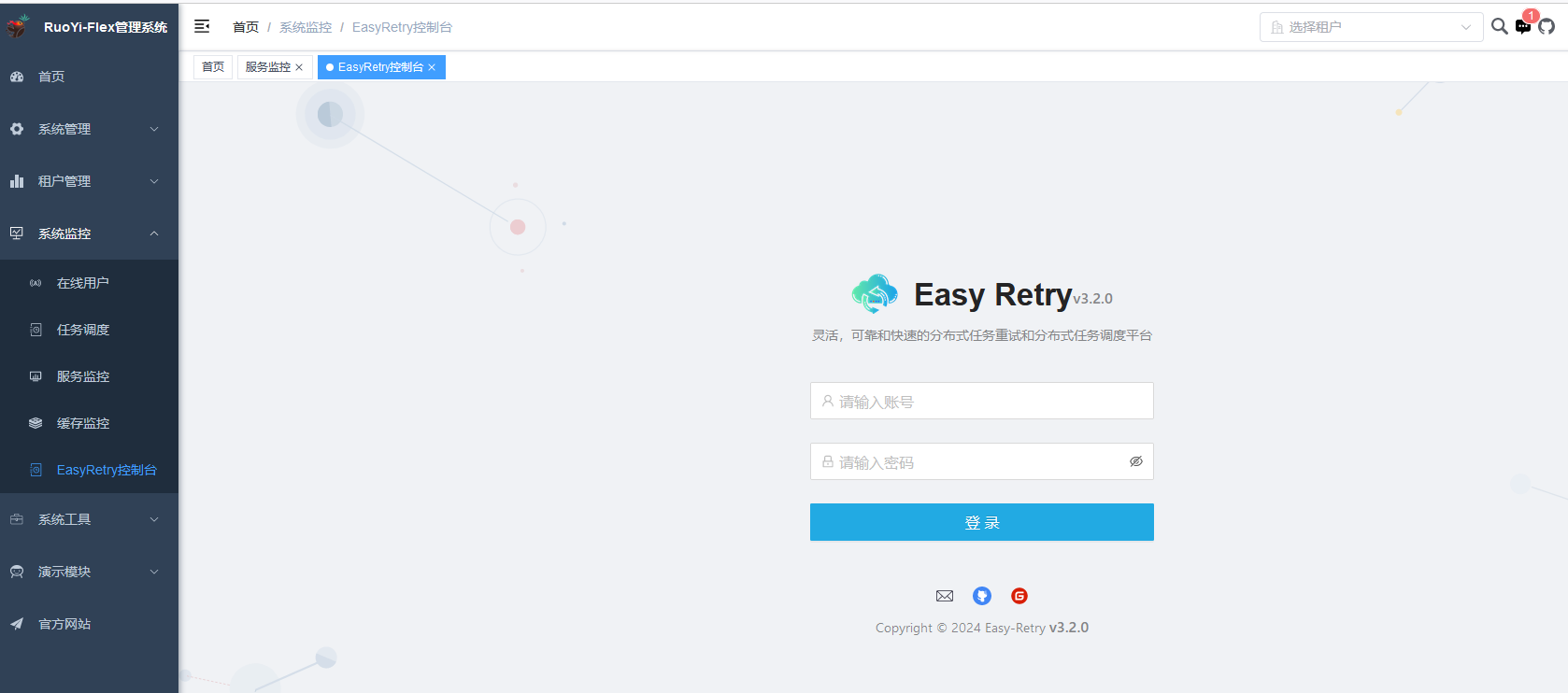
默认监控中心Spring Boot Admin用户名为：ruoyi，密码为：123456



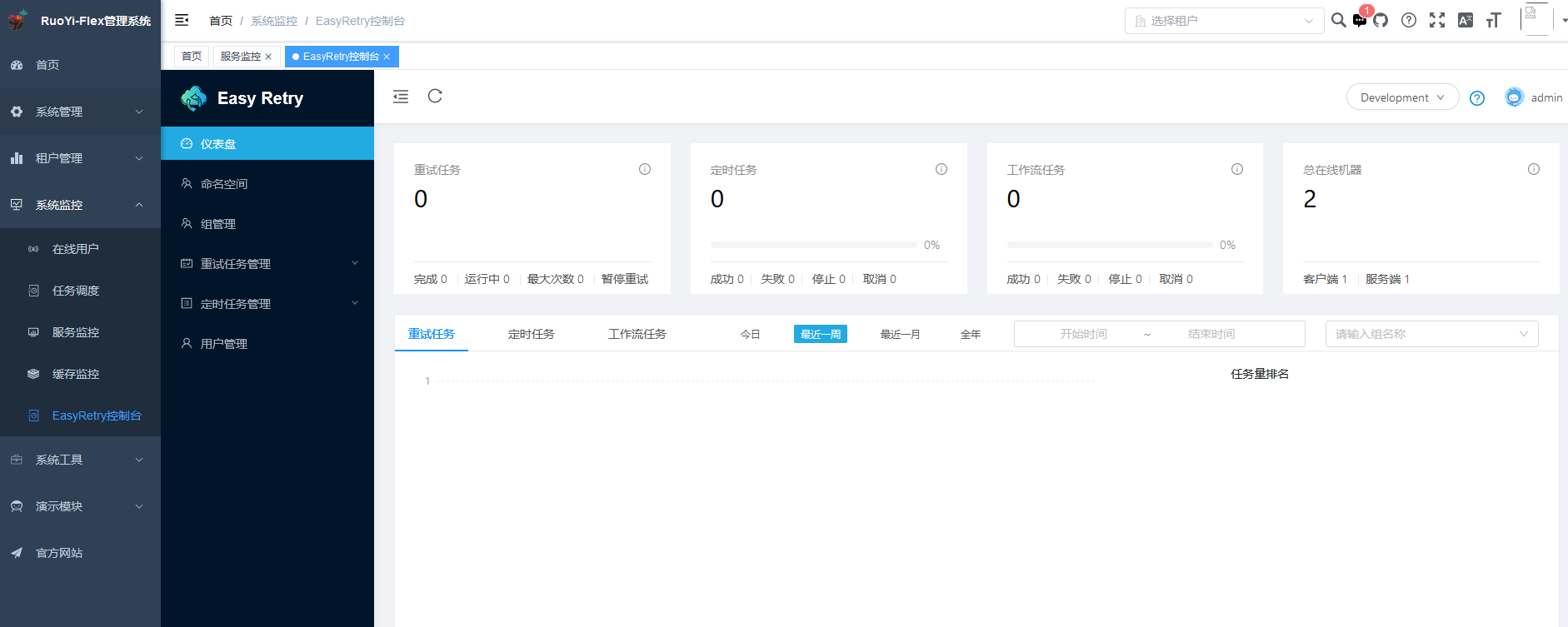
### 5.2.5、任务调度



默认调度中心PowerJob用户名为：ruoyi-worker，密码为：123456



默认EasyRetry控制台用户名为：admin，密码为：admin



# 6、项目结构

Ruoyi-Flex

├─ bin // maven脚本包

├─ doc // 开发文档

├─ image // 系统截图

├─ ruoyi-admin // 入口模块 [8080,9101]

│ └─ RuoYiApplication // 启动类

│ └─ RuoYiServletInitializer // 容器部署初始化类

│ └─ resources // 资源文件

│ └─ i18n/messages.properties // 国际化配置文件

│ └─ application.yml // 框架总配置文件

│ └─ application-dev.yml // 开发环境配置文件

│ └─ application-prod.yml // 生产环境配置文件

│ └─ banner.txt // 框架启动图标

│ └─ logback.xml // 日志配置文件

├─ ruoyi-extra // 额外扩展模块

│ └─ ruoyi-monitor // 监控中心模块 [9090]

│ └─ ruoyi-powerjob-server // 调度中心模块 [7700]

├─ ruoyi-common // 通用模块

│ └─ ruoyi-common-bom // common依赖包管理

│ └─ ruoyi-common-core // 核心模块

│ └─ ruoyi-common-encrypt // 加密模块

│ └─ ruoyi-common-excel // excel模块

│ └─ ruoyi-common-job // 定时任务模块

│ └─ ruoyi-common-json // 序列化模块

│ └─ ruoyi-common-log // 日志模块

│ └─ ruoyi-common-orm // 数据库模块

│ └─ ruoyi-common-oss // 对象存储服务模块

│ └─ ruoyi-common-ratelimiter // 限流功能模块

│ └─ ruoyi-common-redis // 缓存服务模块

│ └─ ruoyi-common-security // 安全认证模块

│ └─ ruoyi-common-springdoc // 系统接口模块

│ └─ ruoyi-common-tenant // 租户模块

│ └─ ruoyi-common-web // web模块

├─ ruoyi-modules // 模块组

│ └─ ruoyi-demo // 演示模块

│ └─ ruoyi-generator // 代码生成模块

│ └─ ruoyi-job // 任务调度服务

│ └─ ruoyi-system // 业务模块

├─ sql // 数据库脚本

├─ .editorconfig // 编辑器编码格式配置

├─ LICENSE // 开源协议

├─ mybatis-flex.config // mybatis-flex参数配置文件

├─ pom.xml // 公共依赖

├─ README.md // 框架说明文件

# 7、演示模块

位于ruoyi-flex\ruoyi-modules\ruoyi-demo是系统内置的演示模块。

演示模块包含两套风格的示例：

（1）传统mybatis风格

这是为了兼容历史mybatis风格代码而设的，毕竟原来使用mybatis开发了大量的代码，如果使用mybatis-flex重构的话，费时费力，意义不大，直接拿来历史代码运行即可。

位于com.ruoyi.demo包下的单表、树表、主子表是纯mybatis的代码，没有引入mybatis-flex，为了兼容历史代码而设的，使用pagehelper分页，xml文件中有SQL代码。

（2）mybatis-flex风格

com.ruoyi.mf包下的单表、树表、主子表是纯mybatis-flex的代码，不再使用pagehelper分页，xml文件中也没有SQL代码。

# 8、代码生成

自V4.2.0版本开始，代码生成模块仅仅生成mybatis-flex风格代码，不再生成mybatis风格代码。

代码生成模块支持的数据库表结构，必须含有下列公共字段：tenant\_id、version、create\_by、create\_time、update\_by、update\_time，树表还得加上parent\_id、ancestors、order\_num三个公共字段。

具体SQL建表语句可以看考“演示模块”的例子。