如何下载、部署、运行华容道项目

本文章默认你什么都不会,将指导你如何成功运行这个项目

1. 如何创建本项目所需的数据库?

1.1 下载配置MySQL和DataGrip

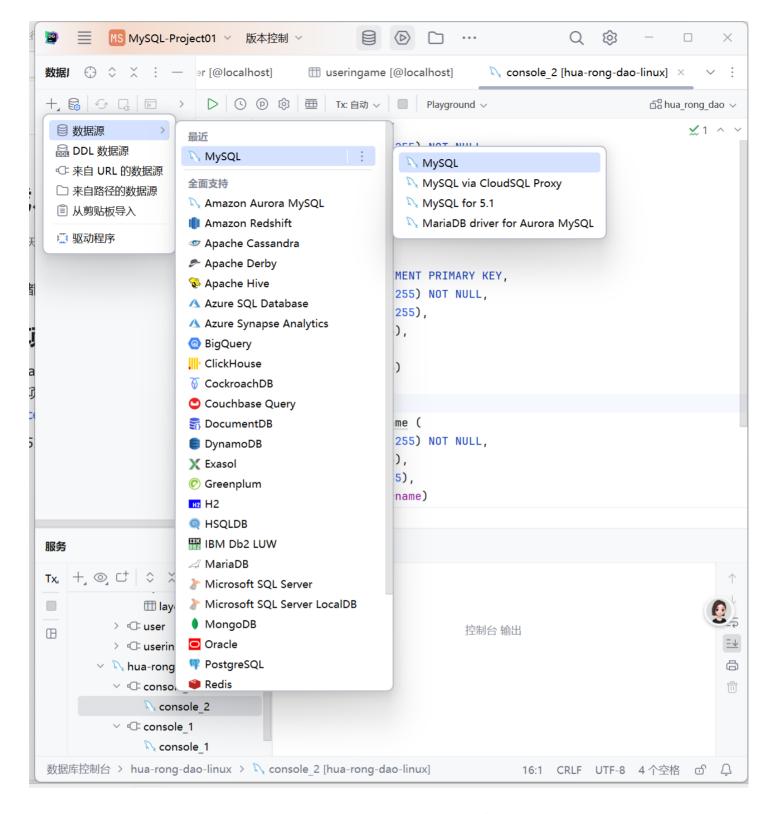
【全网首发AI+JavaWeb开发入门,Tlias教学管理系统项目实战全套视频教程,从需求分析、设计、前后端开发、测试、程序优化到项目部署一套搞定】

https://www.bilibili.com/video/BV1yGydYEE3H/?

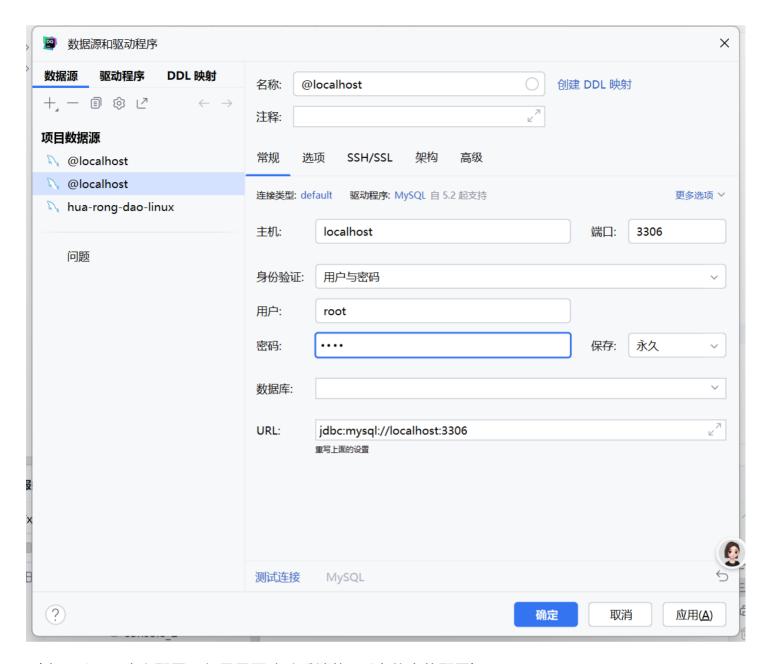
p=51&share_source=copy_web&vd_source=1eb3ec14476ecc498848d58cf48a175d

请参照这个教程的51集下载MySQL,52集下载DataGrip。

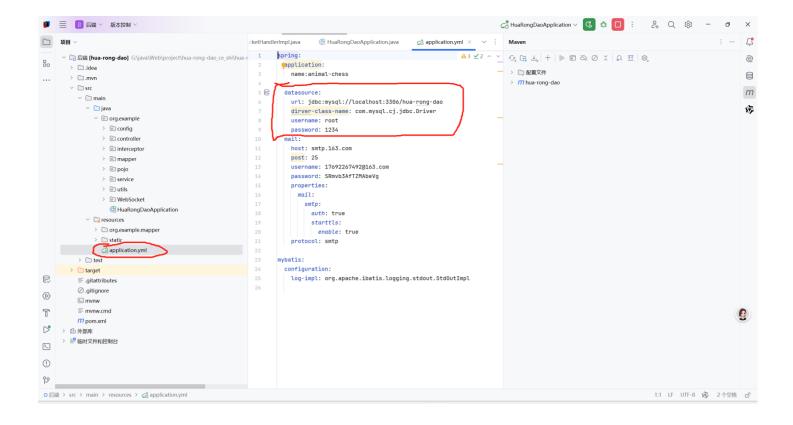
1.2 创建一个项目,然后点击左侧的加号,点击数据源,点击MySQL,点击右侧MySQL



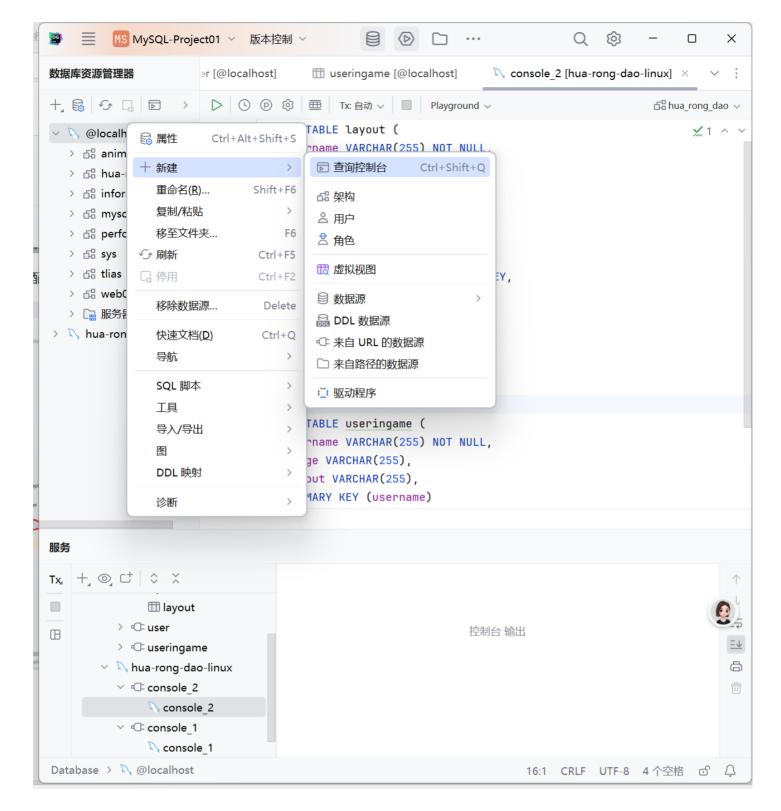
1.3 创建一个名叫@localhost,用户是root,密码为1234的数据库



(也可以不用这么配置,但是需要改动后端的yml文件中的配置)



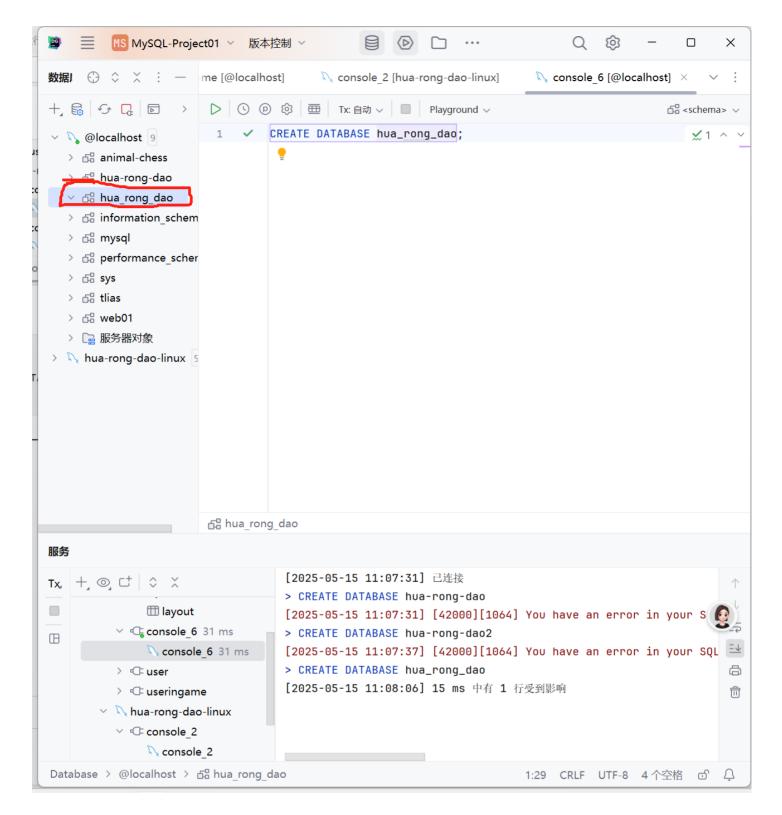
1.4 然后右键点击左侧@localhost选择新建,选择查询控制台,



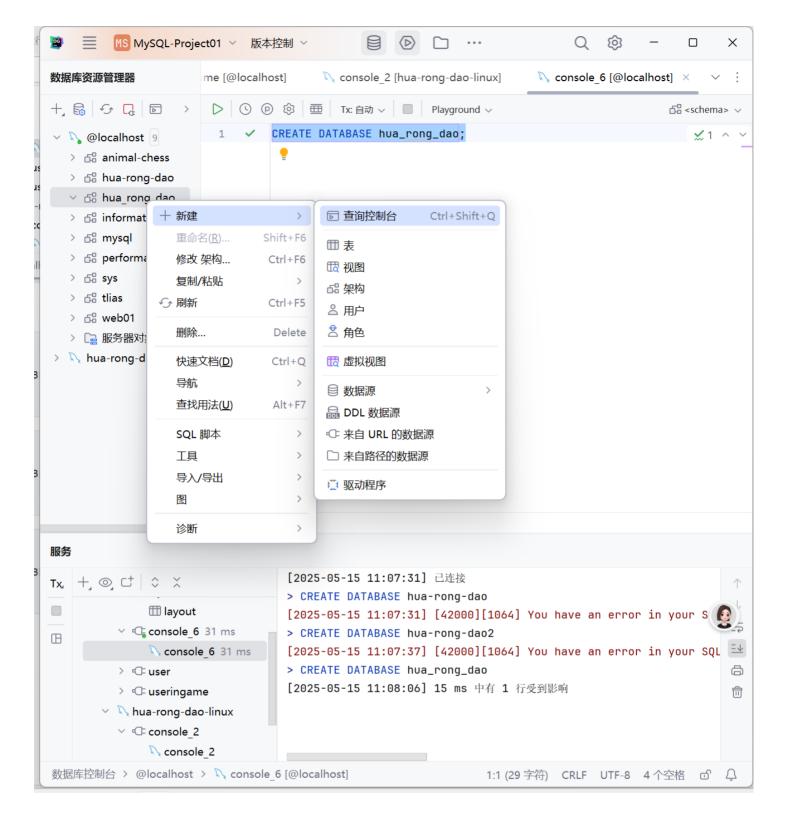
1.5 在控制台中输入

```
代码块
1 CREATE DATABASE hua_rong_dao;
```

运行后应该左侧有一个名叫hua_rong_dao的数据库(下图中其他的数据库不用在意,那些是我之前自己创建的其他数据库,在本项目中用不到)



1.6 创建查询控制台



1.7 然后输入建表语句

代码块

CREATE TABLE layout (username VARCHAR(255) NOT NULL, count INT, layout VARCHAR(255), PRIMARY KEY (username));

代码块

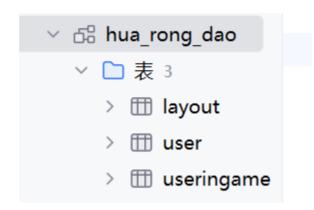
1 CREATE TABLE user (id INT AUTO_INCREMENT PRIMARY KEY, username VARCHAR(255) NOT NULL, password VARCHAR(255), email VARCHAR(255), award_count INT, image

```
VARCHAR(255) );
```

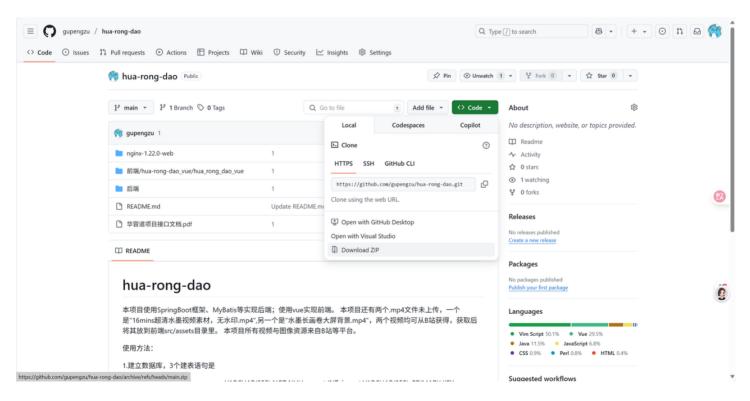
代码块

1 CREATE TABLE useringame (username VARCHAR(255) NOT NULL, image VARCHAR(255), layout VARCHAR(255), PRIMARY KEY (username));

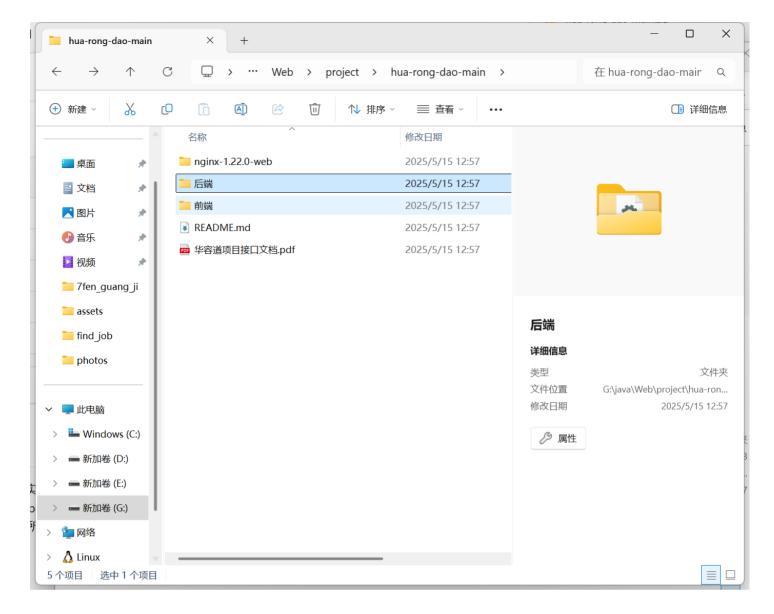
运行后



2. 下载GitHub上的项目并下载必要插件



点击Code,然后点击Download ZIP,将下载的压缩包解压在一个文件夹中,右键后端,用idea打开



Maven的安装

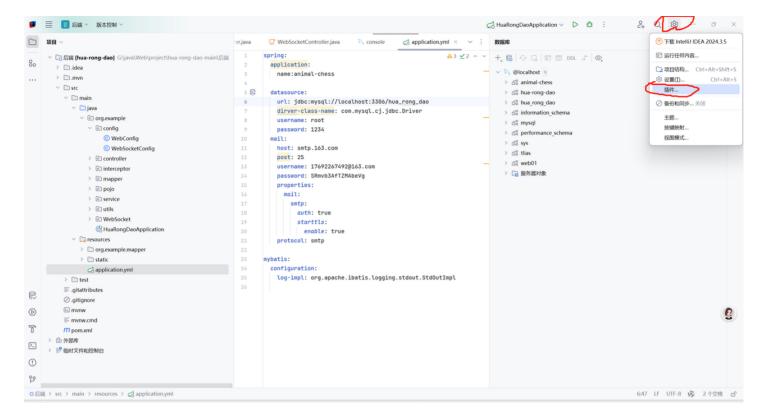
【全网首发AI+JavaWeb开发入门,Tlias教学管理系统项目实战全套视频教程,从需求分析、设计、前后端开发、测试、程序优化到项目部署一套搞定】

https://www.bilibili.com/video/BV1yGydYEE3H/?

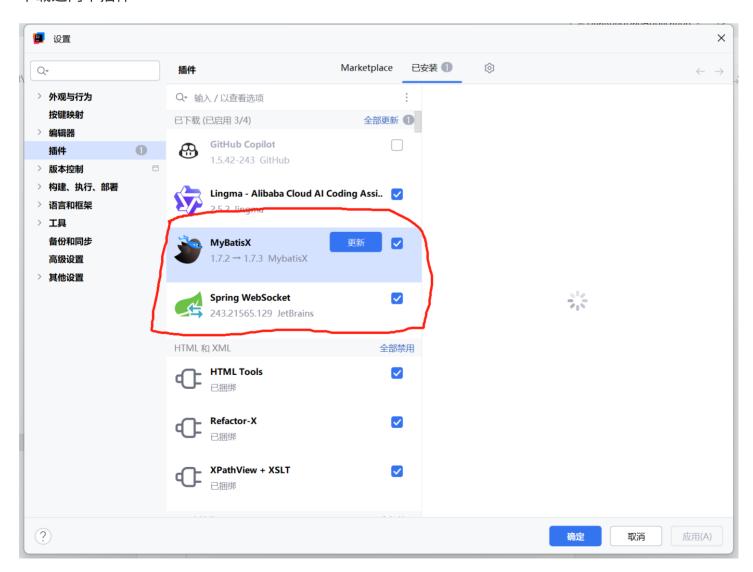
p=32&share_source=copy_web&vd_source=1eb3ec14476ecc498848d58cf48a175d

按照这个视频里的讲解进行安装。

MyBatis和WebSocket插件安装

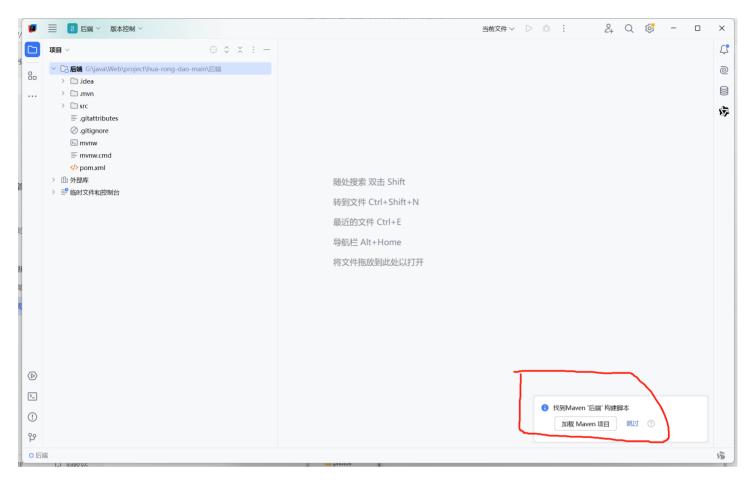


下载这两个插件



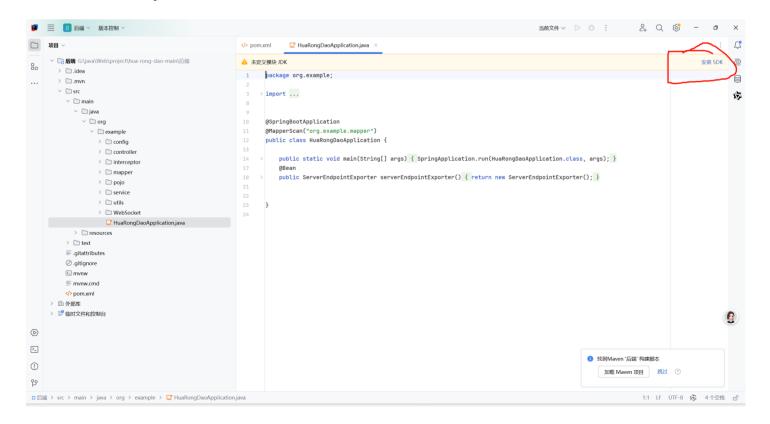
3. 后端配置与调整

加载Maven项目



加载后的效果是xml文件中基本没有红色报错(还是有少数几个报错)

然后随便打开一个java后缀的文件,安装SDK,选择17



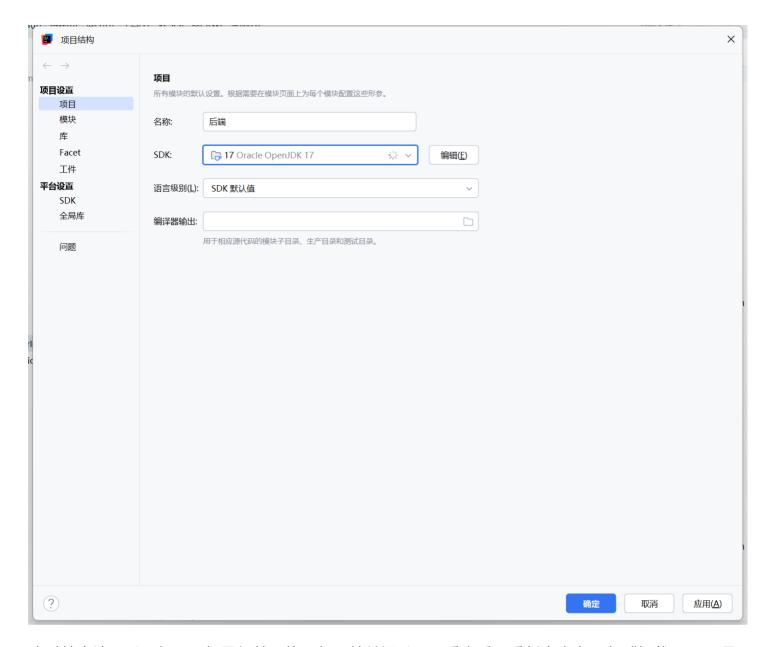
然后打开项目结构

```
文件(E) 编辑(E) 视图(M) 导航(M) 代码(C) 重构(B) 构建(B) 运行(U) 工具(I) VCS(S) 窗口(M) 帮助(H)
新建(N)
                                                                                                                                                                                                                       Ą
      □ 打开(Q)...
                                                                package org.example.WebSocket;
                                                                                                                                                                                                                       @
        最近的项目(R)
         关闭项目(1)
                                                                                                                                                                                                                       远程开发...
                                                                                                                                                                                                                       Š
                                                                       public class WebSocketHandlerImpl extends TextWebSocketHandler {
       设置(I).
                                                                           private final Map<String, WebSocketSession> sessions = new ConcurrentHashMap<>();
                        trl+Alt+Shift+S
                                                                            public void afterConnectionEstablished(WebSocketSession session) throws Exception {
        本地历史记录(H)
                                                                                String username = (String) session.getAttributes().get("username");
      凹 全部保存(S)
                              Ctrl+S
                                                                                System.out.println("WebSocket 连接开启: " + username):

← 从磁盘全部重新加载

                           Ctrl+Alt+Y
                                                                                sessions.put(username, session);
        修复 IDE
        清除缓存..
                                                                            @Override
        等理 IDE 设置
                                                                            protected void handleTextMessage(WebSocketSession session, TextMessage message) throws Exception {
        新建项目设置
                                                                                String payload = message.getPayload();
System.out.println("WebSocket 收到消息: " + payload);
        将文件另存为模板(L)...
                                     andlerImpLiava
        5.44
                                      plication.java
      高 打印(P)...
                                                                                for (WebSocketSession s : sessions.values()) {
        省电模式
                                                                                    System.out.println("}ffWebSocket session: " + s.getId() + ", isOpen: " + s.isOpen());
         退出(X)
                                                                                    if (s.isOpen()) {
    System.out.println("发送消息给: " + s.getId());
          ⊘ .gitignore
           E mvnw
                                                                                        s.sendMessage(new TextMessage(payload));
           = mvnw.cmd
      D
                                                                            public void afterConnectionClosed(WebSocketSession session, CloseStatus status) throws Exception {
>_
                                                                                String username = (String) session.getAttributes().get("username");
System.out.println("WebSocket 连接关闭: " + username);
(!)
စ္
□后端 > src > main > java > org > example > WebSocket > ☑ WebSocketHandl
```

把SDK改为17



这时就应该可以运行了,如果仍然不能运行,就关闭idea,重启后,重新点击右下角"加载Maven项目"

```
■ 1 后端 ~ 版本控制

⟨ HuaRongDaoApplication.java 

⟨ ○ WebSocketHandlerImpl.java

    AliyunOSSOperator.java

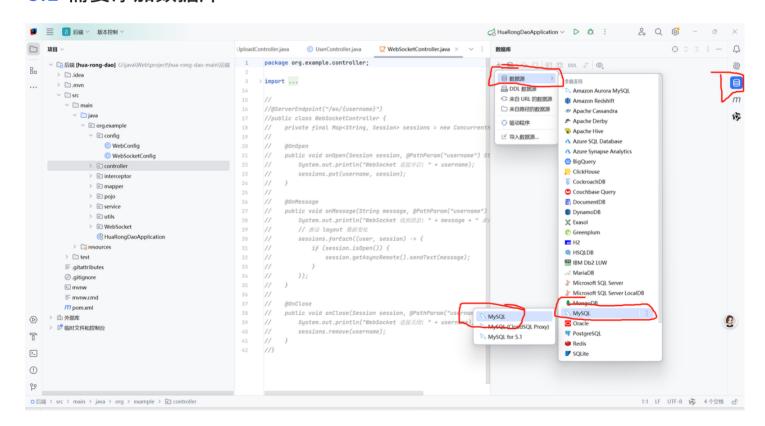
                                                                                                                                                               © JWTUtils.java
                                                                                                                                                                                                  Ų
                                                                 package org.example:
                                                                                                                                                                                                   @
       ) idea
                                                               > import ...
                                                                                                                                                                                                   > mvn.mvn
                                                                                                                                                                                                  m
         ∨ 🗀 mair
                                                                 @SpringBootApplication
           ∨ Diava
                                                                                                                                                                                                   Š
                                                                 @MapperScan("org.example.mapper")
              ∨ ⊚ org.example
                                                          12
                                                                 public class HuaRongDaoApplication {
                > @ controller
                                                                     public static void main(String[] args) { SpringApplication.run(HuaRongDaoApplication.class, args); }
                > intercepto
                > @ mapper
                                                                    public ServerEndpointExporter serverEndpointExporter() { return new ServerEndpointExporter(): }
                > 💿 pojo
                > @ service
                  utils
               ( HuaRong Dao Application
         > 🗀 test

    □ .gitattributes

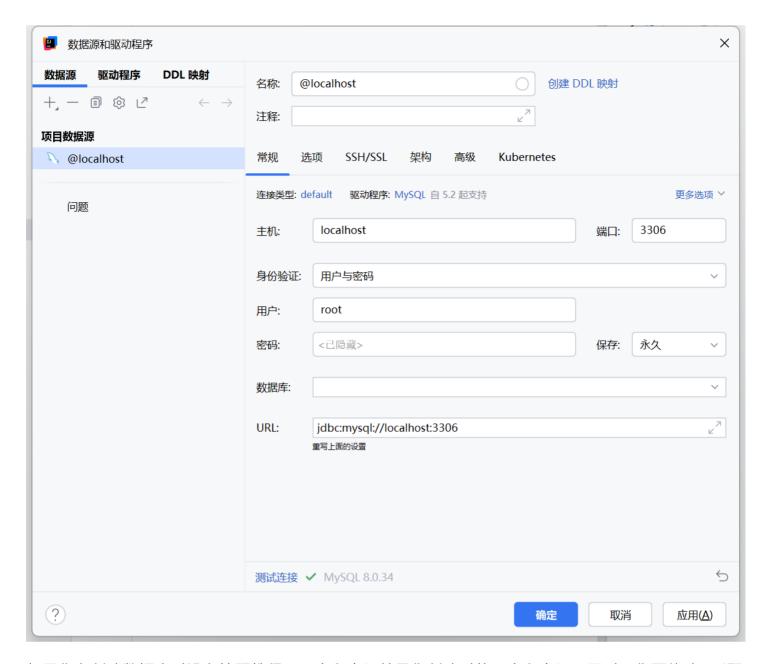
         aitianore
         ≡ mvnw.cmd
         m nom vml
     > 🚅 临时文件和控制台
D
T
>_
(!)
ဗ္
□后端 > src > main > java > org > example > 億 HuaRongDaoApplication
```

但是运行后仍然会有问题。

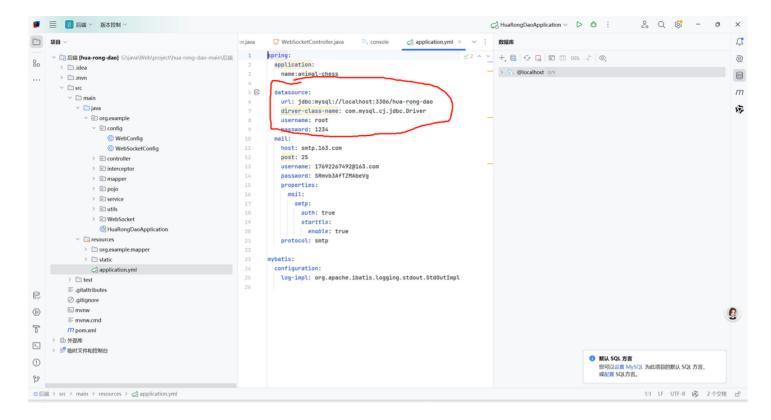
3.1 需要添加数据库



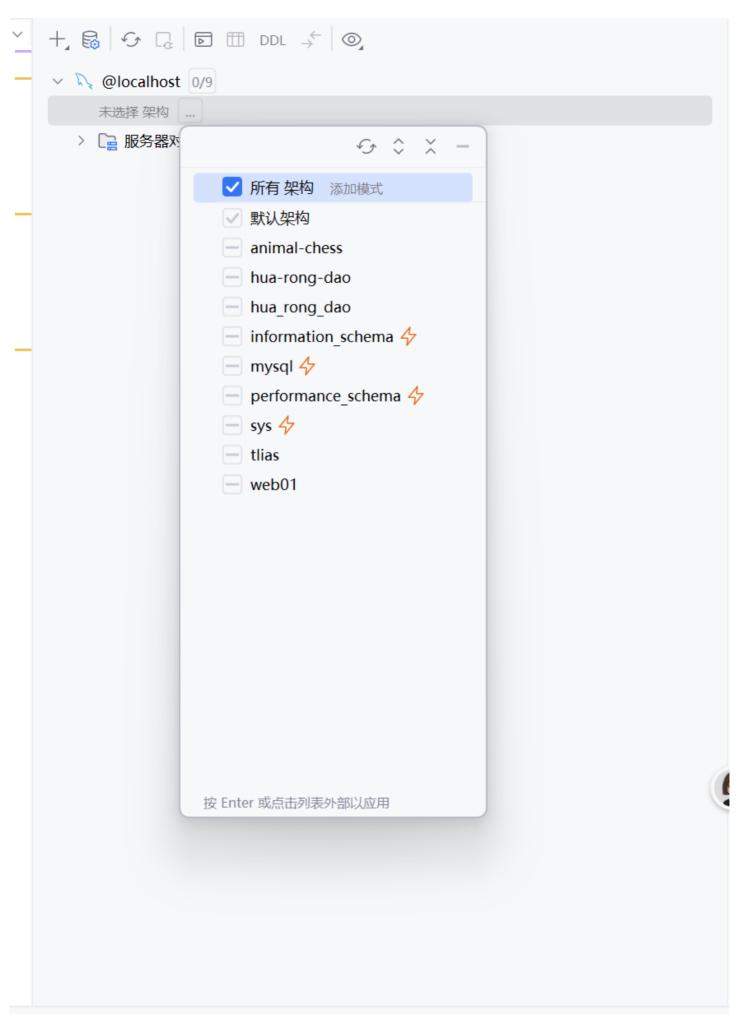
如果你在创建数据库时按照教程创建的,这里用户就是root,密码是1234.



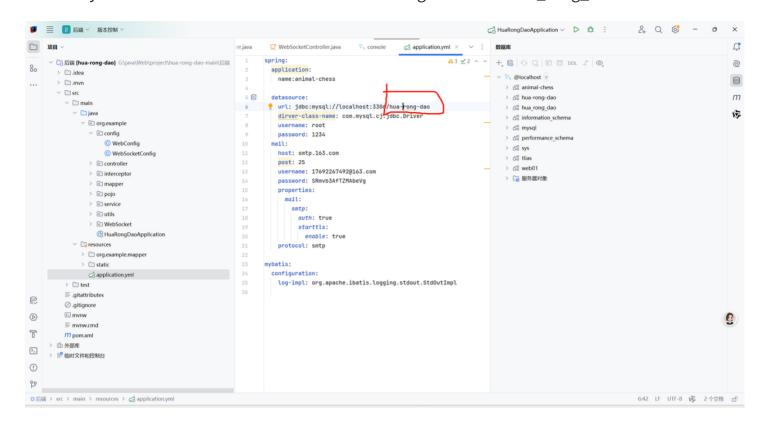
如果你在创建数据库时没有按照教程,用户和密码就是你创建时的用户和密码。同时,你要修改yml配置文件中的内容



顺便提醒,数据库中要选择所有架构才能看到你创建的数据库。



然后修改yml文件中的数据库名称。(把图中的hua-rong-dao修改为hua_rong_dao)



3.2 修改阿里云OSS配置

【全网首发AI+JavaWeb开发入门,Tlias教学管理系统项目实战全套视频教程,从需求分析、设计、前后端开发、测试、程序优化到项目部署一套搞定】

https://www.bilibili.com/video/BV1yGydYEE3H/?

p=106&share_source=copy_web&vd_source=1eb3ec14476ecc498848d58cf48a175d

请看106、107、、108、109节内容,把阿里云配置改为自己的阿里云。

当然你也可以跳过这个修改阿里云OSS配置这一步,但是博主仅在其中充值2元,如果2元耗尽,就无法继续使用。

3.3 修改yml中邮箱即密码

这个yml中配置的邮箱是用来在注册时发送验证码的,项目中是博主自己的邮箱。在yml中配置后,发送验证码时就会由博主的邮箱发送到目标邮箱。

【最新全网首发项目实战-SpringBoot注册登录及邮箱发送账号激活,涨薪必备】

https://www.bilibili.com/video/BV1eA411V7jj/?

share_source=copy_web&vd_source=1eb3ec14476ecc498848d58cf48a175d

当然这一步你也可以跳过,但是请不要过度使用博主的邮箱,发送次数超过一定上限后就会无法使 用。

前端准备

文件准备

前端有两个过大的文件无法上传,

【水墨长画卷大屏背景】 https://www.bilibili.com/video/BV1ab411b716/? share source=copy web&vd source=1eb3ec14476ecc498848d58cf48a175d

【超清水墨视频素材,无水印】 https://www.bilibili.com/video/BV1go4y1C7uz/? share source=copy web&vd source=1eb3ec14476ecc498848d58cf48a175d

用bilibili相关插件(edge有插件可以下载哔哩哔哩的视频内容)等方法下载这两个文件。

并分别命名为:

水墨长画卷大屏背景.mp4

16mins超清水墨视频素材,无水印.mp4

然后放到

hua-rong-dao-main\前端\hua-rong-dao_vue\hua_rong_dao_vue\src\assets文件夹里

npm的配置和准备

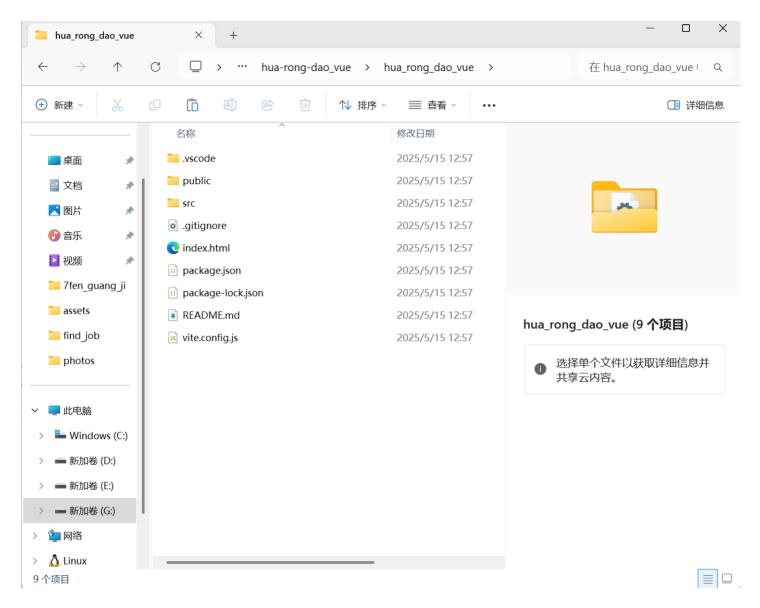
【全网首发AI+JavaWeb开发入门,Tlias教学管理系统项目实战全套视频教程,从需求分析、设计、前后端开发、测试、程序优化到项目部署一套搞定】

https://www.bilibili.com/video/BV1yGydYEE3H/?

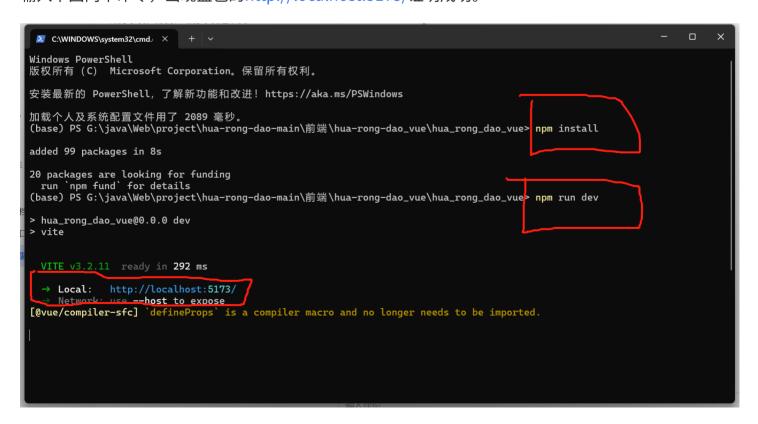
p=153&share_source=copy_web&vd_source=1eb3ec14476ecc498848d58cf48a175d

先照着这个视频配置好npm,然后就能在终端使用npm install等命令。

然后在下图这个文件夹里右键打开终端



输入下图两个命令,出现蓝色的http://localhost:5173/证明成功。



然后就能访问localhost:5173(ctrl加左键点击图中蓝色的localhost:5173)(这里有个小bug,刚进入时应该先点击右上角的退出按钮才能正常使用)。

大概这样

