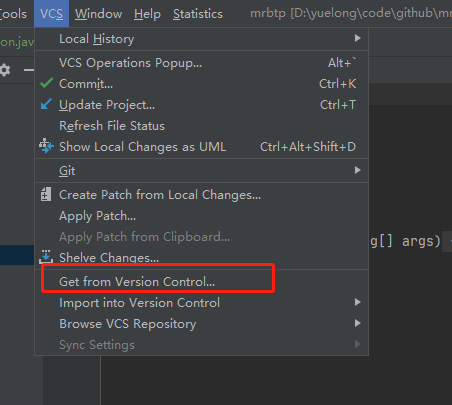
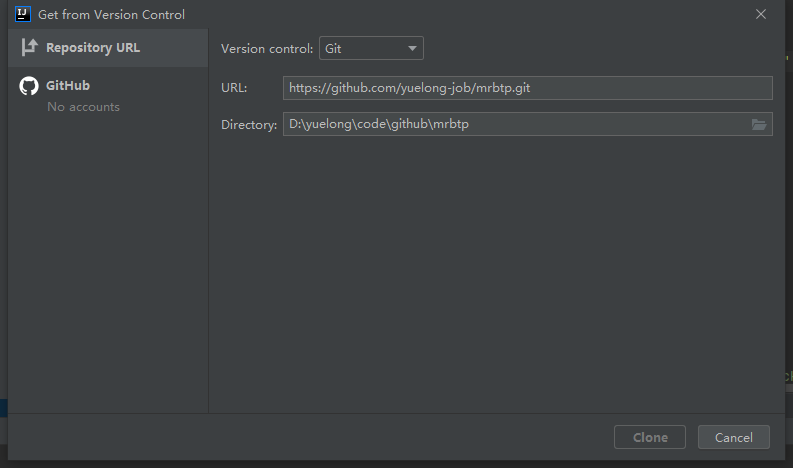
1. github导入项目

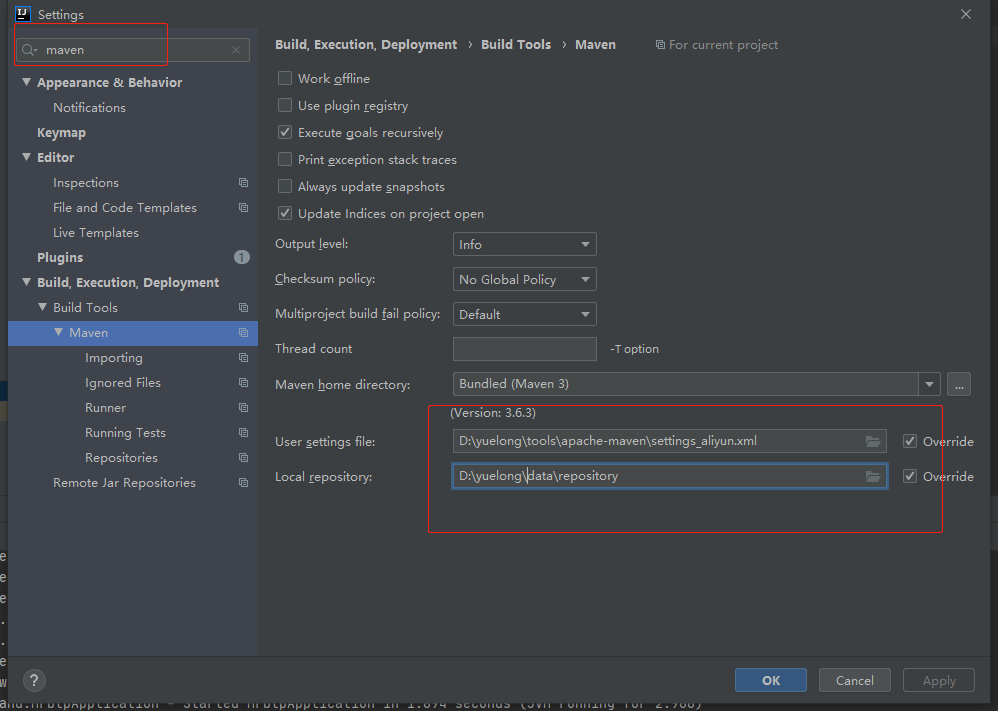
本文档都是以 开发工具intellij idea截图

<https://github.com/yuelong-job/mrbtp.git>





1. 配置maven 拉取jar包



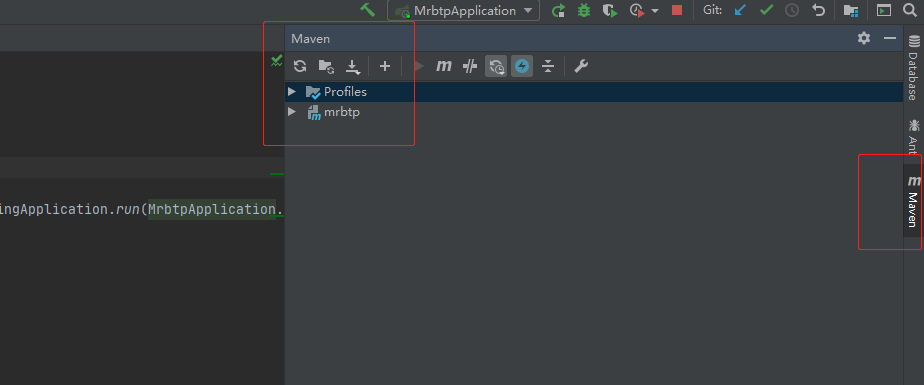
这里需要一个settins.xml文件 一般配置maven仓库 因为maven官方仓库在国外，国内使用阿里巴巴的仓库

下面一个是gf的仓库一个是aliyun的仓库 看网络情况使用

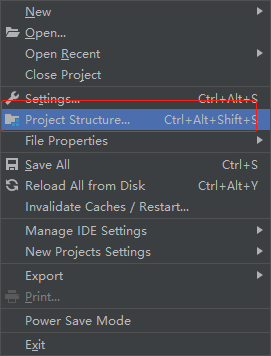


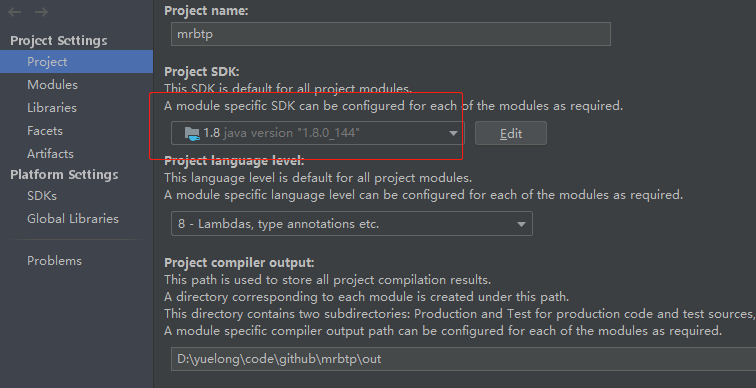
Local repository是下载jar包存放本地的路径 找个大点的盘

完成配置后 点击更新 下载jar



然后记得配置jdk





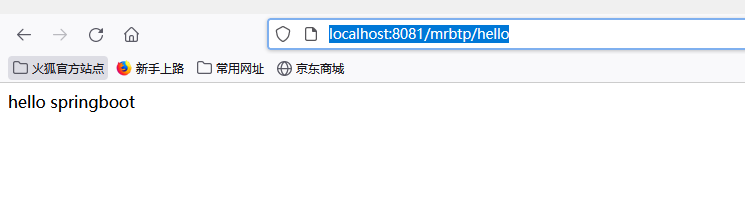
1. 启动项目

执行 com.zoneland. MrbtpApplication 的 main方法 启动项目

由于我们什么都没写 这时候启动项目 只能看到启动成功

4 完成 HelloWorldController

然后启动项目 这时候在浏览器敲<http://localhost:8081/mrbtp/hello>



Springboot 搭建java-web后端完成

以上4步 需要掌握

a.学会使用 开发工具intellij idea ，要学会拉取github上面的项目，学会配置jdk 学会配置maven

b.学会如何快速搭建springboot项目 springboot内部知识点很多 一般工作七八年的以后才去研究 ，初学者 会使用就行和知道怎么启动就可以

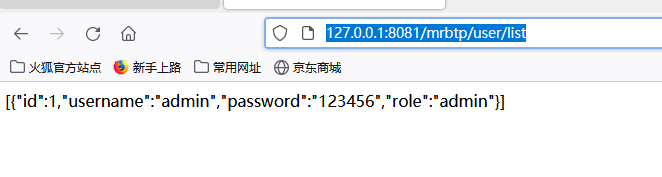
本项目我已经创建好了 直接github下载就行 如果自己要快速创建项目访问 <https://start.spring.io>快速创建

1. Maven框架也是，达到会使用 会配置 会下载和管理jar包
2. Springboot的Controller层的写法 要知道 @RestController @GetMapping @PostMapping 注解的含义
3. Mysql和数据库连接框架

执行init.sql 创建表t\_user表

<http://127.0.0.1:8081/mrbtp/user/list>

返回



连接数据库完成

Mybatis-plus 主要分 mapeer.java 和mapper.xml 两个文件

DAO/Mapper层：负责访问数据库进行数据的操作，取得结果集，之后将结果集中的数据取出封装到VO类对象之后返回给service层。

service层：业务层，用来实现业务逻辑。能调用dao/mapper层

Cotroller层：叫做控制层，主要的功能是处理用户发送的请求。主要处理外部请求。调用service层，返回json数据（http接口，前后分离使用）或者返回页面-view层

View层：叫做显示层，主要是负责现实数据。（view层可以在一个项目里 传统的就是jsp 也可以是springboot+thymeleaf 或者view层独立出来 前后分离）

1. 引入thymeleaf引擎

目前主流都是前后分离架构，也就是前端页面和后端服务器分开写，前端程序员专门写前端，后端程序员专门写后端，前端主流使用node.js环境下，主流为vue.js react.js等

后端java go c 都可以

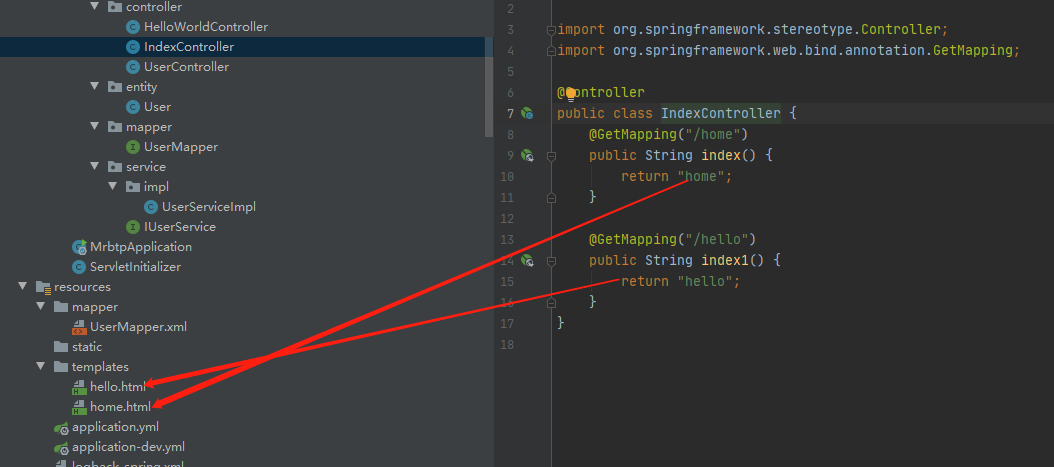
Springboot微服务目前是java后端架构里主流框架

小型项目也可以引入thymeleaf 在springboot里加页面

首先pom引入

<!--引入thymeleaf依赖-->  
<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-thymeleaf</artifactId>  
</dependency>

然后写controllor



@RestController @Controller 两者区别 弄清楚 然后就能理解  
为何<http://127.0.0.1:8081/mrbtp/user/list>返回的是json 而<http://localhost:8081/mrbtp/home> 跳转了页面

1. 等待更新