创建一个springboot项目

Group com.TianzeJ9

## //一般是项目名或者模块名

Artifact 项目名-版本

一些解释

<!-- html注释的内容 css5.com.cn -->

#yml的注释 数据库的注释也是#

Resource 里创建 application.yml

右键海豚 open consle 打开数据库的控制台

主程序运行在 java.com.xx.xx的xxApplication中

用thymeleaf html文件第一行得改成

<html lang="en" xmlns:th="http://www.thymeleaf.org">

Lombok的三个注释

@Data  
@AllArgsConstructor  
@NoArgsConstructor

(写实体类的时候加入 有参构造 无参构造 get set tostring hashcode equals方法)

Springboot的常用注释

//localhost:8080/login 将请求响应到/login网站上

@GetMapping("/login")

连接mysql数据库(最右边的database)

root

1234

jdbc:mysql://localhost:3306?serverTimezone=UTC(设置时区)

//操作数据表 数据表得加进来

右键数据库的海豚 Open query consle 操作数据库

<!--Mybatis Plus 依赖

https://mvnrepository.com/artifact/com.baomidou/mybatis-plus-boot-starter -->

<dependency>

<groupId>com.baomidou</groupId>

<artifactId>mybatis-plus-boot-starter</artifactId>

<version>3.4.2</version>

</dependency>

<!--mysql依赖

https://mvnrepository.com/artifact/mysql/mysql-connector-java -->

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>8.0.23</version>

</dependency>

<dependency>  
 <groupId>org.projectlombok</groupId>  
 <artifactId>lombok</artifactId>  
 <optional>true</optional>  
</dependency>

<!-- (代码生成器的依赖) -->

<dependency>

<groupId>com.baomidou</groupId>

<artifactId>mybatis-plus-generator</artifactId>

<version>3.4.1</version></dependency>

<!-- https://mvnrepository.com/artifact/org.thymeleaf/thymeleaf (模板引擎依赖)  
-->  
<!-- 这个是springboot默认集成的thymeleaf依赖 不能改版本 用了thymeleaf依赖可以把web依赖去了 默认自带了  
<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-thymeleaf</artifactId>  
</dependency>  
<!- https://mvnrepository.com/artifact/org.thymeleaf/thymeleaf  
这个是官网集成的依赖 可以改thymeleaf各个版本-->  
<dependency>  
 <groupId>org.thymeleaf</groupId>  
 <artifactId>thymeleaf</artifactId>  
 <version>3.0.11.RELEASE</version>  
</dependency>

(swagger依赖 )

spring:  
 datasource:  
 username: root  
 #密码根据自己的密码改  
 password: 123456  
 #数据库名字根据自己的改  
 url: jdbc:mysql://localhost:3306/mf?serverTimezone=UTC&useUnicode=true&characterEncoding=utf-8  
 driver-class-name: com.mysql.cj.jdbc.Driver  
 thymeleaf:  
 *#放弃开启缓存 这样后台更新代码会立刻显示新的网页* cache: false

*#整合mybatis*mybatis:

#实体类的包指向mybatis  
 type-aliases-package: com.mf.project01.pojo

#写mapper的配置文件和一些增删改查语句

#“classpath是指类路径，也就是编译之后的target文件夹下的WEB-INF/class文件夹；而resources文件夹存放的是各种配置文件，当项目被编译时resources下的所有配置文件均被放在类路径下。”  
 mapper-locations: classpath:mybatis/mapper/\*.xml

*#配置模板路径，默认是templates，可以不用配置  
 #prefix: classpath:/templates*

创建一个简单数据库数据表

#数据库的名字  
create database if not exists mf;  
use mf;  
#改数据库表名  
create table `user`(  
 `id` int(16) NOT NULL AUTO\_INCREMENT primary key COMMENT '用户id',  
 `name` varchar(5) NOT NULL COMMENT '用户名字',  
 `password` varchar(16) NOT NULL COMMENT '密码',  
 `pid` varchar(16) NOT NULL COMMENT '手机号码'  
  
) ENGINE=InnoDB AUTO\_INCREMENT=105 DEFAULT CHARSET=utf8 ;

pojo 里的 User 实体类

@Data  
@AllArgsConstructor  
@NoArgsConstructor  
public class User1 {  
 private int id;  
 private String name;  
 private String password;  
 //手机密码  
 private String pid;

}

Controller层

//网页为/hello  
@GetMapping("/hello")  
public String test(Model model) {  
 //hello里面的name 指向到 go  
 model.addAttribute("name","go");  
 //将/hello页面内容指向templates下的hello.html  
 return "hello";  
}

@Autowired  
private UserMapper userMapper;  
@GetMapping("/zhenghe")  
public String test(Model model){  
 model.addAttribute("name",userMapper.selectById(1)

);  
 return "zhenghe";  
}  
@GetMapping("/addMsg")  
@ResponseBody  
public String insert(){  
 //Mapper里的insert接口  
 userMapper.insert(new User(4, "杨树", "baiyang", "17895010212"));  
  
 return "ok";  
}

集成Mybatis

在项目路径下建一个mapper

路径表

//接口的实现

@Controller

Controller

//接口

@Repository

Mapper

//实体类

Pojo pojo的实体类名和表名要一一对应

Resource里面建mybatis.mapper文件夹

里面建mapper.xml 用来操作增删改查的语句

创建完记得在菜单栏file的setting里的 language里的SQL 方言里 把数据库添加到这个文件夹

<?xml version="1.0" encoding="UTF-8" ?>  
  
<!DOCTYPE mapper  
  
 PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"  
 "http://mybatis.org/dtd/mybatis-3-mapper.dtd">  
<mapper namespace="com.thmy.mybat.mapper.UserMapper">  
 <!-- mapper里的接口名 返回结果类型 实体类类型-->  
 <select id="selectById" resultType="User">  
 select \* from mf.user where id = #{id}  
 </select>  
 <insert id="insert" parameterType="User">  
  
 insert into mf.user (id, name, password, pid) values (#{id}, #{name}, #{password}, #{pid})  
 </insert>  
</mapper>