1. springboot项目配置

<1>:引入springboot所需依赖

<parent>

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

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

<version>1.4.1.RELEASE</version>

</parent>

<dependency>

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

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

</dependency>

<dependency>

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

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

</dependency>

<dependency>

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

<artifactId>spring-boot-devtools</artifactId>

<optional>true</optional>

</dependency>

<2>:配置application.properties 文件

例如服务端口,数据源等

<3>:创建一个启动类

//标识一个springboot应用

@SpringBootApplication

//告诉Spring 哪个packages 的用注解标识的类 会被spring自动扫描并且装入bean容器。

@ComponentScan(basePackages={"com.byron"})

//指定mybatis接口映射类所在包

@MapperScan("com.byron.dao")

public class StartApp extends SpringBootServletInitializer{

/\*\*

\* 实现SpringBootServletInitializer让spring-boot项目在web容器中运行

\*/

protected SpringApplicationBuilder configure(SpringApplicationBuilder application) {

return application.sources(StartApp.class);

}

public static void main(String[] args){

SpringApplication.run(StartApp.class,args);

}

## 二.springboot整合mybatis

\*\*\*\*\*添加依赖

<dependency>

<groupId>org.mybatis.spring.boot</groupId>

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

<version>1.0.0</version>

</dependency>

方式一

<1>:

springboot启动类上加上如下注解:

//mybatis扫描接口映射包

@MapperScan("com.byron.dao")

<2>:如果\*.mapper文件和接口不在同一文件夹下，

则需要在springboot文件application.properties中加入如下配置:

##配置接口映射的xml文件(如果xml文件和接口在同一包下，则不需要该配置)

mybatis.mapper-locations=classpath:com/byron/dao/impl/\*.xml

同时可以使用如下配置:

##配置mybatis的主配置文件（一些基本配置）

mybatis.config=classpath:mybatis-config.xml

##开启驼峰命名法

mybatis.configuration.mapUnderscoreToCamelCase=true

方式二

<1>: