**分片连接池整合到springboot-修改User的登录功能**

# 1.分片连接池对象整合到springboot

## 1.1 目的

有了分片连接池,可以高效的使用多个redis节点的集群完成redis功能操作,我们可以利用框架技术,将连接池对象**创建在框架中**,哪里使用在哪里注入

## 1.2.操作代码步骤

### 1.2.1配置properties属性值(创建对象初始化过程时动态导入的)

**打开user项目的application.properties：**

spring.redis.shardedpool.nodes=192.168.243.133:6379,192.168.243.133:6380,192.168.243.133:6381

spring.redis.shardedpool.maxTotal=200

spring.redis.shardedpool.maxIdle=10

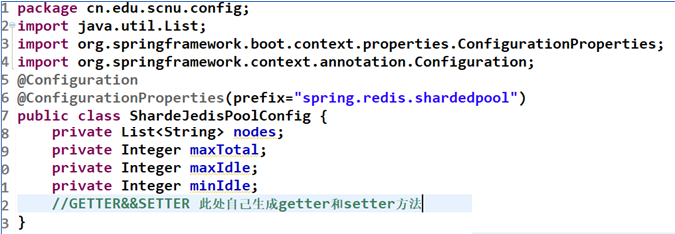
spring.redis.shardedpool.minIdle=2

图形用户界面, 应用程序

描述已自动生成

### 1.2编写配置类

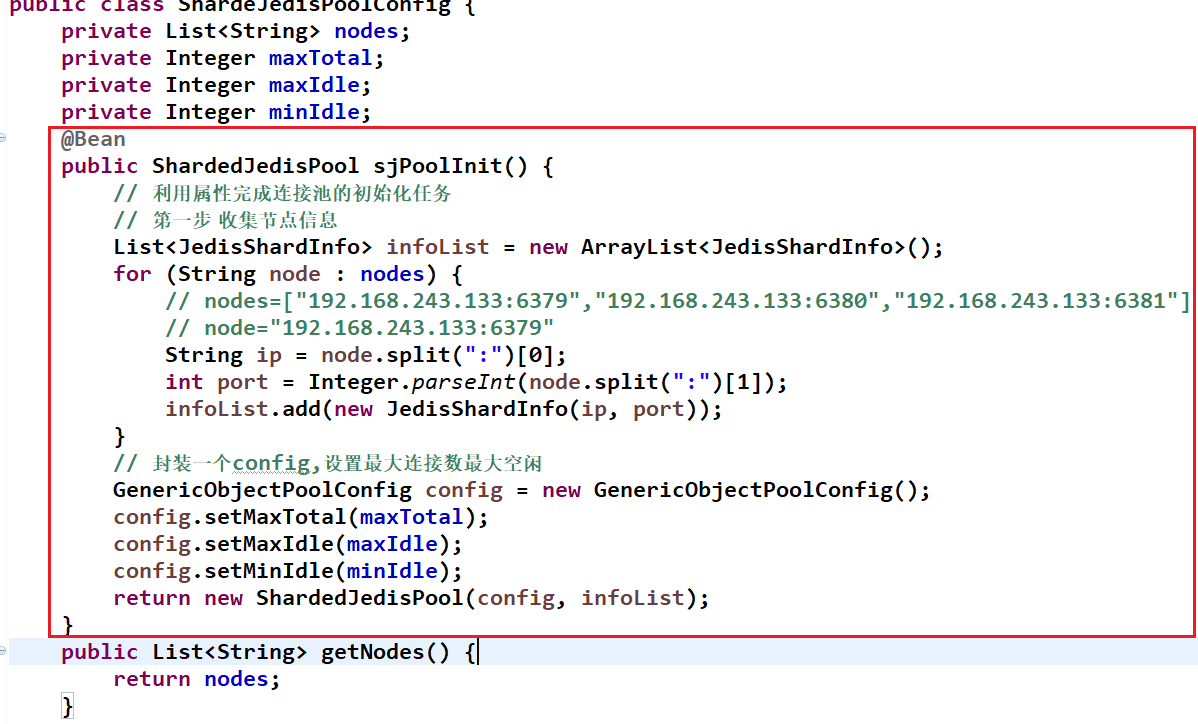
**编写配置类，利用ConfigurationProperties(prefix)前缀读取属性**



### 1.3编写一个连接池的初始化方法

@Bean维护返回对象

在上面SharedJedisPoolConfig配置类中编写一个连接池初始化方法sjPoolInit()，下图红色方框：



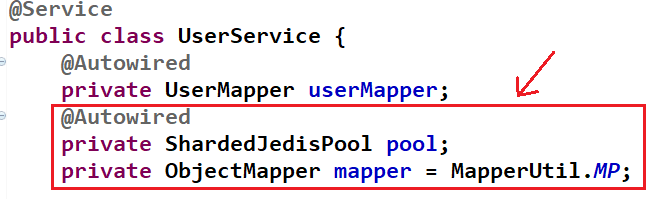
# 2.修改user的登录功能

## 2.1 修改代码

整合的springboot的分片连接池对象.修改usexr的登录逻辑

**UserController**(不发生任何代码变化的)

**在UserService中修改代码逻辑**





**//从redis中获取userJson**

文本

描述已自动生成

## 2.2 测试

启动Eureka、zuul、product、user,启动niginx