SpringBoot使用StringRedisTemplate对Redis数据库进行操作。

### 添加依赖

<dependency>

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

<artifactId>spring-boot-starter-data-redis</artifactId>

</dependency>

### RedisTemplate中定义了对5中数据结构操作

redisTemplate.opsForValue();//操作字符串

redisTemplate.opsForHash();//操作hash

redisTemplate.opsForList();//操作list

redisTemplate.opsForSet();//操作set

redisTemplate.opsForZSet();//操作有序set

### StringRedisTemplate与RedisTemplate

* StringRedisTemplate继承RedisTemplate。
* 两者数据是不共通的
* SDR默认采用的序列化策略有两种，一种是String的序列化策略，一种是JDK的序列化策略。
  + StringRedisTemplate默认采用的是String的序列化策略，保存的key，value都是采用此策略序列化保存的。
  + RedisTemplate默认采用JDK的序列化策略。

### RedisTemplate配置如下

@Bean

public RedisTemplate<String, Object> redisTemplate(RedisConnectionFactory redisConnectionFactory)

{

Jackson2JsonRedisSerializer<Object> jackson2JsonRedisSerializer = new Jackson2JsonRedisSerializer<Object>(Object.class);

ObjectMapper om = new ObjectMapper();

om.setVisibility(PropertyAccessor.ALL, JsonAutoDetect.Visibility.ANY);

om.enableDefaultTyping(ObjectMapper.DefaultTyping.NON\_FINAL);

jackson2JsonRedisSerializer.setObjectMapper(om);

RedisTemplate<String, Object> template = new RedisTemplate<String, Object>();

template.setConnectionFactory(redisConnectionFactory);

template.setKeySerializer(jackson2JsonRedisSerializer);

template.setValueSerializer(jackson2JsonRedisSerializer);

template.setHashKeySerializer(jackson2JsonRedisSerializer);

template.setHashValueSerializer(jackson2JsonRedisSerializer);

template.afterPropertiesSet();

return template;

}

### Redis的String数据结构（推荐使用StringRedisTemplate）

#### 操作字符串

stringRedisTemplate.opsForValue().set("test2", "1");

#### 操作集合

Long aLong = stringRedisTemplate.opsForList().leftPush("list", "a");

#### 操作hashMap

stringRedisTemplate.opsForHash().put("hash","hash-key","hash-value");

stringRedisTemplate

#### 操作set集合

Long add = stringRedisTemplate.opsForSet().add("set", "a", "b", "c");

System.out.println(add);

#### 操作ZSet

Boolean add = stringRedisTemplate.opsForZSet().add("zset", "a", 1);

### 五大常用数据类型使用场景

#### String

缓存：将数据以字符串方式存储

计数器功能：比如视频播放次数，点赞次数。

共享session：数据共享的功能，redis作为单独的应用软件用来存储一些共享数据供多个实例访问。

字符串的使用空间非常大，可以结合字符串提供的命令充分发挥自己的想象力

#### hash

字典。键值对集合，即编程语言中的Map类型。适合存储对象，并且可以像数据库中update一个属性一样只修改某一项属性值。适用于：存

储、读取、修改用户属性。也可以用Hash做表数据缓存

#### list

链表(双向链表)，增删快，提供了操作某一段元素的API。适用于：最新消息排行等功能；消息队列。

#### set

集合。哈希表实现，元素不重复，为集合提供了求交集、并集、差集等操作。适用于：共同好友；利用唯一性，统计访问网站的所有独立ip；> 好友推荐时，根据tag求交集，大于某个阈值就可以推荐。

#### sorted set

有序集合。将Set中的元素增加一个权重参数score，元素按score有序排列。数据插入集合时，已经进行天然排序。适用于：排行榜；带权重的消息队列。