



链滴

# Spring 集成 Redis

作者: [Veasion](#)

原文链接: <https://ld246.com/article/1505543031147>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

## 使用redisTemplate的一些坑。 <br>

### redisTemplate配置:

```
<bean id="redisPoolConfig" class="redis.clients.jedis.JedisPoolConfig">
  <property name="maxTotal" value="${redis.pool.max.active}" />
  <property name="maxIdle" value="${redis.pool.max.idle}" />
  <property name="maxWaitMillis" value="${redis.pool.max.wait}" />
  <property name="testOnBorrow" value="${redis.pool.testOnBorrow}" />
  <property name="testWhileIdle" value="${redis.pool.testWhileIdle}" />
  <property name="testOnReturn" value="${redis.pool.testOnReturn}" />
  <property name="timeBetweenEvictionRunsMillis" value="${redis.pool.timeBetweenEvictio
RunsMillis}" />
</bean>

<bean id="redisConnectionFactory" class="org.springframework.data.redis.connection.jedis.J
disConnectionFactory">
  <property name="hostName" value="${redis.host}" />
  <property name="port" value="${redis.port}" />
  <property name="password" value="${redis.password}" />
  <property name="usePool" value="true" />
  <property name="poolConfig" ref="redisPoolConfig" />
  <property name="timeout" value="${redis.timeout}" />
</bean>

<bean id="stringRedisSerializer" class="org.springframework.data.redis.serializer.StringRedis
erializer" />

<bean id="redisTemplate" class="org.springframework.data.redis.core.StringRedisTemplate"

  <qualifier value="redis"/>
  <property name="connectionFactory" ref="redisConnectionFactory" />
  <!-- 如果不配置Serializer, 那么存储的时候智能使用String, 如果用User类型存储, 那么会提示
  误User can't cast to String -->
  <property name="keySerializer">
    <bean class="org.springframework.data.redis.serializer.StringRedisSerializer" />
  </property>
  <property name="valueSerializer">
    <bean class="org.springframework.data.redis.serializer.JdkSerializationRedisSerializer" />
  </property>
</bean>
```

### Java代码:

```
@Resource
private RedisTemplate<String, Serializable> redisTemplate;

public void add(String key, Serializable obj, long timeout, TimeUnit timeUnit) {
  if(redisTemplate.hasKey(key)){
    delete(key);
  }
  redisTemplate.opsForValue().set(key, obj, timeout, timeUnit);
}
```

```
}  
  
public Serializable get(String key) {  
    return redisTemplate.opsForValue().get(key);  
}  
  
public void delete(String key) {  
    redisTemplate.delete(key);  
}
```

## 入坑点:

- 1.redisTemplate的注入, 不能用@Autowired, 要用@Resource, 因为配置中是id="redisTemplate"的bean, @Resource是优先按name注入的!!!
- 2.关于序列化问题, 坑爹的我在redis中手动存了个key, 然后忘了删除, 而程序调用该key就报错因为程序存入的是序列化后的值!!!