JSR107缓存规范

JAVA Caching提供了5个接口

CachingProvider缓存提供者

CachingManager缓存管理器

Cache缓存

Entry储存在Cache中的kv对

Expiry每一个Cache中都定义一个有效日期

Redis和mybatis、guava的缓存比的优势----数据与本地分离，当系统做集群时，本地缓存无法共享，而redis可以与所有集群交互。

set【key】【value】px【time】 nx

Spring缓存抽象

@Cacheable方法注解

在方法调用前先从缓存中查找，缓存未命中则调用方法返回，并将数据同步到缓存，缓存中有了不会更新缓存数据。

@Cacheable(

value = **"缓存的名字"** ,

key = **"缓存数据使用的key，可以用它来指定。默认方法接受参数的值，也支持el表达式#{【方法参数的名字】}"**,

keyGenerator = **"注册在spring容器中的类，用于生成key的生成器组件,与key参数二选一"**,

cacheManager = **"缓存管理器，用来指定从哪个缓存中拿到缓存"**,

cacheResolver = **"缓存解析器，和缓存解析器作用相同，与管理器二选一"**,

condition = **"指定符合条件的情况才缓存"**

unless = **"与condition用法想法你，返回值为true，就不缓存"**

)

@CacheEvict方法注解

方法执行后删除缓存

参数：

beforeInvocation=true提前执行删除缓存，在方法调用之前

@CachePut方法注解

更新缓存，缓存中有会被覆盖。

@Caching方法注解

多个缓存注解复合操作

@Caching(  
 cacheable = {@Cacheable()},  
 put = {@CachePut()},  
 evict = {@CacheEvict()}  
)

@CacheConfig类的注解

对类的缓存同一配置

开启基于注解的缓存

标注缓存注解