SpringMVC添加Ehcache缓存

Spring集成Ehcache十分简单,只需要添加如下配置即可:

Spring-config.xml

|  |
| --- |
| **<?**xml *version*="1.0" *encoding*="UTF-8"**?>**  <beans *xmlns*="http://www.springframework.org/schema/beans"  *xmlns:xsi*="http://www.w3.org/2001/XMLSchema-instance" *xmlns:context*="http://www.springframework.org/schema/context"  *xmlns:aop*="http://www.springframework.org/schema/aop" *xmlns:cache*="http://www.springframework.org/schema/cache"  *xmlns:p*="http://www.springframework.org/schema/p"  *xsi:schemaLocation*="http://www.springframework.org/schema/beans  http://www.springframework.org/schema/beans/spring-beans-4.3.xsd  http://www.springframework.org/schema/context  http://www.springframework.org/schema/context/spring-context-4.3.xsd  http://www.springframework.org/schema/aop  http://www.springframework.org/schema/aop/spring-aop-4.3.xsd  http://www.springframework.org/schema/cache  http://www.springframework.org/schema/cache/spring-cache-4.3.xsd"  *default-autowire*="byName">  *<!-- 使用properties文件中的内容填充spring配置文件中的占位符 -->*  <bean  *class*="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">  <property *name*="locations">  <list>  <value>classpath:config/\*.properties</value>  </list>  </property>  </bean>  <import *resource*="classpath:spring-mybatis.xml" />  <import *resource*="classpath:spring-shiro.xml" />  *<!-- 系统缓存框架 目前使用ehcache -->*  <import *resource*="classpath:spring-ehcache.xml" />  </beans> |

绿框部分，是将我们单独写的一个ehcache配置导入到当前xml中。

spring-ehcache.xml

|  |
| --- |
| **<?**xml *version*="1.0" *encoding*="UTF-8"**?>**  <beans *xmlns*="http://www.springframework.org/schema/beans"  *xmlns:xsi*="http://www.w3.org/2001/XMLSchema-instance"  *xmlns:cache*="http://www.springframework.org/schema/cache"  *xsi:schemaLocation*="http://www.springframework.org/schema/beans  http://www.springframework.org/schema/beans/spring-beans-4.3.xsd  http://www.springframework.org/schema/cache  http://www.springframework.org/schema/cache/spring-cache-4.3.xsd">  *<!-- Spring提供的基于的Ehcache实现的缓存管理器 -->*  *<!-- 目前Spring版本仅支持Ehcache2.5以上版本，但不支持Ehcache3 -->*    *<!-- 这个类作用：加载 Ehcache配置文件*  *如果有多个ehcacheManager要在bean加上p:shared="true" -->*  <bean *id*="ehcacheManager" *class*="org.springframework.cache.ehcache.EhCacheManagerFactoryBean">  <property *name*="configLocation" *value*="classpath:xml/ehcache.xml"/>  </bean>    *<!-- 这个类的作用：支持net.sf.ehcache.CacheManager -->*  <bean *id*="cacheManager" *class*="org.springframework.cache.ehcache.EhCacheCacheManager">  <property *name*="cacheManager" *ref*="ehcacheManager"/>  <property *name*="transactionAware" *value*="true"/>  </bean>    *<!-- cache注解，和spring-redis.xml中的只能使用一个 -->*  <cache:annotation-driven *cache-manager*="cacheManager" *proxy-target-class*="false"/>  </beans> |

Spring中默认就支持ehcache，直接声明bean后，就可以在service中使用ehcache的注解，来使用缓存了。具体使用参考代码AccountService.java。

关于ecache.xml是里面配置的是关于ehcache的一些参数，比如缓存清空的时间等。下面看它的配置。

Ehcache.xml

|  |
| --- |
| **<?**xml *version*="1.0" *encoding*="UTF-8"**?>**  <ehcache>  *<!-- 磁盘缓存位置 -->*  <diskStore *path*="java.io.tmpdir/ehcache" />  *<!-- 字典缓存 -->*  <cache *name*="dict-activeSessionCache" *maxElementsInMemory*="1000"  *eternal*="false" *timeToIdleSeconds*="1800" *timeToLiveSeconds*="1800"  *maxEntriesLocalDisk*="10000000" *overflowToDisk*="false"  *memoryStoreEvictionPolicy*="LRU" />  *<!-- 缓存半小时 -->*  <cache *name*="halfHour" *maxElementsInMemory*="10000"  *maxElementsOnDisk*="100000" *eternal*="false" *timeToIdleSeconds*="1800"  *timeToLiveSeconds*="1800" *overflowToDisk*="false" *diskPersistent*="false" />  *<!-- 默认缓存 -->*  <defaultCache  *maxEntriesLocalHeap*="10000"  *eternal*="false"  *timeToIdleSeconds*="120"  *timeToLiveSeconds*="120"  *maxEntriesLocalDisk*="10000000"  *diskExpiryThreadIntervalSeconds*="120"  *memoryStoreEvictionPolicy*="LRU">  <persistence *strategy*="localTempSwap" />  </defaultCache>  </ehcache> |

Xml中分别定义了几种不同的缓存。在代码中，我们通过注解的name属性来制定要用哪一个缓存。如果不指定的话，就会使用defaultCache这个缓存。

代码中使用的是这个缓存。详情参考代码AccountService.java。

Controller和service抓图：





实例运行：

Tomcat启动后，首先通过登录页<http://localhost:8088/qd/mvc/login>登录（以为我们有shiro，不登录不行）。

当登陆后，就可以执行<http://localhost:8088/qd/mvc/ehcache>，观察eclipse控制台，会发现，第一次访问时，控制台会打印出。这是因为第一次访问，会先拿着参数username的值，去缓存中找有没有对应的数据，因为没找到，所以就执行getAccountByName这个方法。

第二次在访问的时候，控制台没有输出任何东西。这是因为，首先参宿username的值到缓存中查询，发现有对应的数据。就不再执行getAccountByName方法了，而是直接把缓存中的数据返回了。