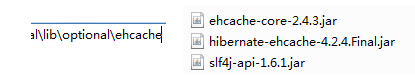
# 二级缓存

## 加jar



## 加配置文件

在src下增加一个配置文件：ehcache.xml。这个文件直接拷贝来用即可了，不用理会里面的内容。有须要的时候再研究也不迟

在\project\etc目录下拷贝过来

## <shared-cache-mode> 节点

<shared-cache-mode> 节点：若 JPA 实现支持二级缓存，该节点可以配置在当前的持久化单元中是否启用二级缓存，可配置如下值：

ALL：所有的实体类都被缓存

NONE：所有的实体类都不被缓存.

ENABLE\_SELECTIVE：标识 @Cacheable(true) 注解的实体类将被缓存

DISABLE\_SELECTIVE：缓存除标识 @Cacheable(false) 以外的所有实体类

UNSPECIFIED：默认值，JPA 产品默认值将被使用

# Demo测试

## persistence.xml

<!-- 配置二级缓存的策略 ALL：所有的实体类都被缓存 NONE：所有的实体类都不被缓存. ENABLE\_SELECTIVE：标识 @Cacheable(true)

注解的实体类将被缓存 DISABLE\_SELECTIVE：缓存除标识 @Cacheable(false) 以外的所有实体类 UNSPECIFIED：默认值，JPA

产品默认值将被使用 -->

<shared-cache-mode>ENABLE\_SELECTIVE</shared-cache-mode>

<!-- 二级缓存相关 -->

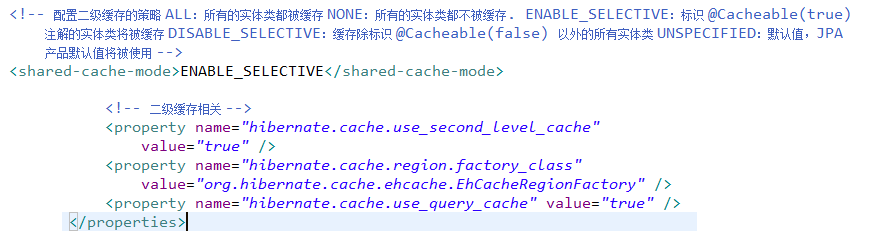
<property name=*"hibernate.cache.use\_second\_level\_cache"*

value=*"true"* />

<property name=*"hibernate.cache.region.factory\_class"*

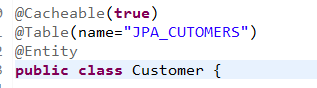
value=*"org.hibernate.cache.ehcache.EhCacheRegionFactory"* />

<property name=*"hibernate.cache.use\_query\_cache"* value=*"true"* />

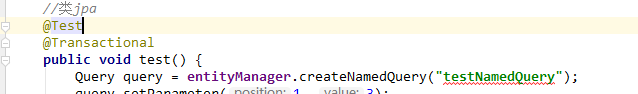


注意：这个配置要放在**provider** 节点和**class** 节点后面

## 注解



## 记得缓存事务注解



## 测试

@Test

**public** **void** testSecondLevelCache(){

Customer customer1 = entityManager.find(Customer.**class**, 1);

transaction.commit();

entityManager.close();

entityManager = entityManagerFactory.createEntityManager();

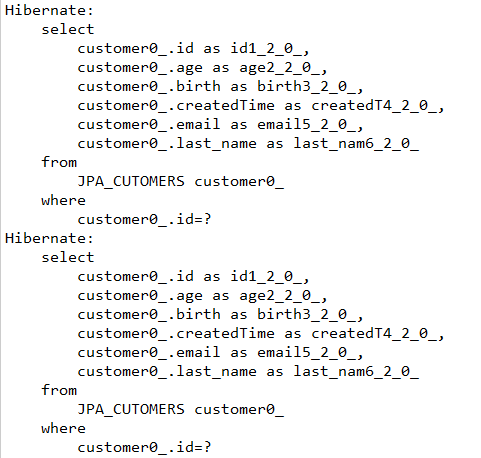
transaction = entityManager.getTransaction();

transaction.begin();

Customer customer2 = entityManager.find(Customer.**class**, 1);

}

没加缓存之前查询2次



加了后只有一次

