在此记录hibernate4源码阅读笔记，希望能够长期对Hibernate的一些错误的认识

hibernate二级缓存不同类型实体是分开存放的，类型如下

\* 1、单个实体

\* 2、集合

\* 3、有自然主键的实体

\* 4、查询缓存

\* 都是放在不同的地方，这个需要注意一下

\*

\* hibernate二级缓存四种访问策略的理解

\*

\* 1、只读，不需要有事务及锁的支持，应用只管插入

\* 2、读写，需要锁的支持，可以没有事务的支持，操作流程，获取缓存中的锁，然后开始操作数据库，数据库操作成功后，更新缓存。如果缓存是集群，集群更新采用推的策略

\* 3、不严格读写，不需要锁及事务的支持，数据库开始前清理缓，然后操作数据库，然后再次清理缓存，缓存为集群时，采用拉的方式之间删除，不做更新

\* 4、事务性，需要JTA环境的支持，缓存和数据库都要支持，在同一个事务中同时更新缓存及数据库，这个涉及到两阶段提交

Hibernate4主要新特性

\* 更换日志组件，主要是让日志支持i18n

\* 提供了Service Api ,这个是的hibernate具有更多的可扩展性，主要是想模块化，为osgi扫清障碍

\* 支持多租户

\* 其他一些小的增强或是改进

Hibernate查询接口详解

1. list(),一次性及时加载，如果开启二级查询缓存且设置setCacheble(true)时会以集合的形式保存查询结果，否则不予缓存
2. iterate() 一次性及时加载，在使用到相关实体时才把其从ResultSet转换为Java对象，无论二级缓存是否开启都不会使用二级缓存

load与get

1. load找不到对象报错，get返回null
2. load返回代理，使用的时候才查找二级缓存或是数据库去加载其他字段，get总是返回已经加载了所有数据的对象（有时候可能并不是实体类，而是代理，只是已经加载了数据）
3. load与get都会去查找及写二级缓存（前提该对象可以被缓存及Session的cacheMode可以对其操作，这个是可配置的，默认值读写均可以）