

一剑飘红

- [博客](#)
- [微博](#)
- [相册](#)
- [收藏](#)
- [留言](#)
- [关于我](#)

Hibernate中session的clear(),flush(),evict()方法详解

- 博客分类：
- [java/开源框架](#)

hibernate

一、Clear 方法

无论是Load 还是 Get 都会首先查找缓存（一级缓存）如果没有，才会去数据库查找，调用Clear() 方法，可以强制清除Session缓存。例：

```

1. publicvoid testClear(){
2. Session session =HibernateUtil.getSessionFactory().getCurrentSession();
3. session.beginTransaction();
4. Teacher t =(Teacher) session.get(Teacher.class,3);
5. System.out.println(t.getName());
6. session.clear();//这里不clear只会执行一次sql语句，有clear会执行2次
7. Teacher t2 =(Teacher) session.get(Teacher.class,3);
8. System.out.println(t2.getName());
9. session.getTransaction().commit();
10. }
```

二、Flush 方法

可以强制进行从内存到数据库的同步。例：

```

1. publicvoid testFlush(){
2. Session session =HibernateUtil.getSessionFactory().getCurrentSession();
3. session.beginTransaction();
```

```
4. Teacher t =(Teacher) session.get(Teacher.class,3);  
5. t.setName("yyy");  
6. session.flush();//有flush会执行2次UPDAE,没有会只执行一次  
7. t.setName("yyyyy");  
8. session.getTransaction().commit();  
9. }
```

Flush方法是可以设置的，也就是 flush 什么时候执行是可以设置的，在session.beginTransaction 前设置 FlushMode

```
1. session.setFlushMode(FlushMode.Always|AUTO|COMMIT|NEVER|MANUAL);
```

这里 FlushMode 有 5 个值可选

Always: 任何代码都会 Flush

AUTO: 默认方式 – 自动

Commit: COMMIT 时

Never: 始终不

MANUAL: 手动方式

设置 FlushMode 有个好处是可以节省开销，比如默认 session 只做查询时，就可以不让他与数据库同步了。

三、Evict 方法

会把指定的缓冲对象进行清除，而 Clear 方法则是把缓冲区内的全部对象清除，但不包括操作中的对象。

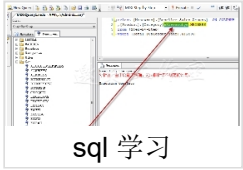
注意事项： Flush 方法和 Evict 方法一起使用出现的问题

Flush 方法和 Evict 方法一起使用出现的时候，一定要注意使用的顺序，否知会出现异常。下面的代码就是正确的使用顺序：

```
1. //因为user的主键生成策略采用的是uuid，所以调用完成save后，  
2. //只是将user纳入到了session的管理  
3. //不会发出insert语句，但是id已经生成，session中existsInDatebase状态为false  
4. session.save(user);  
5.  
6. //flush后hibernate会清理缓存，会将user对象保存到数据库中  
7. //（强制更新数据库，结果但不一定，因为commit时候可能会出现事物回滚）  
8. //并且设置session中existsInDatebase的状态为true  
9. session.flush();  
10.  
11. //将user对象从session中逐出，即session的EntityEntries属性中逐出，一般调用前必须保证该实体与  
12. //数据库相关属性一致,即调用之前执行flush()方法。  
13. session.evict(user);  
14.
```

- 15. //可以成功提交，因为hibernate在清理缓存时，
- 16. //在session的insertions集合中无法找到user对象
- 17. //所以就不会发出insert语句，也不会更新session中的existsInDatabase的状态
- 18. tx.commit();

注意：凡是在数据库集群环境中，最好每个对象都调用evict（）方法让其变成游离态，这样才会出现异常，



sql 学习



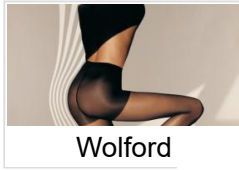
柴犬转让



精华液的正确使用



虚拟办公室



Wolford

分享到：

[jdk5-jdk8各版本的新特性](#) | [jdk1.7和jdk1.8区别](#)

- 2016-05-17 20:01
- 浏览 1459
- [评论\(0\)](#)
- 分类:[开源软件](#)
- [查看更多](#)

相关资源推荐

- [hibernate session中clear、evict、flush方法的区别](#)
- [Hibernate学习--- Session.evict\(\)方法](#)
- [session.clear\(\)与session.flush的区别](#)
- [Hibernate一级缓存管理-evict和clear的用法](#)
- [Hibernate Clear 与 Flush 方法](#)
- [Hibernate中session的clear\(\),flush\(\),evict\(\)方法详解](#)
- [Hibernate 里 session.flush\(\) session.clear\(\) session.close\(\) 区别](#)
- [Hibernate Session中flush、clear、evict方法的区别](#)
- [Hibernate clear\(\), flush\(\), evict\(\)区别](#)
- [session中的save\(\)与saveOrUpdate\(\),load\(\)与get\(\)区别](#)

- [Hibernate 中的session 的flush、reflush 和clear 方法，及数据库的隔离级别](#)
- [Hibernate中get和load方法的区别以及close\(\)、clear\(\)、evict\(\)](#)
- [hibernate中Session常用方法那些事 \(save\(\),flush\(\),delete\(\).....\)](#)
- [关于flush和evict](#)
- [Hibernate的clear\(\),flush\(\),evict\(\)方法详解](#)
- [Hibernate深入理解----03操作Session缓存方法 \(flush、refresh、clear，事务隔离级别 \)](#)
- [hibernate中flush\(\)、refresh\(\)、clear\(\)缓存操作](#)
- [Hibernate session flush commit](#)
- [org.hibernate.Session.evict\(Object object\)方法的使用](#)
- [图解 Hibernate，session.close\(\),session.clear\(\)区别](#)

评论

发表评论



[您还没有登录,请您登录后再发表评论](#)

