Hibernate 的 10 个常见面试问题及答案

2013/08/20 | 分类： [基础技术](http://www.importnew.com/cat/basic) | [0 条评论](http://www.importnew.com/6004.html#respond) | 标签： [HIBERNATE](http://www.importnew.com/tag/hibernate)

分享到：**23**

原文出处： [开源中国](http://www.oschina.net/translate/10-hibernate-interview-questions-answers-java-j2ee-senior)

在Java J2EE方面进行面试时，常被问起的Hibernate面试问题，大多都是针对基于Web的企业级应用开发者的角色的。Hibernate框架在Java界的成功和高度的可接受性使得它成为了Java技术栈中最受欢迎的对象关系影射（ORM）解决方案。Hibernate将你从数据库相关的编码中解脱了出来，使你可以更加专注地利用强大的[面向对象的设计原则](http://javarevisited.blogspot.com/2012/03/10-object-oriented-design-principles.html)来实现核心的业务逻辑。采用Hibernate后，你就能够相当容易地在不同的数据库间进行切换，而且你还可以利用Hibernate提供的开箱即用的二级缓存以及查询缓存功能。你也知道，大部分Java面试中所提的问题不仅仅会涉及Java的核心部分，而且还会涉及其它的Java框架，比如，根据项目的要求也有可能会问到[Spring 框架方面的问题](http://javarevisited.blogspot.com/2011/09/spring-interview-questions-answers-j2ee.html)或者[Struts方面的问题](http://javarevisited.blogspot.com/2011/11/struts-interview-questions-answer-j2ee.html)。如果你要参加的项目使用了Hibernate作为ORM解决方案，你就应该同时准备好回答Spring和Hibernate这两个框架方面的问题。好好看看JD或者职位说明，如果其中的任何地方出现了Hibernate这个词，就要准备好怎样来面对Hibernate方面的问题。

本文给出了一个Hibernate面试问题列表，这些都是我从朋友以及同事那里搜集来的。Hibernate 是一个非常流行的对象关系影射框架，熟捻Hibernate的优势所在以及Hibernate的Sesion API是搞定Hibernate面试之关键所在。

**Hibernate中get和load有什么不同之处?** 把get和load放到一起进行对比是Hibernate面试时最常问到的问题，这是因为只有正确理解get()和load()这二者后才有可能高效地使用Hibernate。get和load的最大区别是，如果在缓存中没有找到相应的对象，get将会直接访问数据库并返回一个完全初始化好的对象，而这个过程有可能会涉及到多个数据库调用；而load方法在缓存中没有发现对象的情况下，只会返回一个代理对象，只有在对象getId()之外的其它方法被调用时才会真正去访问数据库，这样就能在某些情况下大幅度提高性能。你也可以参考 [Hibernate中get和load的不同之处](http://javarevisited.blogspot.com/2012/07/hibernate-get-and-load-difference-interview-question.html)， 此链接给出了更多的不同之处并对该问题进行了更细致的讨论。

**Hibernate中save、persist和saveOrUpdate这三个方法的不同之处？**除了get和load，这又是另外一个经常出现的Hibernate面试问题。 所有这三个方法，也就是save()、saveOrUpdate()和persist()都是用于将对象保存到数据库中的方法，但其中有些细微的差别。例如，save()只能INSERT记录，但是saveOrUpdate()可以进行 [记录的INSERT和UPDATE](http://javarevisited.blogspot.com/2013/01/jdbc-batch-insert-and-update-example-java-prepared-statement.html)。还有，save()的返回值是一个Serializable对象，而persist()方法返回值为void。你还可以访问 [save、persist以及saveOrUpdate](http://javarevisited.blogspot.com/2012/09/difference-hibernate-save-vs-persist-and-saveOrUpdate.html)，找到它们所有的不同之处。

**Hibernate中的命名SQL查询指的是什么?** Hibernate的这个面试问题同Hibernate提供的查询功能相关。命名查询指的是用<sql-query>标签在影射文档中定义的SQL查询，可以通过使用Session.getNamedQuery()方法对它进行调用。命名查询使你可以使用你所指定的一个名字拿到某个特定的查询。 Hibernate中的命名查询可以使用注解来定义，也可以使用我前面提到的xml影射问句来定义。在Hibernate中，@NameQuery用来定义单个的命名查询，@NameQueries用来定义多个命名查询。

**Hibernate中的SessionFactory有什么作用? SessionFactory是线程安全的吗？** 这也是Hibernate框架的常见面试问题。顾名思义，SessionFactory就是一个用于创建Hibernate的Session对象的工厂。SessionFactory通常是在应用启动时创建好的，应用程序中的代码用它来获得Session对象。作为一个单个的数据存储，它也是 [线程安全的](http://javarevisited.blogspot.com/2012/12/how-to-create-thread-safe-singleton-in-java-example.html)，所以多个线程可同时使用同一个SessionFactory。Java JEE应用一般只有一个SessionFactory，服务于客户请求的各线程都通过这个工厂来获得Hibernate的Session实例，这也是为什么SessionFactory接口的实现必须是线程安全的原因。还有，SessionFactory的内部状态包含着同对象关系影射有关的所有元数据，它是 [不可变的](http://javarevisited.blogspot.com/2013/03/how-to-create-immutable-class-object-java-example-tutorial.html)，一旦创建好后就不能对其进行修改了。

**Hibernate中的Session指的是什么? 可否将单个的Session在多个线程间进行共享？** 前面的问题问完之后，通常就会接着再问这两个问题。问完SessionFactory的问题后就该轮到Session了。Session代表着Hibernate所做的一小部分工作，它负责维护者同数据库的链接而且 **不是线程安全的**，也就是说，Hibernage中的Session不能在多个线程间进行共享。虽然Session会以主动滞后的方式获得数据库连接，但是Session最好还是在用完之后立即将其关闭。

**hibernate中sorted collection和ordered collection有什么不同?** T这个是你会碰到的所有Hibernate面试问题中比较容易的问题。sorted collection是通过使用 [Java的Comparator](http://java67.blogspot.com/2012/10/how-to-sort-object-in-java-comparator-comparable-example.html)在内存中进行排序的，ordered collection中的排序用的是数据库的order by子句。对于比较大的数据集，为了避免在内存中对它们进行排序而出现 [Java中的OutOfMemoryError](http://javarevisited.blogspot.com/2011/09/javalangoutofmemoryerror-permgen-space.html)，最好使用ordered collection。

**Hibernate中transient、persistent、detached对象三者之间有什么区别？** 在Hibernate中，对象具有三种状态：transient、persistent和detached。同Hibernate的session有关联的对象是persistent对象。对这种对象进行的所有修改都会按照事先设定的刷新策略，反映到数据库之中，也即，可以在对象的任何一个属性发生改变时自动刷新，也可以通过调用Session.flush()方法显式地进行刷新。如果一个对象原来同Session有关联关系，但当下却没有关联关系了，这样的对象就是detached的对象。你可以通过调用任意一个session的update()或者saveOrUpdate()方法，重新将该detached对象同相应的seesion建立关联关系。Transient对象指的是新建的持久化类的实例，它还从未同Hibernate的任何Session有过关联关系。同样的，你可以调用persist()或者save()方法，将transient对象变成persistent对象。可要记住，这里所说的transient指的可不是 [Java中的transient关键字](http://javarevisited.blogspot.com/2012/03/difference-between-transient-and.html)，二者风马牛不相及。

**Hibernate中Session的lock()方法有什么作用?** 这是一个比较棘手的Hibernate面试问题，因为Session的lock()方法重建了关联关系却并没有同数据库进行同步和更新。因此，你在使用lock()方法时一定要多加小心。顺便说一下，在进行关联关系重建时，你可以随时使用Session的update()方法同数据库进行同步。有时这个问题也可以这么来问：*Session的lock()方法和update()方法之间有什么区别？*。这个小节中的关键点也可以拿来回答这个问题。

**Hibernate中二级缓存指的是什么？** 这是同Hibernate的缓存机制相关的第一个面试问题，不出意外后面还会有更多这方面的问题。二级缓存是在SessionFactory这个级别维护的缓存，它能够通过节省几番[数据库调用往返](http://javarevisited.blogspot.com/2012/01/improve-performance-java-database.html)来提高性能。还有一点值得注意，二级缓存是针对整个应用而不是某个特定的session的。

**Hibernate中的查询缓存指的是什么？** 这个问题有时是作为上个Hibernate面试问题的后继问题提出的。查询缓存实际上保存的是sql查询的结果，这样再进行相同的sql查询就可以之间从缓存中拿到结果了。为了改善性能，查询缓存可以同二级缓存一起来使用。Hibernate支持用多种不同的开源缓存方案，比如EhCache，来实现查询缓存。

**为什么在Hibernate的实体类中要提供一个无参数的构造器这一点非常重要？**

每个Hibernate实体类必须包含一个 [无参数的构造器](http://javarevisited.blogspot.com/2012/12/what-is-constructor-in-java-example-chainning-overloading.html), 这是因为Hibernate框架要使用Reflection API，通过调用Class.newInstance()来创建这些实体类的实例。如果在实体类中找不到无参数的构造器，这个方法就会抛出一个InstantiationException异常。

**可不可以将Hibernate的实体类定义为final类?**  
是的，你可以将Hibernate的实体类定义为final类，但这种做法并不好。因为Hibernate会使用代理模式在延迟关联的情况下提高性能，如果你把实体类定义成final类之后，因为 [Java不允许对final类进行扩展](http://javarevisited.blogspot.com/2011/12/final-variable-method-class-java.html)，所以Hibernate就无法再使用代理了，如此一来就限制了使用可以提升性能的手段。不过，如果你的持久化类实现了一个接口而且在该接口中声明了所有定义于实体类中的所有public的方法轮到话，你就能够避免出现前面所说的不利后果。

Java开发者的Hibernate面试问答列表就到此为止了。没人会对Hibernate作为ORM解决方案的受欢迎程度产生怀疑，如果你要申请的是Java J2EE方面的职位，你就等着人来问你Hibernate方面的面试问题吧。在JEE界，Spring和Hibernate是两个最流行的Java框架。要是你被问到了其它也值得分享的Hibernate方面的面试问题， 别忘了在Java社区中同大家分享一下。