1. 调用saveorupdate方法时，若传入的数据带有主键时则执行update操作，若没有主键则执行insert操作。解决方法：在jsp页面中把Id放入隐藏域中。
2. HQL和SQL的本质区别是HQL的操作对象是类，实例及其属性，而SQL的操作对象是数据表和字段等。
3. 在HQL语句中可以插入大多数的运算符和函数等，如：=，<,>,!=,between，not between，in，not in，is，like，is null等运算符
4. HQL有两种参数传递方式：名称占位符和位置占位符。
5. 在hibernate应用中，持久化类的访问方法有两个调用者：java应用程序和Hibernate。
6. Java应用程序不能方访问持久化类的private类型的getxxx()和setxxx（）方法，而Hibernate没有这种限制
7. Hibernate访问属性的策略有：property（默认值）表示通过相应的set和get方法，field方式是通过反射的方式访问可以绕过set和get方法
8. Session接口提供了基本的保存，更新，删除和加载java对象的方法。查询不属于session接口提供的方法。
9. 位于Session缓存中的对象称为称为持久化对象，它和数据库中的相关的记录对应，Session能够在某些时间点，按照缓存中对象的变化来执行相关的SQL语句，同步更新数据库这个过程称为清理缓存（flush）
10. 当Session的save（）方法持久化一个对象时，对象就被加入到Session缓存中，只要session的缓存还没有被清空，对象依然处在生命周期中。当session的get（）方法试图加载一个对象时，session会先判断缓存中是否存在这个对象，如果不存在就到数据库找。
11. Session的三大作用：
    1. 减少访问数据库的频率，提高数据访问的性能。
    2. 保证数据库中的相关记录与缓存中的相应的对象保持同步，session在清理缓存的时候会自动进行脏检查（dirty - check），如果发现不一致就会执行update语句。
    3. 当缓存中的持久化对象之间存在循环关联关系时，session会保证不出现访问对象图的死循环，以及由死循环引起的JVM堆栈溢出异常
12. Session缓存中对象的属性每次发生变化时，session并不会立刻清理缓存和执行相关的SQL语句，而是在特定的时间点清理缓存，这可以减少数据库的访问，提高性能。
13. 在默认情况下，session会在一下时间点清理缓存：
    1. 调用commit（）方法的时候，commit（）方法先清理缓存，然后再向数据库提交事务。Hibernate之所以把清理缓存的时间点安排在事务快结束时，一方面因为可以减少访问数据库的频率，还可以尽可能缩短当前事务对数据库中相关资源的锁定时间。
    2. 当应用程序执行一些查询操作时和显示的调用session的flush（）方法。
14. Session的commit（）和flush（）方法的区别：flush（）方法进行清理缓存的操作，执行一系列的SQL语句，但不会提交事务；commit（）方法会先调用flush（）方法，然后提交事务，提交事务意味着对数据库所做的更新被被永久保存下来。
15. 在持久化的角度，一个java对象有4个状态：
    1. 临时状态：刚刚用new语句创建，还没有被持久化，并且不处于session的缓存中。
    2. 持久化状态：已经被持久化，并且加入到session的缓存中。
    3. 游离状态：已经被持久化，但不处于session的缓存中。
    4. 删除状态：不再处于session的缓存中，并且session已经计划将其从数据库中删除。
16. Load()方法和get（）方法的区别：
    1. 当数据库不存在与OID对应的记录时，load（）方法会抛出异常，而get（）方法则返回null。
    2. 在默认情况下，load（）方法会采用延迟检索策略加载持久化对象（默认情况下lazy属性为true），而get（）方法（会忽略lazy属性）则采用立即检索。
    3. Load方法返回的是一个代理（只有一个OID属性），当要使用对象的属性时才会发出Sql语句。而get（）方法则是返回一个实例，执行该方法时就执行了SQL语句。
    4. Get（）只能使用一级缓存，而load（）可以使用一级和二级缓存。
17. 类级别的检索策略可以通过lazy属性进行设置，如果程序加载一个对象的目的是为了访问它的属性，可以采取立即检索。如果加载对象的目的仅仅是为了获得它的引用，可以采用延迟检索，此时要注意懒加载异常。
18. Batch-size属性:设定一次性初始化的数量，用来为延迟检索策略或立即检索策略设定批量的数量，批量检索能减少SELECT语句的数目，提高延迟检索或立即检索的运行性能
19. Set集合的fetch属性：确定初始化对象集合的方式
    1. 默认值为select，通过正常的方式来初始化set元素
    2. 可以取值为subselect， 通过子查询的方式来初始化所有的set集合，子查询作为where字句的in的条件出现，此时lazy属性失效。
    3. 若取值为join，则在加载一端的对象时，使用迫切左外连接的方式检索n的一端的集合属性，此时忽略lazy属性，HQL查询忽略fetch=join的取值。