hibernate和mybatis类似都是model层的框架，整合起来也很类似，都是将SessionFactory 交给spring来管理，他们的配置文件也很类似在大的配置文件中都是关于数据连接的配置，还有就有关POJO与数据库表的映射关系的配置文件的配置，就是相当于mybatis中的mapper只不过这个配置文件中内容比较少不用写sql语句，只写一些POJO对象与表中列的对应关系。以及java中的持久化类与数据库中哪个表的对应关系，这个就成就hibernate能够使用java代码对应到数据库操作，也就实现了hibernate传说中的跨数据库平台。也就是它底层会将sql语句根据数据库做对应的转换。还有啊就是，哎呀，刚刚看了下其他的东西，忘记想要说啥了。现在说到配置文件中类与表的对应关系了，这个配置文件中只对应关系就行了，不用写sql语句了，它的sql语句是底层封装了一些，然后比较精致、细致的配置文件还是要自己写的，就是底层只封装了，比较常用的sql语句。或者不是hibernate的问题，只是我不喜欢用这个，总感觉这个比较麻烦，特别是在sql语句的预编译时，和JDBC的原装方法很相似，就是很麻烦的那种一个一个的set进去，不如mybatis那么方便。对于对数据库的操作，最最特别的就是sql语句而sql语句能做到最好的就是mybatis的那种将参数使用动态取值的方式。虽然hibernate号称不用写sql语句但是，如果业务负责，不可避免的，而且他的sql也不够智能。总而言之，言而总之，就是我不喜欢这个。正所谓从入门到放弃。刚刚入门，或许还没有入门就已经放弃了。

hibernate是一种可以根据需求自动生成sql并提交的ORM框架，要比mybatis更加简洁一些，主要体现在不用再编写sql语句了。

hibernate分为两个配置文件hibernate.cfg.xml是主要的配置文件，在其中配置数据源，还有数据库的方言，最主要的配置mapping.hbm.xml文件的位置，其他的我也没看懂，不知道如果删除会出现什么后果，对只要有他们两个没有问题，起码查询是没有问题的。对了就数据库方言都注掉都没有问题。

mapping.hbm.xml中配置hibernate对应的实体类的包名，还有就是具体的实体类对应的表，还有就是具体哪个列对应哪个列名，哪一个是主键。目前就是这里配置这些。

hibernate的HQL语句还是要编写sql语句，而且有占位符的还是需要一个一个的set进去，这就和JDBC很相像了，这样编写起来就会代码很长，而且不便于维护，因为它是在java代码中，如果需要改变时就会很麻烦，总而言之就是维护性很差，对了，也不是一无是处，据听说好像比原生sql好的是具有数据库的可移植性。