<http://blog.csdn.net/xxssyyyyssxx/article/details/50125077>

Hibernate初识

1.开源的持久层框架.

2.ORM(Object/Relational Mapping)映射工具，建立面向对象的域模型和关系数据模型之间的映射.

3.连接java应用和数据库的中间件.

4.对JDBC进行封装，负责java对象的持久化.

5.在分层结构中处于持久化层，封装对数据库的访问细节，

  使业务逻辑层更专注于实现业务逻辑

JDBC优点

       直接底层操作，提供了很简单、便捷的访问数据库的方法，跨平台性比较强。灵活性比较强，可以写很复杂的SQL语句。支持连接池。

JDBC缺点

       因为JAVA是面向对象的，JDBC没有做到使数据能够面向对象的编程，使程序员的思考仍停留在SQL语句上。操作比较繁琐，

       很多代码需要重复写很多次，在trycatch之间徘徊。 如果遇到批量操作，频繁与数据库进行交互，容易造成效率的下降。

Hibernate优点

       Hibernate利用面向对象的技术操作数据库,是一个面向对象操作数据库的orm框架，是基于JDBC开发的。

       Hibernate实现了面向对象的数据库编程

       Hibernate比起JDBC来，在代码的书写上比较简单化了。

       Hibernate提出了缓存机制，这样可以使访问数据的效率提高很大。

       Hibernate切换数据库非常简单

      1、Hibernate对JDBC访问数据库的代码做了封装，大大简化

         了数据访问层繁琐的重复性代码。   
      2、Hibernate是一个基于jdbc的主流持久化框架，是一个优秀

          的orm实现，它很大程度的简化了dao层编码工作。   
      3、Hibernate使用java的反射机制，而不是字节码增强程序类实现

          透明性   
      4、Hibernate的性能非常好，因为它是一个轻量级框架。映射的灵

          活性很出色。它支持很多关系型数据库，从一对一到多对多的各

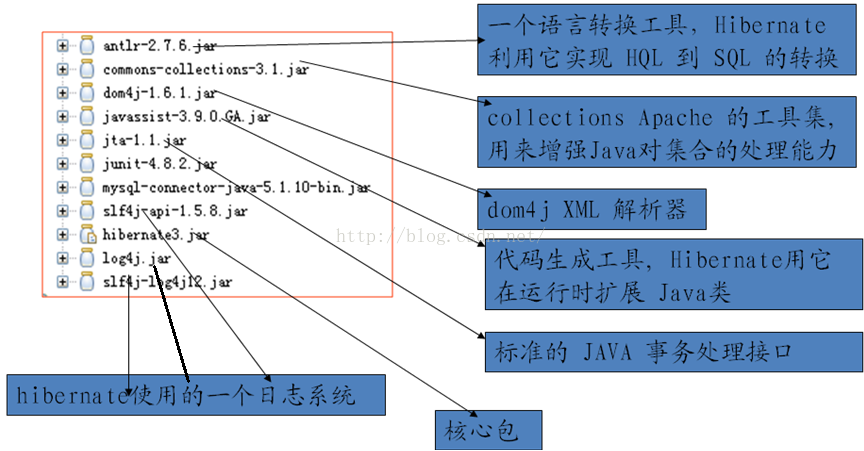
          种复杂关系。

     5、Hibernate使数据库的操作完全面向对象。而不是从前的面向关系进行操作。

Hibernate缺点

      效率不是很高，在大数据量的情况下不适合用，程序员无法控制sql语句的生成。所以在sql优化要求比较高的项目中也不能使用，可以用ibatis或者mybatis

Hibernate需要的jar



hibernate基本系统中还需要一个配置文件hibernate.cfg.xml及与持久化类关联的映射文件\*\*.hbm.xml

hibernate.cfg.xml中的一些属性

Hibernate.connection.url  表示要链接的数据库地址

Hibernate.connection.driver\_class    表示要链接的数据库的驱动类

Hibernate.connection.username     要连接的数据库的用户名

Hibernate.connection.password      要连接的数据库的密码

Hibernate.dialect   表示要使用的数据库的类型

         org.hibernate.dialect.MySQL5Dialect      mysql数据库

         org.hibernate.dialect.Oracle9Dialect        oracle数据库

         org.hibernate.dialect.SQLServerDialect   SQLServer数据库

hibernate.hbm2ddl.auto

         validate:加载hibernate时验证创建表结构

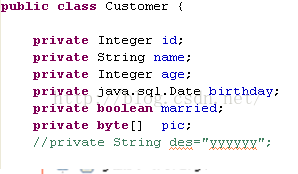
         update:加载hibernate时自动更新数据库结构，如果表存在不用创建，如果不存在就创建。

         create:每一次加载hibernate时都创建表结构

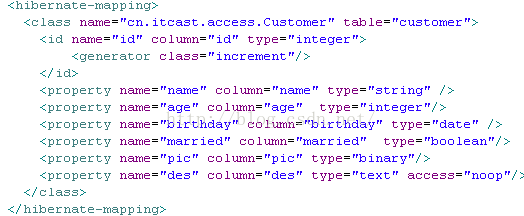
         create-drop:加载hibernate时创建，退出时删除

一个基本的例子

持久化类：



映射文件：



客户端：



l  Cnfiguration 类负责管理 Hibernate 的配置信息。包括如下内容：

•   **Hibernate运行的底层信息：数据库的URL、用户名、密码、JDBC驱动类，数据库Dialect,数据库连接池等（对应hibernate.cfg.xml 文件）。**

**•    持久化类与数据表的映射关系（\*.hbm.xml 文件）**

l  创建 Configuration 的两种方式

•    属性文件（hibernate.**properties**）:

**Configurationcfg = new Configuration();**

•    Xml文件（hibernate.cfg.**xml**）

**Configurationcfg = new Configuration().configure();**

第二种创建方式较为常见。Hibernate.cfg.xml文件默认的目录为系统的根目录。

l  **Configuration对象根据当前的配置信息生成 SessionFactory 对象。SessionFactory 对象一旦构造完毕，即被赋予特定的配置信息(SessionFactory 对象中保存了当前的数据库配置信息和所有映射关系以及预定义的SQL语句。同时，SessionFactory还负责维护Hibernate的二级缓存)。**

**Configurationcfg = new Configuration().configure();**

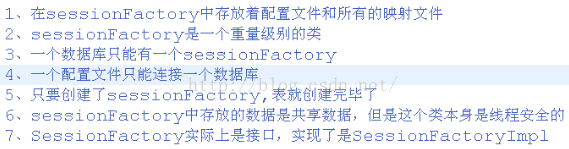
**SessionFactory sf = cfg.buildSessionFactory();**

l  **是线程安全的。**

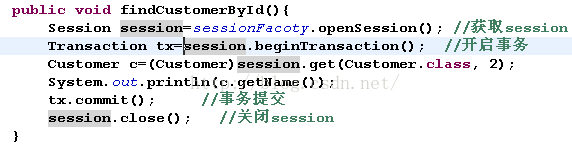
**l  SessionFactory是生成Session的工厂：**

**Session session = sf.openSession();**

l  构造 SessionFactory 很消耗资源，一般情况下一个应用中只初始化一个 SessionFactory 对象。这个对象中包含有所有的配置信息和映射信息，所以是一个重量级类。一个sessionFactory只对应一个数据库。



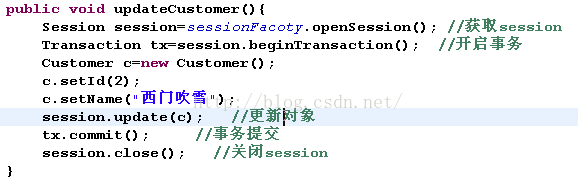
查询例子：



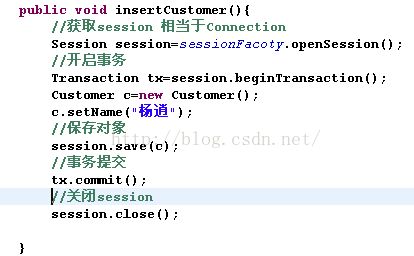
删除例子：



更新例子：



保存例子：



Hibernate的运行周期

l  Hibernate的整个运行过程如下：

1、应用程序先调用Configuration类,该类读取Hibernate配置文件及映射文件中的信息，

2、并用这些信息生成一个SessionFactory对象，

3、然后从SessionFactory对象生成一个Session对象，

4、并用Session对象生成Transaction对象；

    A、可通过Session对象的get(),load(),save(),update(),delete()和saveOrUpdate()等方法对PO进行加载、保存、更新、删除、等操作；

    B、在查询的情况下，可通过Session对象生成一个Query对象，然后利用Query对象执行查询操作；如果没有异常，Transaction对象将提交这些操作到数据库中。

