一.配置Table的POJO Bean

二.配置Model的ORM(檔名為Model.hbm.xml

1.宣告為Hibernate的xml

<!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN""http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">

2.ORM

<hibernate-mapping>

<class name=*"com.hibtest1.entity.Users"* table=*"users"* >

<id name=*"id"* column=*"Id"*>

PK自動生成方式為自動增加，class=”assign”為手動增加

<generator class=*"native"*></generator>

</id>

<property name=*"loginName"* column=*"LoginName"* />

<property name=*"loginPwd"* column =*"LoginPwd"*/>

</class>

</hibernate-mapping>

三.設置Hibernate.cfg.xml

1.宣告為Hibernate\_config檔

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

2.配置DB connect參數(*show\_sql/ connection.url/ connection.username/ connection.password/ connection.driver\_class/ dialect*)，和使用的Model配置

<hibernate-configuration>

<session-factory>

<property name=*"show\_sql"*>true</property>

<property name=*"connection.url"*>jdbc:sqlserver://ip;databaseName=bookshop;</property>

<property name=*"connection.username"*>RS</property>

<property name=*"connection.password"*>readsave</property>

<property name=*"connection.driver\_class"*>com.microsoft.sqlserver.jdbc.SQLServerDriver</property>

<property name=*"dialect"*>org.hibernate.dialect.SQLServerDialect</property>

<!—-設置會使用到的POJO Bean設定檔-->

<mapping resource=*"com/hibtest1/entity/Users.hbm.xml"* />

</session-factory>

</hibernate-configuration>

四.配置主方法

//1.初始化,讀取組態檔hibernate.cfg.xml

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

//2.讀取組態檔映射檔案(Model.hbm.xml)，建立sessionFactory

SessionFactory sessionFactory=config.buildSessionFactory();

//3.開啟session

Session session=sessionFactory.openSession();

//4-1查資料

Model model=session.get

//4-2ADD資料

Transaction tx=**null**;

**try** {

tx=session.beginTransaction();

session.save(user);

tx.commit();

} **catch** (Exception e) {

**if**(tx!=**null**){

tx.rollback();

}

}**finally**{

session.close();

}

//4-3刪資料

//載入要移除的資料

Users user=(Users)session.get(Users.**class**, **new** Integer(1));

**try** {

tx=session.beginTransaction();

session.delete(user);

tx.commit();

} **catch** (Exception e) {

**if**(tx!=**null**){

tx.rollback();

}

e.printStackTrace();

}**finally**{

session.close();

}

4-4改資料

//載入要修改的資料

Users user=(Users)session.get(Users.**class**, **new** Integer(2));

//修改資料

user.setLoginName("popopo");

**try** {

tx=session.beginTransaction(); //開始一個交易

session.update(user); //執行更新

tx.commit(); //傳送交易

} **catch** (Exception e) {

**if**(tx!=**null**){

tx.rollback(); //交易返回

}

e.printStackTrace();

}**finally**{

session.close(); //關閉session

}

五.Jar

hibernate3.jar、SQL的JAR