

一:hibernate 环境搭建

- 1.添加hibernate依赖(hibernate 4.3.11) 数据库驱动
- 2.创建hibernate.cfg.xml主配置文件(在src/main/resource)
- 3.创建实体类 (持久化类) 和映射文件 (hbm.xml) (在src/main/java)
- 4.在hibernate.cfg.xml中导入映射文件 (hbm.xml)
- 5.使用hibernate api 进行数据库操作

二:ORM 含义

ORM：对象关系映射，是将关系数据库中表的数据映射成为对象，是对象与数据库中的列的关系，在底层对JDBC进行了一次封装。

O：Object

R：Relative

M：Mapping

三：save() saveOrUpdate() meger() 区别

- 1.save()方法能够保存实体到数据库。
 - 2.saveOrUpdate()方法会执行插入或者更新操作。如果该对象在数据库中已经存在则更新，不存在则插入。
 - 3.merge()方法会先select 再update，调用merge()方法是把对象复制一份，原来的对象状态仍为游离状态。
- 判断一个对象在数据库中是否存在，是判断该id在数据库中是不是存在。如果id为空，就会直接执行插入操作；
如果该id存在，执行update操作。

四：hibernate 每个接口的含义(六大核心接口)

Configuration：Configuration接口负责配置并启动Hibernate。

SessionFactory：SessionFactroy接口负责初始化Hibernate。

Session：Session接口负责执行持久化对象的CRUD操作。

Transaction：Transaction接口负责事务相关的操作。

Query和Criteria：Query和Criteria接口负责执行各种数据库查询。

五：get() 和 load() 的区别

- 1.get()方法，及时加载。及时查询操作；只要调用get方法立刻向数据库查询。
 - 2.load()方法，默认懒加载，即在使用数据的时候，才向数据库发送查询的sql语句。
- session关闭以后，不可以使用懒加载。

3.当数据库中查询不到东西时，get()方法会返回一个null,load()方法则会抛出异常ObjectNotFoundException。

六：list() 和 iterator()区别

1.list查询：

一次性把所有的记录都查询出来了；会放入缓存，不会从缓存中取数据。

2.iterate (N+1次查询)：

N表示所有的记录总数，即会发送一条语句查询所有的记录的主键，这是第一条查询语句，再根据每一个主键去数据库查询，这是根据第一次查询的条数进行N次查询操作；会放入缓存，也会从缓存中取出数据。

七：hql 编写规范

hql 是Hibernate查询语言，是针对持久化对象用来获取对象，

hql 语句中关键字不区分大小写，但是实体类和对象属性要区分大小写，语句 from 实体类 类名。

1.获取Hibernate Session对象。

2.编写HQL语句。

3.以HQL语句作为参数，调用Session的createQuery方法创建查询对象。

4.如果HQL语句包含参数，则调用Query的setXxx方法为参数赋值。

5.调用Query对象的list()或uniqueResult()方法返回查询结果列表（持久化实体集）。

八：HQL中设置参数占位符有哪几种方式

1.按参数位置绑定：使用"?"来占位，在使用query时，必须将"?"及其索引位置和字段名对应上。

2.按参数名称绑定：使用命名参数，冒号+字段名（或别名）。

九：Hibernate中如何实现分页查询

使用了两个方法：

1.setFirstResult()：设置从第几条开始。

2.setMaxResults()：设置每页显示的条数。

开始条数：(pageNo-1)*pageSize

十：Query 常见的方法

1.uniqueResult()：获取的唯一结果集。

2.list()：一次性查出所有的信息。

3.iterate()：先查出所有的id，当调用next(),根据id 查询这条行记录，执行 (n+1) 条sql 语句。

十一：HQL投影查询的方式有哪些

三种方式:

1.直接查(object[])：直接使用select 属性 from 类名的方式查询。

2.查询返回对象(构造方法)：使用select new 类名(属性名...) from 类名的方式,返回实体类。

注意:如果使用实体类的方式,那么必须在实体类中有这样的构造方法。

3.查询返回Map键值对：使用select new Map(属性名...) from 类名的方式,返回Map集合,通过下标获取属性值。

注意:如果使用实体类的方式,那么必须在实体类中有这样的构造方法。

十二：单向多对一关联和双向一对多如何配置

1.配置单向多对一：

在实体中，在多的一方的实体类中添加一的一方的对象；

在hbm.xml(映射文件)中，在多的一方的(hbm.xml) 添加many-to-one。

2.配置双向一对多：

在实体中，在一的一方的实体类中添加一个set多的一方的集合对象；

在hbm.xml(映射文件)中，在一的一方的(hbm.xml) 添加set标签，配置cascade和inverse，在set标签中添加one-to-many。

十三：如何配置级联操作

Casade用来说明当对主对象进行某种操作时是否对其关联的从对象也作类似的操作，在映射文件中添加<set>标签，在<set>标签中配置cascade属性。

常用的cascade: none,all,save-update,delete, merge。

注：一般对many-to-one,many-to-many不设置级联，在<one-to-one>和<one-to-many>中设置级联。

十四：inverse属性的作用

inverse属性是否维护关联关系，指定了关联关系中的方向，

inverse设置为false，则为主动方，由主动方负责维护关联关系，默认是false，

inverse设置为true，不负责维护关联关系。

十五：延迟加载含义

延迟加载lazy，默认为true，不调用不加载，调用的时候，才加载。

lazy的好处：提高程序的效率。

十六：left join 和right join， inner join区别

left join(左联接)：返回包括左表中的所有记录和右表中联结字段相等的记录 (左表为主表，右表为从表)，

right join(右联接)：返回包括右表中的所有记录和左表中联结字段相等的记录 (右表为主表，左表为从表)，

inner join(内连接)：只返回两个表中联结字段相等的行。

十七：hibernate 常见的注解都有哪些

1.@Entity：将一个类声明为一个持久化类

2.@Table：为持久化类映射指定表

3.@Id：声明了持久化类的标识属性

4.@GeneratedValue：定义标识属性值的生成策略

5.@SequenceGenerator：定义序列生产器

6.@Column：将属性映射到列（字段）

7.@Transient：将忽略这些属性

8.@OneToOne：建立持久化类之间的一对一关联关系

9.@OneToMany：建立持久化类之间的一对多关联关系

10.@ManyToOne：建立持久化类之间的多对一关联关系

11.@JoinColumn：和@ManyToOne配合，指定外键列

12.@ManyToMany：建立持久化类之间的多对多关联关系

十八.hibernate工作原理：

1.通过Configuration config = new Configuration().configure();//读取并解析

hibernate.cfg.xml配置文件

2.由hibernate.cfg.xml中的<mapping resource="com/xx/User.hbm.xml"/> 读取并解析映射信息

3.通过SessionFactory sf = config.buildSessionFactory();//创建SessionFactory

4.Session session = sf.openSession();//打开Session

5.Transaction tx = session.beginTransaction();//创建并启动事务Transaction

6.persistent operate操作数据，持久化操作

7.tx.commit();//提交事务

8.关闭Session

9.关闭SessionFactory

十九.Hibernate的优/缺点：

优点：

- 1.更加对象化：以对象化的思维操作数据库，我们只需要操作对象就可以了，开发更加对象化。
- 2.移植性：因为Hibernate做了持久层的封装，你就不知道数据库，你写的所有的代码都具有可复用性。
- 3.Hibernate是一个没有侵入性的框架，没有侵入性的框架我们称为轻量级框架。
4. 对比Struts的Action和ActionForm，都需要继承，离不开Struts。Hibernate不需要继承任何类，不需要实现任何接口。这样的对象叫POJO对象。
- 5.Hibernate代码测试方便。
- 6.提高效率，提高生产力。

缺点：

- 1.使用数据库特性的语句，将很难调优
- 2.对大批量数据更新存在问题
- 3.系统中存在大量的攻击查询功能