# Day36 Hibernate

## Hibernate简介

### 1.Hibernate概念

Hibernate是一个面向对象的在Dao层来使用的,基于ORM对象关系模型(映射)的这么一个开源框架.Hibernate是对JDBC的一个轻量级的封装框架.

Hibernate是一个基于ORM思想的,对JDBC进行封装的Dao持久层,面向Java的开源的对象关系模型框架.

### 2.轻量级和重量级框架的区别

轻量级框架的特点:

对我们自己的代码侵入少,对容器的依赖少,配置简单,使用方便,启动时间相对也较短,对内存资源的消耗也比较少.

经典的轻量级框架:

Hibernate,mybatis(ibatis),Toplink;

重量级框架代表:

EJB:配置和使用都较复杂.

### 3.Hibernate的作用

①.建立起Java类与数据库中的表之间的映射关系;

②.实现对数据库的增删改查,CRUD;

### 4.Hibernate优点

①.不需要再写大量的SQL语句,缩短了开发时间,提高了开发效率;

②.能够使得开发人员的角色更专业,专注于业务层代码的实现;

③.分层更清晰,耦合性更小;

④.可以很轻松的从一个数据库平台移植到另一个数据库平台;

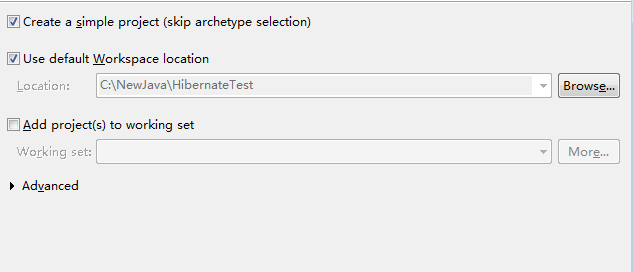
⑤.Hibernate可以使用较优秀的SQL语句,保证项目的性能;

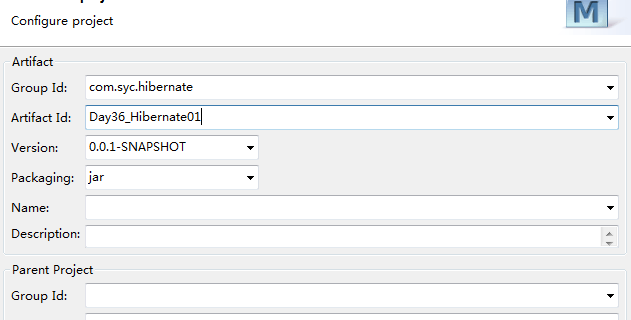
⑥.Hibernate相对于Struts2框架来说,不仅局限于JavaEE项目,还可以用于JavaSE项目.

## Hibernate开发

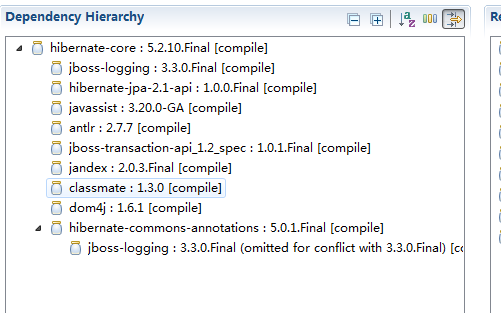
### 1.开发步骤

#### .创建JavaSE(EE)项目





#### .添加依赖包



hibernate-core:hibernate的核心功能包

jboss-logging.jar:jboss服务器所提供的日志包;

antlr.jar:用来创建SQL语句;

jboss-transaction-api:提供数据库的事务API;

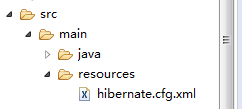
dom4j:负责xml解析的;

mysql的驱动包:最好是5.1.0以下的版本.

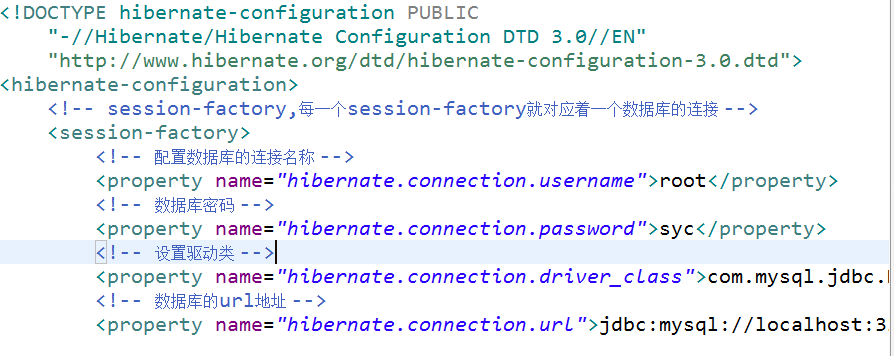


#### .进行代码的实现:

##### 第一步:创建hibernate.cfg.xml配置文件:



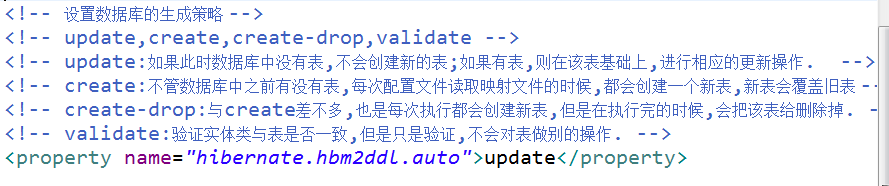
配置文件的具体内容:





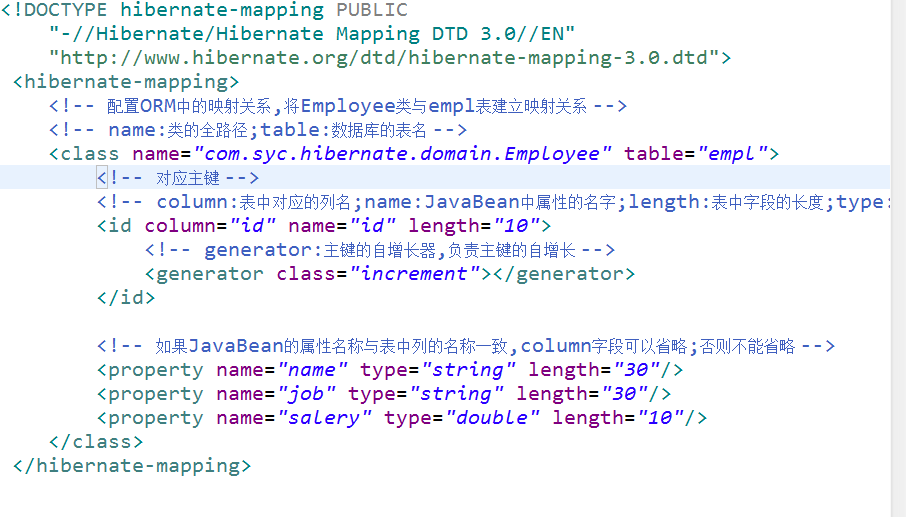
注意:不要忘了添加映射文件.

hbm2ddl.auto:表的创建策略

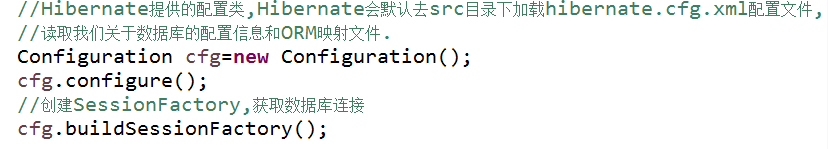


##### 第二步:创建ORM映射文件

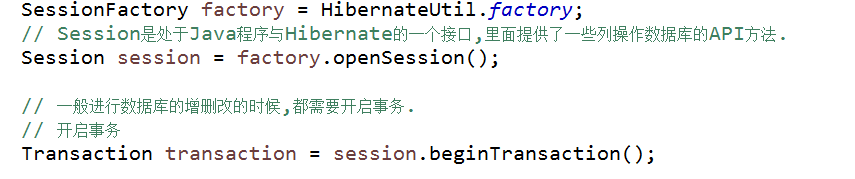
映射文件的命名,应该是类名.hbm.xml

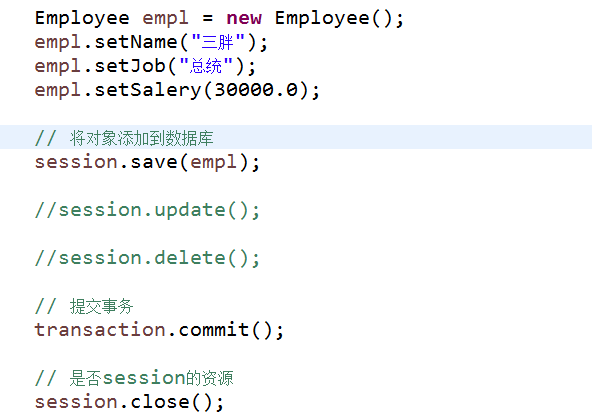


##### 第三步:调用Hibernate的API方法,实现CRUD操作.

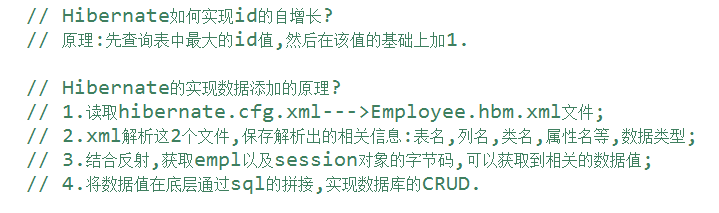


##### 数据库的添加实现:

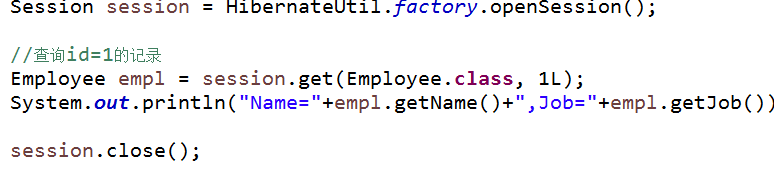




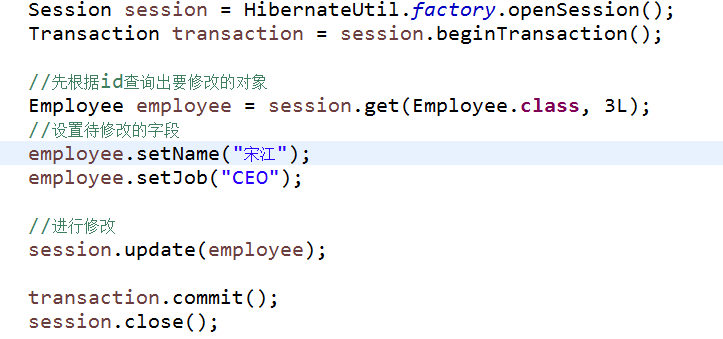
##### CRUD的实现原理(重点)



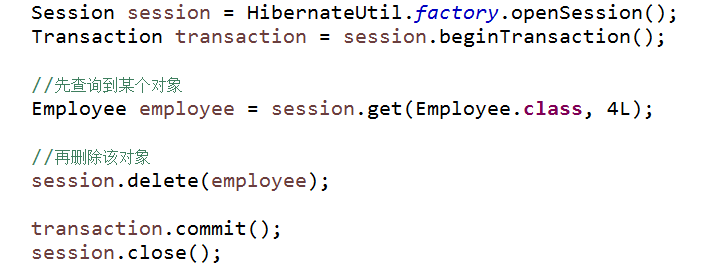
##### 数据库的查询实现:



##### 数据库的修改实现:

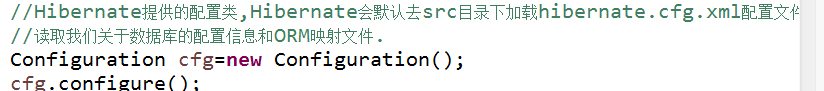


##### 数据库的删除实现:

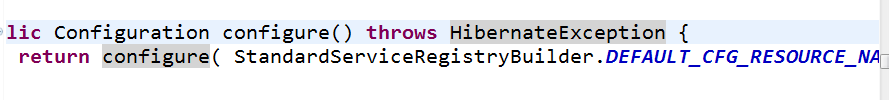


### 2.Hibernate源码分析

#### ①.源码分析入口:



#### ②.进入到configure()方法内部:



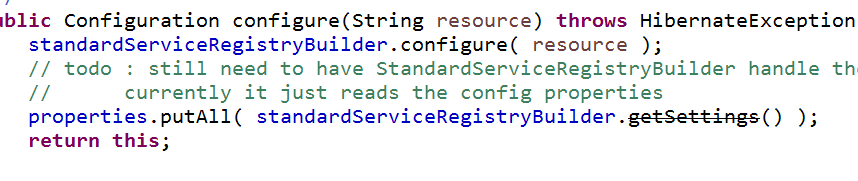
会发现



就是一个字符串常量,该常量的值是:hibernate.cfg.xml;

所以我们就知道我们的配置文件默认的名字必须是hibernate.cfg.xml.

#### .进入到configure()方法的内部:





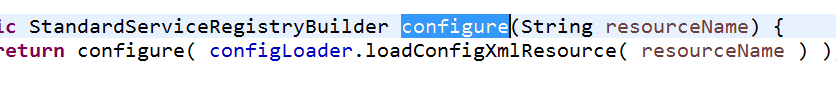
该方法读取resource资源所代表的xml配置文件,进行xml解析,解析出来的信息存放到....Builder类中.



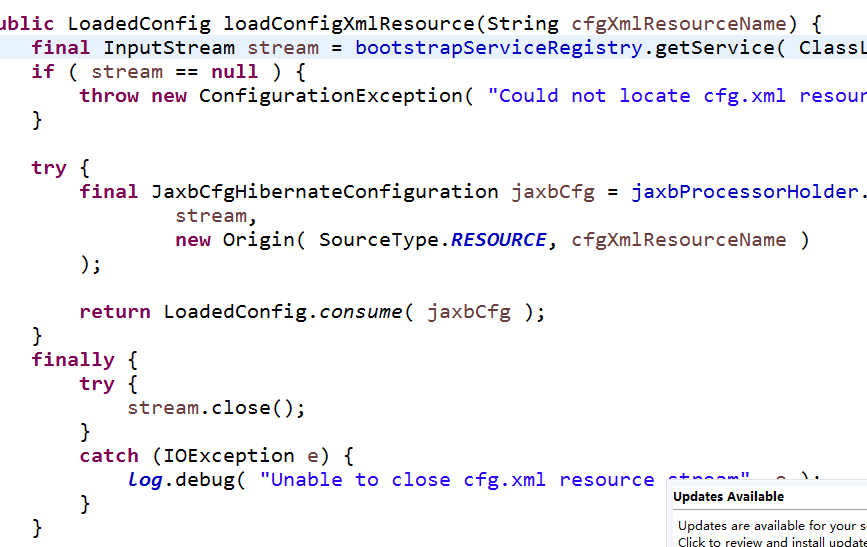
将hibernate.cfg.xml中读取的配置信息存到Properties配置文件中.

#### **. 分析**进入到StardardServiceRegistryBuilder类中,分析该方法.





#### .进行xml解析,解析出来的数据再进行封装

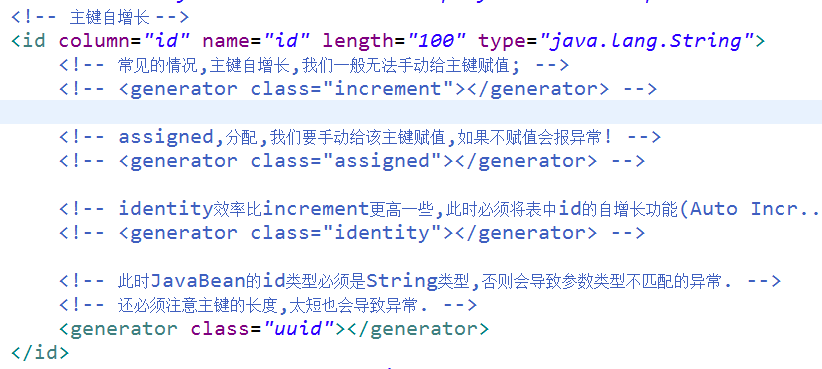


1. .解析完xml之后,将xml中的所有配置信息和映射信息,进过层层封装,最终变为StandardServiceRegistryBuilder类中的一个Map-----settings.

#### .然后将该settings集合存放到Properties中,Hibernate源码再去加载读取该Properties类,通过一些列的getString(key)读取配置信息,最终得到类名,属性名,表名,列名等必要信息;

#### .在具体的代码中,再结合反射获取JavaBean中的数据值,把这些数据值,通过Hibernate底层封装好的SQL语句,实现数据库的一些操作.

### 3.Hibernate的id主键生成策略



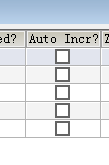
#### 如果id的策略为assigned,赋值重复时,会导致如下异常:



#### 当id的策略为identity的时候,如果主键的AutoIncr自增长功能没有打开,会报如下异常:



解决方法:打开该功能.



#### 如果id策略为uuid,但是id的类型不是string,会报异常:

