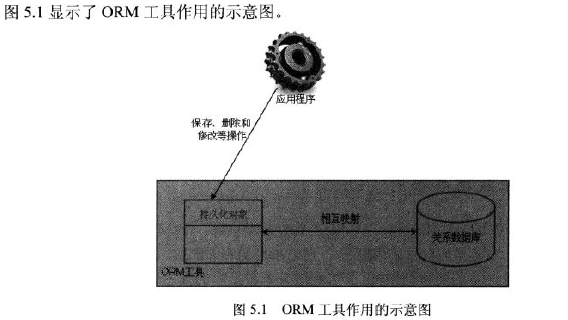
Hibernate学习笔记

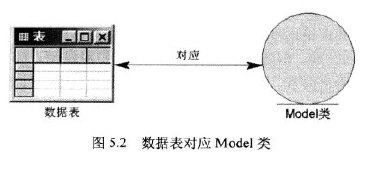
# ORM和Hibernate

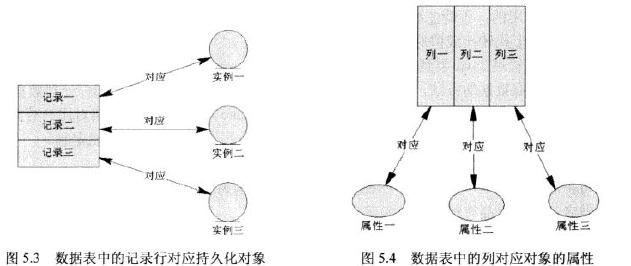
## 对象/关系数据库映射（ORM）



ORM工具的唯一作用就是：把对持久化对象的保存、删除、修改等操作，转换成对数据库的操作。

## 基本映射方式





# Hibernate入门

## Hibernate的数据库操作

### PO

HibernateQs\src\org\crazyit\app\domain\News.java

**package** org.crazyit.app.domain;

/\*\*

\* Description:

\* <br/>网站: <a href="http://www.crazyit.org">疯狂Java联盟</a>

\* <br/>Copyright (C), 2001-2012, Yeeku.H.Lee

\* <br/>This program is protected by copyright laws.

\* <br/>Program Name:

\* <br/>Date:

\* **@author** Yeeku.H.Lee kongyeeku@163.com

\* **@version** 1.0

\*/

**public** **class** News

{

//消息类的标识属性

**private** Integer id;

//消息标题

**private** String title;

//消息内容

**private** String content;

//id属性的setter和getter方法

**public** **void** setId(Integer id)

{

**this**.id = id;

}

**public** Integer getId()

{

**return** **this**.id;

}

//title属性的setter和getter方法

**public** **void** setTitle(String title)

{

**this**.title = title;

}

**public** String getTitle()

{

**return** **this**.title;

}

//content属性的setter和getter方法

**public** **void** setContent(String content)

{

**this**.content = content;

}

**public** String getContent()

{

**return** **this**.content;

}

}

HibernateQs\src\org\crazyit\app\domain\News.hbm.xml

<?xml version=*"1.0"* encoding=*"gb2312"*?>

<!-- 指定Hiberante3映射文件的DTD信息 -->

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<!-- hibernate-mapping是映射文件的根元素 -->

<hibernate-mapping package=*"org.crazyit.app.domain"*>

<!-- 每个class元素对应一个持久化对象 -->

<class name=*"News"* table=*"news\_table"*>

<!-- id元素定义持久化类的标识属性 -->

<id name=*"id"*>

<!-- 指定主键生成策略（此为Oracle的主键生成策略） -->

<generator class=*"sequence"*/>

</id>

<!-- property元素定义常规属性 -->

<property name=*"title"*/>

<property name=*"content"*/>

</class>

</hibernate-mapping>

**PO=POJO+映射文件**

### Hibernate配置文件

HibernateQs\src\hibernate.cfg.xml

<?xml version=*"1.0"* encoding=*"GBK"*?>

<!-- 指定Hibernate配置文件的DTD信息 -->

<!DOCTYPE hibernate-configuration PUBLIC

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

"http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<!-- hibernate- configuration是连接配置文件的根元素 -->

<hibernate-configuration>

<session-factory>

<!-- 指定连接数据库所用的驱动 -->

<property name=*"connection.driver\_class"*>oracle.jdbc.driver.OracleDriver</property>

<!-- 指定连接数据库的url，hibernate连接的数据库名 -->

<property name=*"connection.url"*>jdbc:oracle:thin:@localhost:1521:LEON</property>

<!-- 指定连接数据库的用户名 -->

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

<!-- 指定连接数据库的密码 -->

<property name=*"connection.password"*>Leonhard8</property>

<!-- 指定连接池里最大连接数 -->

<property name=*"hibernate.c3p0.max\_size"*>20</property>

<!-- 指定连接池里最小连接数 -->

<property name=*"hibernate.c3p0.min\_size"*>1</property>

<!-- 指定连接池里连接的超时时长 -->

<property name=*"hibernate.c3p0.timeout"*>5000</property>

<!-- 指定连接池里最大缓存多少个Statement对象 -->

<property name=*"hibernate.c3p0.max\_statements"*>100</property>

<property name=*"hibernate.c3p0.idle\_test\_period"*>3000</property>

<property name=*"hibernate.c3p0.acquire\_increment"*>2</property>

<property name=*"hibernate.c3p0.validate"*>true</property>

<!-- 指定数据库方言 -->

<property name=*"dialect"*>org.hibernate.dialect.Oracle10gDialect</property>

<!-- 根据需要自动创建数据表 -->

<property name=*"hbm2ddl.auto"*>update</property>

<!-- 显示Hibernate持久化操作所生成的SQL -->

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

<!-- 将SQL脚本进行格式化后再输出 -->

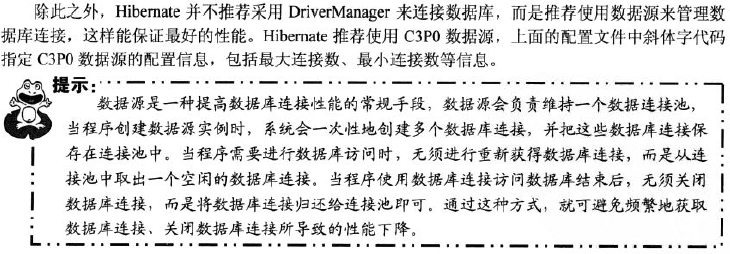
<property name=*"hibernate.format\_sql"*>true</property>

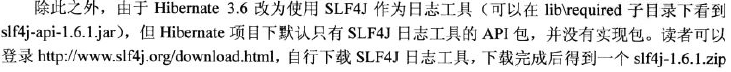
<!-- 罗列所有的映射文件 -->

<mapping resource=*"org/crazyit/app/domain/News.hbm.xml"*/>

</session-factory>

</hibernate-configuration>







### Hibernate数据库操作主类

HibernateQs\src\lee\NewsManager.java

**package** lee;

**import** org.hibernate.\*;

**import** org.hibernate.cfg.\*;

**import** org.crazyit.app.domain.\*;

/\*\*

\* Description:

\* <br/>网站: <a href="http://www.crazyit.org">疯狂Java联盟</a>

\* <br/>Copyright (C), 2001-2012, Yeeku.H.Lee

\* <br/>This program is protected by copyright laws.

\* <br/>Program Name:

\* <br/>Date:

\* **@author** Yeeku.H.Lee kongyeeku@163.com

\* **@version** 1.0

\*/

**public** **class** NewsManager

{

**public** **static** **void** main(String[] args)

**throws** Exception

{

//实例化Configuration，

Configuration conf = **new** Configuration()

//下面方法默认加载hibernate.cfg.xml文件

.configure();

//以Configuration创建SessionFactory

SessionFactory sf = conf.buildSessionFactory();

//创建Session

Session sess = sf.openSession();

//开始事务

Transaction tx = sess.beginTransaction();

//创建消息实例

News n = **new** News();

//设置消息标题和消息内容

n.setTitle("疯狂Java联盟成立了");

n.setContent("疯狂Java联盟成立了，"

+ "网站地址http://www.crazyit.org");

//保存消息

sess.save(n);

//提交事务

tx.commit();

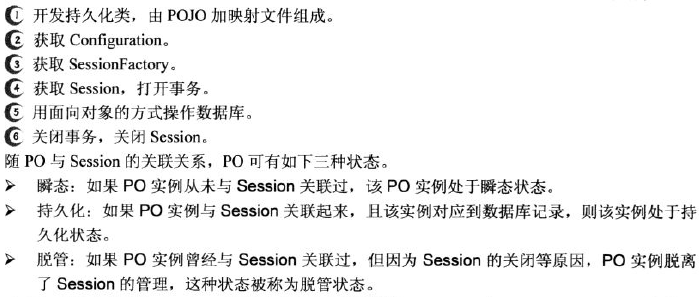
//关闭Session

sess.close();

sf.close();

}

}

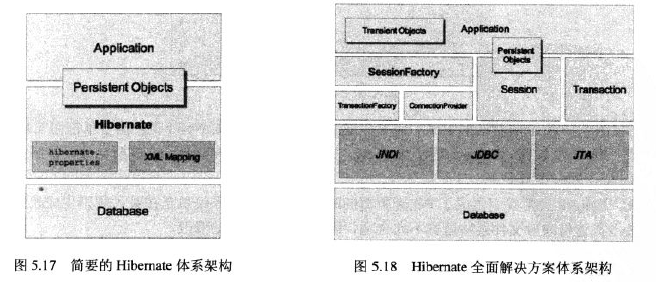


通常一个应用对应一个SessionFactory对象。

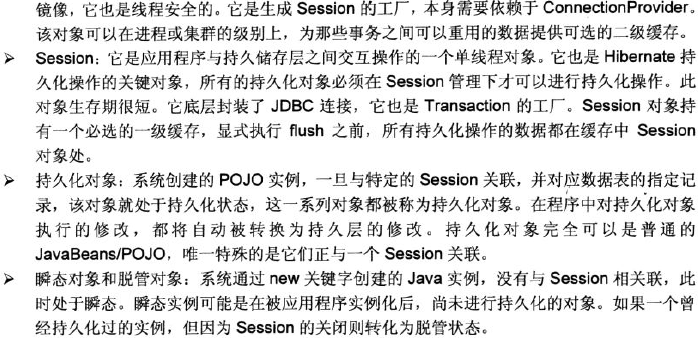


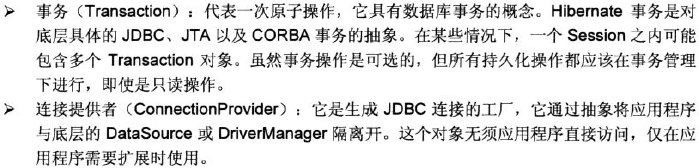


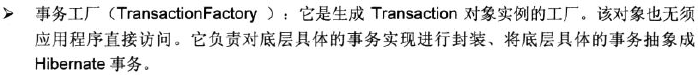
# Hibernate体系结构











# 深入Hibernate的配置文件

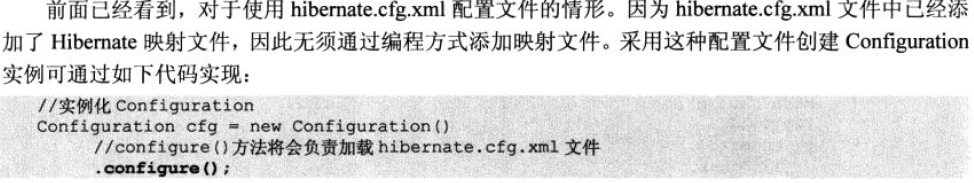
## 创建Configuration对象



### 使用hibernate.properties作为配置文件



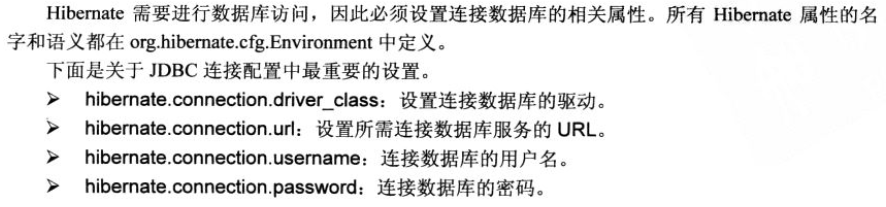
### 使用hibernate.cfg.xml作为配置文件

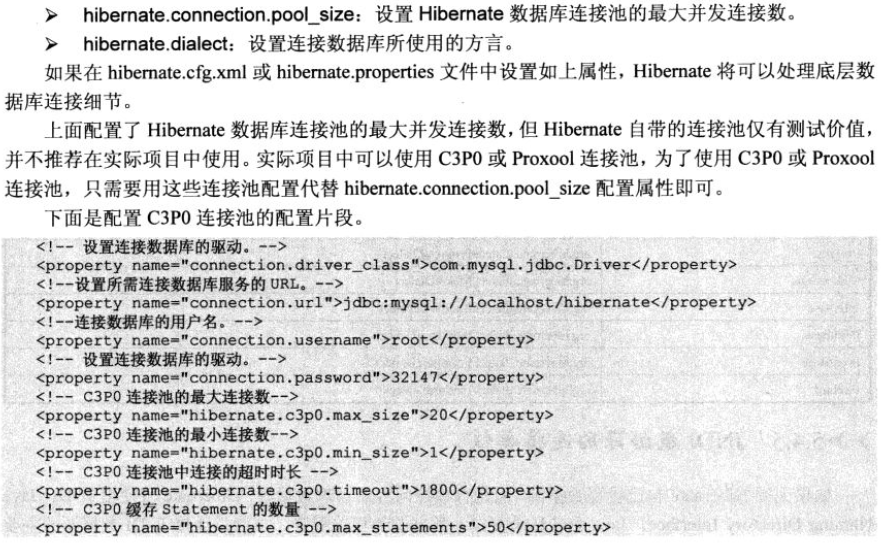


### 不使用配置文件创建Configuration实例

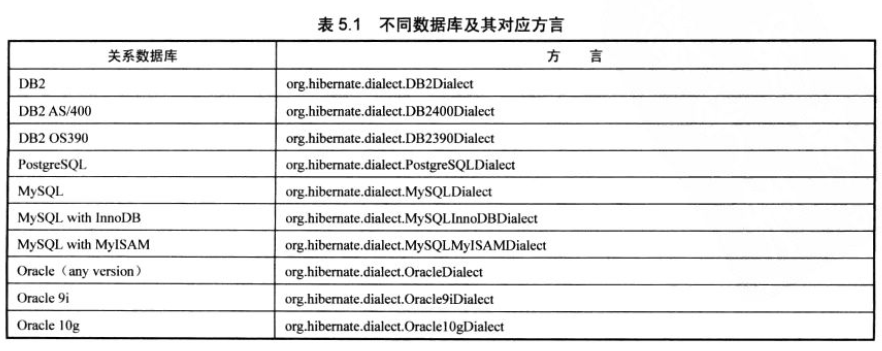
## Configuration属性

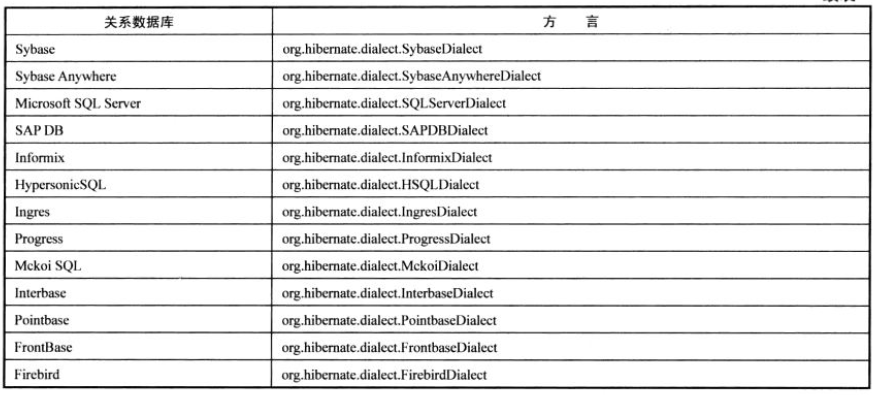
### JDBC连接属性



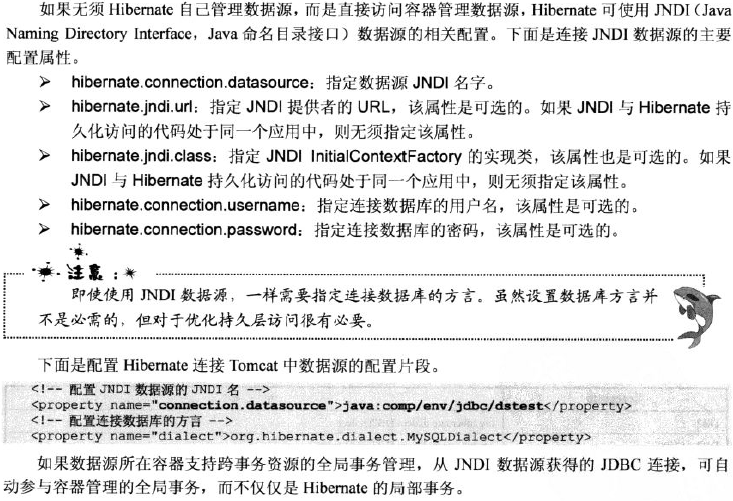


### 数据库方言

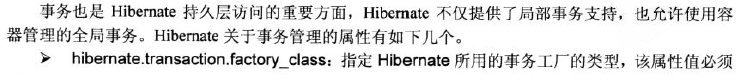


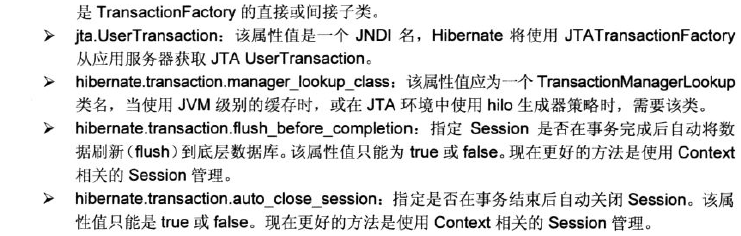


### JNDI数据源的连接属性

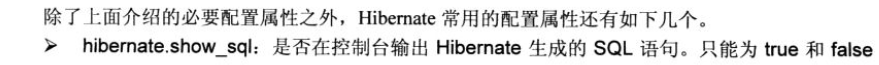


### Hibernate事务属性





### 其他常用的配置属性



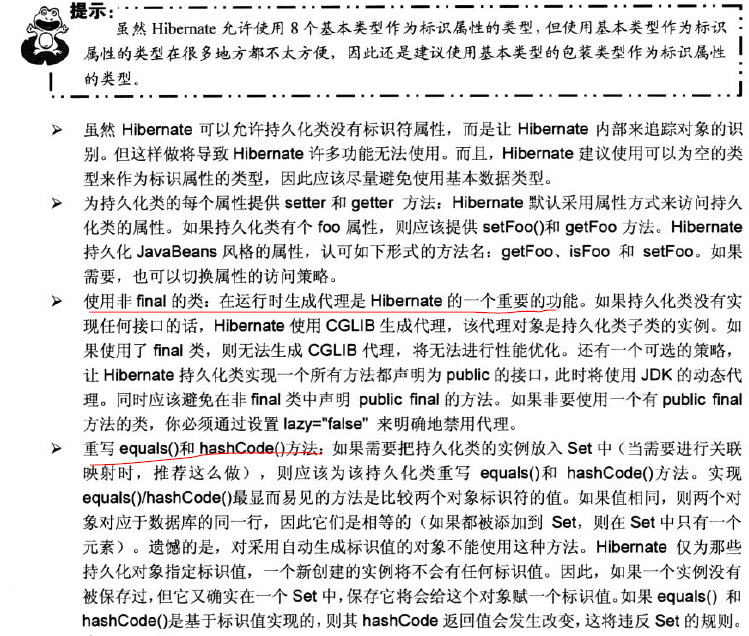


# 深入理解持久化对象

### 持久化类的要求

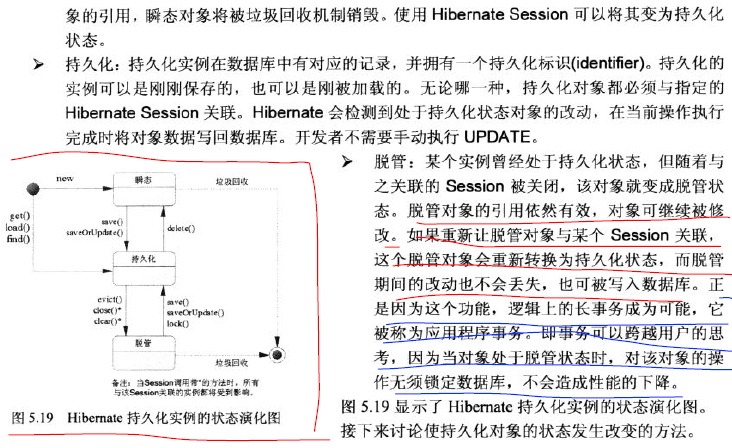
持久化类基本都是普通、传统的Java对象（POJO），但我们还是应该遵守如下规则：





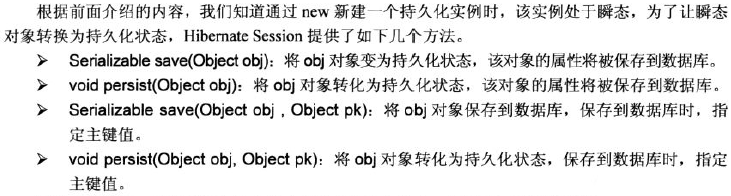
### 持久化类的状态

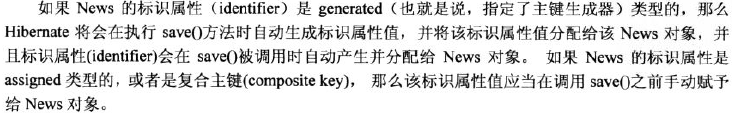


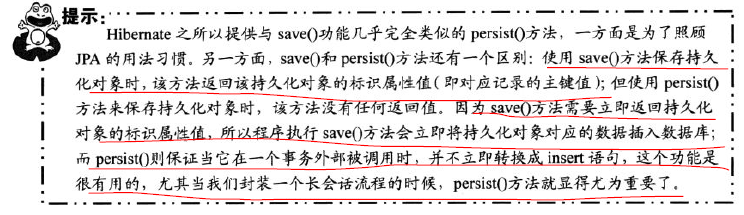


### 改变持久化状态的方法

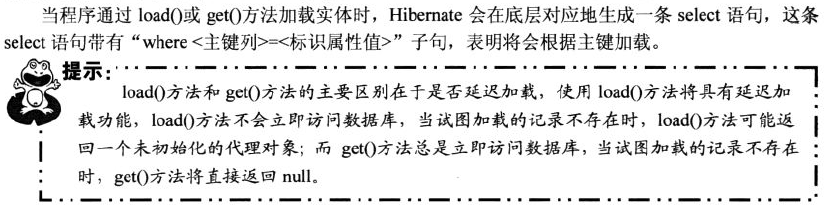
#### save与persist







#### load与get





#### update、updateOrSave与merge

