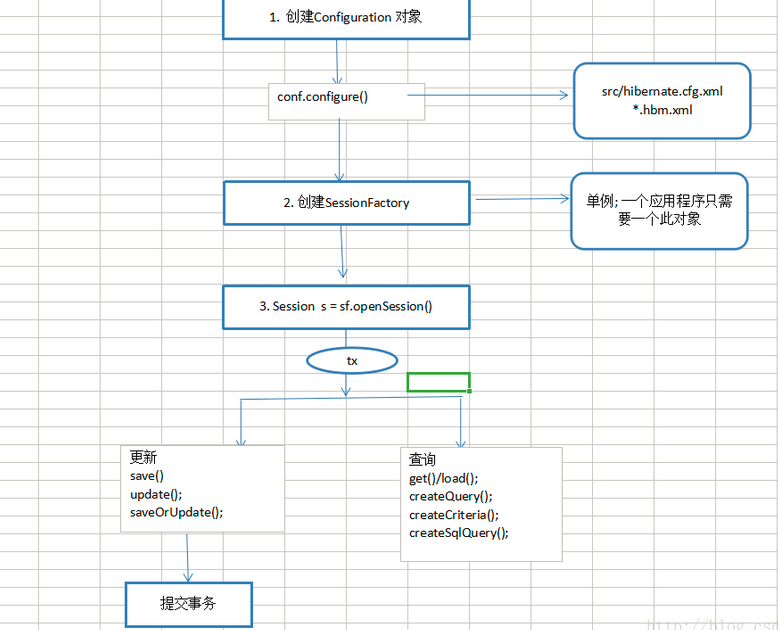
**解决了什么问题？**

JavaBean对象和数据表中的列存在映射关系，借助这种映射关系，自动生成SQL语句。

**处理流程**



**主配置文件如下：**

hibernate.cfg.xml

*<?***xml version="1.0"***?>***<!DOCTYPE hibernate-mapping PUBLIC  
 "-//Hibernate/Hibernate Mapping DTD 3.0//EN"  
 "http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd"*>****<!-- This mapping demonstrates content-based discrimination for the table-per-hierarchy mapping strategy, using a formula discriminator.-->  
<!--在domain包下-->*<**hibernate-mapping package="com.example.Entity"**>  
  
 *<!--类名为User，表名也为User-->* <**class name="Category" table="Category"**>  
  
 *<!--主键映射，属性名为id，列名也为id-->* <**id name="id" column="id"**>  
 *<!--根据底层数据库主键自动增长-->* <**generator class="native"**/>  
  
 </**id**>  
  
 *<!--非主键映射，属性和列名一一对应-->* <**property name="name" column="name"**/>  
 <**property name="createTime" column="create\_time"**/>  
 <**property name="updateTime" column="update\_time"**/>  
 </**class**>  
</**hibernate-mapping**>

**映射文件**

*<?***xml version="1.0"***?>***<!DOCTYPE hibernate-mapping PUBLIC  
 "-//Hibernate/Hibernate Mapping DTD 3.0//EN"  
 "http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd"*>****<!-- This mapping demonstrates content-based discrimination for the table-per-hierarchy mapping strategy, using a formula discriminator.-->  
<!--在domain包下-->*<**hibernate-mapping package="com.example.Entity"**>  
  
 *<!--类名为User，表名也为User-->* <**class name="Category" table="Category"**>  
  
 *<!--主键映射，属性名为id，列名也为id-->* <**id name="id" column="id"**>  
 *<!--根据底层数据库主键自动增长-->* <**generator class="native"**/>  
  
 </**id**>  
  
 *<!--非主键映射，属性和列名一一对应-->* <**property name="name" column="name"**/>  
 <**property name="createTime" column="create\_time"**/>  
 <**property name="updateTime" column="update\_time"**/>  
 </**class**>  
</**hibernate-mapping**>

一般映射文件和JavaBean文件放在同一个包下。

**更新操作**

session.save(obj);

session.update(obj);

session.saveOrUpdate(obj);

**主键查询**

session.get(javaBean.class, int id)

**HQL查询**

Query query = session.createQuery(“FROM User”);

List list = query.list();

**QBC查询**

Criteria criteria = session.createCriteria(User.class);

criteria.add(Restrictions.eq("id", 1));

List list = criteria.list();

**本地SQL查询**

SQLQuery sqlQuery = session.createSQLQuery("SELECT \* FROM user").addEntity(User.class);

List list = sqlQuery.list();