

Hibernate对象有三种状态，分别是：临时态(Transient)、持久态(Persistent)、游离态(Detached)。

**临时状态:**是指从对象通过new语句创建到被持久化之前的状态，此时对象不在Session的缓存中。

处在此状态的对象具备以下特点：

- 1不在Session缓存中，不与任何Session实例相关联。
- 2在数据库中没有与之对应的记录。

通常在下列情况下对象会进入临时状态：

- 1通过new语句创建新对象。
- 2执行delete()方法，对于游离状态的对象，delete()方法会将其与数据库中对应的记录删除；而对于持久化状态的对象，delete()方法会将其与数据库中对应的记录删除，并将其在Session缓存中删除。

例如：Object object = new Object();此时object为临时状态，数据库中没有对应的数据，Session缓存中也没有相关联的实例。

**持久化状态：**是指对象被持久化到Session对象被销毁之前的状态，此时对象在Session的缓存中。

处在此状态的对象具备以下特点：

- 1在Session缓存中，与Session实例相关联。
- 2在数据库中有与之对应的记录。
- 3Session在清理缓存的时候，会根据持久化对象的属性变化更新数据库。

通常在下列情况下对象会进入临时状态：

- 1执行save()或saveOrUpdate()方法，使临时对象转变为持久化对象。
- 2执行update()或saveOrUpdate()方法，使游离对象转变为持久化对象。
- 3执行load()或get()方法，返回的对象都是持久化对象。
- 4执行find()方法，返回List集合中存放的都是持久化对象。
- 5在允许级联保存的情况下，Session在清理缓存时会把与持久化对象关联的临时对象转变为持久化对象。

例如：Session session = factory.openSession(); object.setName("持久化对象"); session.save(object);此时object对象为持久化对象，Session缓存中有相关联的实例，数据库中有相应的记录。

**游离状态**：是指从持久化对象的Session对象被销毁到该对象消失之前的状态，此时对象不在Session的缓存中。

处在此状态的对象具备以下特点：

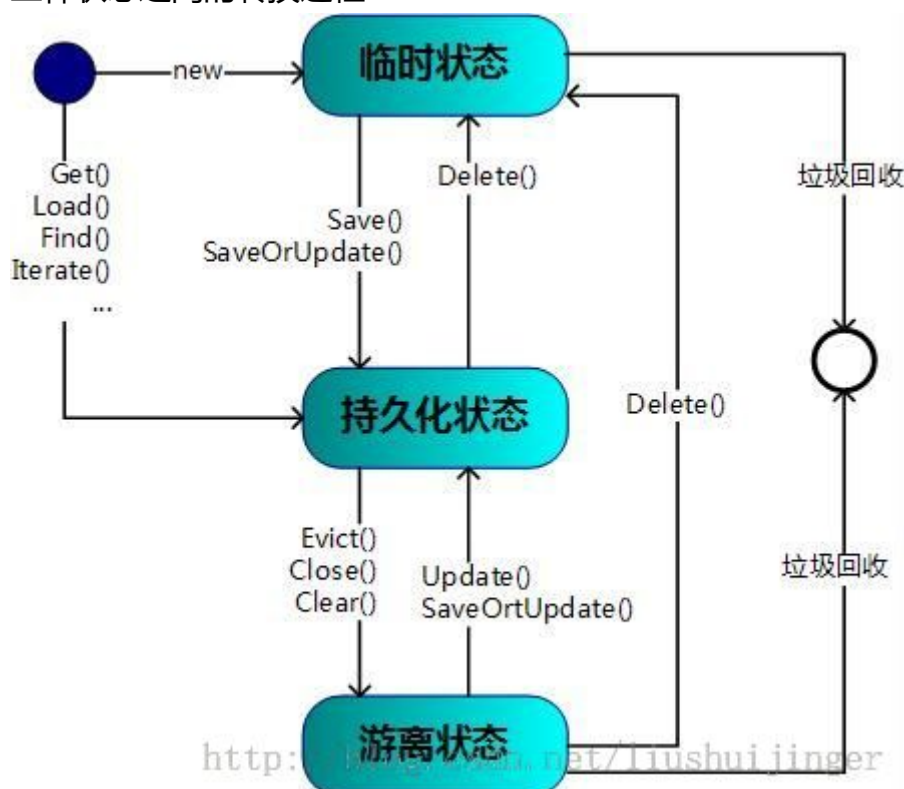
- 1不在Session缓存中，不与任何Session实例相关联。
- 2在数据库中有与之对应的记录（前提是没有其他Session实例删除该条记录）。

通常在下列情况下对象会进入临时状态：

- 1执行close()方法，将Session缓存清空，缓存中的所有持久化对象将转变成游离对象。
- 2执行evict()方法，能从缓存中删除一个持久化对象，使之转变成游离对象。

例如：session.close();此时上文的object对象为游离对象，Session缓存中没有有相关联的实例，数据库中有相应的记录。

三种状态之间的转换过程



对上图的解析：

- 1 当一个对象被new了以后此对象处于临时态（Transient）。
- 2 然后对此对象执行session的save() 或者saveOrUpdate()方法后，此对象被放入session的一级缓存进入持久态。
- 3 当再对此对象执行evict()/close()/clear()的操作后此对象进入游离态（Detached）。
- 4 游离态（Detached）和临时态（Transient）的对象由于没有被session管理会在适当的时机被java的垃圾回收站（garbage）回收。
- 5 执行session的get()/load()/find()/iternte()等方法从数据库里查询的到的对象,处于持久态（Persistent）。
- 6 当对数据库中的纪录进行update()/saveOrUpdate()/lock()操作后游离态的对象就过渡到持久态。
- 7 处于持久态（Persistent）与游离态（Detached）的对象在数据库中都有对应的记录。
- 8 临时态（Transient）与游离态(Detached)的对象都可以被回收可是临时态的对象在数据库中没有对应的纪录,而游离态的对象在数据库中有对用的纪录。

三种状态在程序的转换过程：

代码	对象的生命周期状态	对象的状态
Object object = new Object();	开始生命周期	开始生命周期
Session session = factory.openSession(); Transaction tx = session.beginTransaction();	在生命周期中	临时状态
session.save(object);	在生命周期中	转变为持久化状态
tx.commit();	在生命周期中	持久化状态
session.close();	在生命周期中	转变为游离状态
System.out.println(object.getName());	在生命周期中	游离状态
object = null;	结束生命周期	结束生命周期

三种状态里面，只有持久化状态在Session缓存中有相关联的实例，临时状态跟游离状态都没有。临时状态数据库里没有对应的记录，其他两种可能又记录，也可能没有记录。