# 级联操作：

## 解释：当程序员进行某个操作的时候有hibernate自动帮我们完成

## 比如: Department <---->Student 对象关系，我希望当我删除一个department ,那么就自动删除该部门的所有学生?

## 再比如: bbs项目

## 主帖<---->回帖 , 把主帖删除，那我们就希望把该主帖的回帖自动删除，这样我们可以使用级联(cascade)操作

\

案例:如何配置级联操作，当删除某个部门的时候，我们自动删除其学生.

首先我们在 配置文件中修改:

<!-- 配置one-to-many关系

cascade="delete" 当删除该部门的时候(主对象，则级联删除它的学生从对象) -->

<set name="stus" cascade="delete">

<!-- 指定Student类对应的外键 -->

<key column="dept\_id" />

<one-to-many class="Student" />

</set>

java代码中操作:

//演示删除级联

//获取到某个部分

Department department=(Department) s.get(Department.class, 41);

s.delete(department);

* 演示save-update

配置文件:

<set name="stus" cascade="save-update">

<!-- 指定Student类对应的外键 -->

<key column="dept\_id" />

<one-to-many class="Student" />

</set>

代码：

//添加学生

**Department department=new Department();**

**department.setName("业务部门3");**

**Student stu1=new Student();**

**stu1.setName("顺平6");**

**// stu1.setDept(department);**

**Student stu2=new Student();**

**stu2.setName("小明6");**

**// stu2.setDept(department);**

**Set<Student> students=new HashSet<Student>();**

**students.add(stu1);**

**students.add(stu2);**

**department.setStus(students);**

**s.save(department);**

说明:

① 在集合属性和普通属性中都能使用cascade

② 一般讲cascade配置在one-to-many(one的一方,比如Employee-Department),和one-to-one(主对象一方)

# 一。级联删除

比如现在我们使用one to many中

Department 中的 映射文件

<!-- 下面这个cascade 为delete时候 当删除 department 时候，同时删除 这个对象下的学生，也就是从对象 -->

<set name=*"studentManys"* cascade=*=*"delete" >

<!-- 指定Student类对应的外键 -->

<key column=*"dept\_id"* />

<one-to-many class=*"StudentMany"* />

</set>

# 4.然后进行测试

Department department = (Department) session.get(Department.**class**, 4);

session.delete(department);

# 5.发现成功了

# 二。级联更新下面这个是在one \_to\_many中进行的，我们也可在学生中 添加cascade=*"save-update"* > 在 many-to-one中，一般是在主对象中

<set name=*"studentManys"* cascade=*"save-update"* >

<!-- 指定Student类对应的外键 -->

<key column=*"dept\_id"* />

<one-to-many class=*"StudentMany"* />

</set>

StudentMany studentMany = **new** StudentMany();

studentMany.setName("woshinidei");

Department dept = **new** Department();

dept.setName("english");

//下面这个在测试级联的时候进行 注释了因为这样不能够成功，因为他是 one开始

// studentMany.setDept(dept);

// session.save(dept);

Set<StudentMany> students=**new** HashSet<StudentMany>();

students.add(studentMany);

dept.setStudentManys(students);

session.save(dept);