

## 1 分页

```
setFirstResult(0),(从0开始)
setMaxResults(5),每页显示5条数据
public void Test1() throws Exception{
    Configuration config=new Configuration().configure();
    SessionFactory sessionFactory= config.buildSessionFactory();
    Session session=null;
    Transaction tr=null;
    try{
        session= sessionFactory.openSession();
        tr=session.beginTransaction();
        String hql="select eage from Emp order by eage";
        Query query= session.createQuery(hql).setFirstResult(0).setMaxResults(5); //从0开始
        List<Integer> list = query.list();
        for(Integer message:list){
            System.out.println(message);
        }
        tr.commit();
    }catch(Exception e){
        tr.rollback();
    }finally{
        if(session!=null){
            session.close();
        }
        if(sessionFactory!=null){
            sessionFactory.close();
        }
    }
}
```

Hibernate: select emp0\_eage as col\_0\_0\_ from emp emp0\_ order by emp0\_eage lim  
21  
21  
24  
24  
24

## 2 表连接

Hibernate 支持内链接和外链接 ( 左连接 , 右连接 )

hql: from Emp e inner join fetch e.dept; 内链接

hql: from Emp e left join fetch e.dept; 左连接

hql : from Emp e right join fetch e.dept 右连接

```
public void Test2() throws Exception{ //实体
```

```
    Configuration config=new Configuration().configure();
```

```

SessionFactory sessionFactory= config.buildSessionFactory();
Session session=null;
Transaction tr=null;
try{
session= sessionFactory.openSession();
tr=session.beginTransaction();
Query query=session.createQuery("from Emp e inner join fetch e.dept where eage<

List<Emp> list = query.list();

for(Emp user:list){
System.out.println(user.getEname());
System.out.println(user.getDept().getDaddress());
tr.commit();
}catch(Exception e){
tr.rollback();
}finally{
if(session!=null){
session.close();
}
if(sessionFactory!=null){
sessionFactory.close();
}
}
}

```

结果

Hibernate: select emp0\_.eid as eid2\_0\_, dept1\_.did as did1\_1\_, emp0\_.ename as enan  
emp0\_.eage as eage2\_0\_, emp0\_.esal as esal2\_0\_, emp0\_.did as did2\_0\_, dept1\_.dnan  
dname1\_1\_, dept1\_.daddress as daddress1\_1\_ from emp emp0\_ inner join dept dept  
emp0\_.did=dept1\_.did where eage<30

白百何

301

文章

301

林月如

301

刘诗诗

302

3 批量更新

将年龄在25岁一下的员工改成25岁

hql="update Emp e set e.eage=25 where e.eage<25";

删除25岁一下的员工

hql="delete Emp e where e.eage<25";

使用executeUpdate()方法必须启用事务。

```
public void Test3() throws Exception{
```

```
Configuration config=new Configuration().configure();
```

```
SessionFactory sessionFactory= config.buildSessionFactory();
```

```

Session session=null;
Transaction tr=null;
try{
session= sessionFactory.openSession();
tr=session.beginTransaction();
String hql="update Emp e set e.eage=25 where e.eage<25";
Query query=session.createQuery(hql);
query.executeUpdate();
tr.commit();
}catch(Exception e){
tr.rollback();
}finally{
if(session!=null){
session.close();
}
if(sessionFactory!=null){
sessionFactory.close();
}
}
}

```

#### 4 Hibernate 中使用SQL

HQL不是万能的，无法执行插入语句和非常复杂的查询，Hibernate 也支持SQL查询。通过createSQLQuery(sql)即可  
sql语句中存在问号，同样使用setParameter(位置，属性值)方法设置。问号的位置从0开始  
executeUpdate执行。事务提交后数据库开始工作。

```

public void Test4() throws Exception{
Configuration config=new Configuration().configure();
SessionFactory sessionFactory= config.buildSessionFactory();
Session session=null;
Transaction tr=null;
try{
session= sessionFactory.openSession();
tr=session.beginTransaction();
String sql="insert into emp (ename,eage) values (?,?)";
session.createSQLQuery(sql).setParameter(0, "曹雪芹").setParameter(1, 22).executeU
tr.commit();
}catch(Exception e){
tr.rollback();
}finally{
if(session!=null){
session.close();
}
if(sessionFactory!=null){
sessionFactory.close();
}
}
}

```

}

