hibernate -- HQL语句总结

1.查询整个映射对象所有字段

```
//直接from查询出来的是一个映射对象，即：查询整个映射对象所有字段
String hql = "from Users";
Query query = session.createQuery(hql);
List users = query.list();
for(Users user : users){
System.out.println(user.getName() + " : "+ user.getPasswd() + " : "+ user.getId());
}
输出结果为：
name1 : password1 : 1
name2 : password2 : 2
name3 : password3 : 3
```

2.查询字段

```
//查询其中几个字段
String hql = " select name,passwd from Users";
Query query = session.createQuery(hql);
//默认查询出来的list里存放的是一个Object数组
List list = query.list();
for(Object[] object : list){
String name = (String)object[0];
String passwd = (String)object[1];
System.out.println(name + " : "+ passwd);
}
输出结果为：
name1 : password1
name2 : password2
name3 : password3
```

3.修改默认查询结果(query.list())不以Object[]数组形式返回，以List形式返回

```
//查询其中几个字段,添加new list(),注意list里的l是小写的。也不需要导入包,这样通过query.list()出来的list里存放的不再
是默认的Object数组了，而是List集合了
String hql = " select new list(name,passwd) from Users";
Query query = session.createQuery(hql);
//默认查询出来的list里存放的是一个Object数组，但是在这里list里存放的不再是默认的Object数组了，而是List集合了
List list = query.list();
for(List user : list){
String name = (String)user.get(0);
String passwd = (String)user.get(1);
System.out.println(name + " : "+ passwd);
}
/**
输出结果为：
name1 : password1
name2 : password2
name3 : password3
*/
```

4.修改默认查询结果(query.list())不以Object[]数组形式返回，以Map形式返回

```
//查询其中几个字段,添加new map(),注意map里的m是小写的。也不需要导入包,这样通过query.list()出来的list里存放的不再
是默认的Object数组了，而是map集合了
String hql = " select new map(name,passwd) from Users";
Query query = session.createQuery(hql);
//默认查询出来的list里存放的是一个Object数组，但是在这里list里存放的不再是默认的Object数组了，而是Map集合了
List list = query.list();
for(Map user : list){
//一条记录里所有的字段值都是map里的一个元素,key是字符串0,1,2,3....，value是字段值
//如果将hql改为：String hql = " select new map(name as username,passwd as password) from Users";,那么key将不是字符串0,1,2...
了，而是"username","password"了
String name = (String)user.get("0");//get("0");是get(key),注意:0,1,2...是字符串，而不是整形
String passwd = (String)user.get("1");
System.out.println(name + " : "+ passwd);
}
/**
输出结果为：
name1 : password1
name2 : password2
name3 : password3
*/
```

5.修改默认查询结果(query.list())不以Object[]数组形式返回，以自定义类型返回

6.条件查询

```
//条件查询，参数索引值从0开始，索引位置。通过setString,setParameter设置参数
String hql = "from Users where name=? and passwd=?";
Query query = session.createQuery(hql);
//第1种方式
// query.setString(0, "name1");
// query.setString(1, "password1");
//第2种方式
query.setParameter(0, "name1",Hibernate.STRING);
query.setParameter(1, "password1",Hibernate.STRING);
List list = query.list();
for(Users users : list){
System.out.println(users.getId());
}
```

```
//条件查询，自定义索引名(参数名):username,:password.通过setString,setParameter设置参数
String hql = "from Users where name=:username and passwd=:password";
Query query = session.createQuery(hql);
//第1种方式
// query.setString("username", "name1");
// query.setString("password", "password1");
//第2种方式,第3个参数确定类型
query.setParameter("username", "name1",Hibernate.STRING);
query.setParameter("password", "password1",Hibernate.STRING);
List list = query.list();
for(Users users : list){
System.out.println(users.getId());
}
```

```
//条件查询,通过setProperties设置参数
String hql = "from Users where name=:username and passwd=:password";
Query query = session.createQuery(hql);
//MyUser类的2个属性必须和:username和:password对应
MyUser myUser = newMyUser("name1","password1");
query.setProperties(myUser);
List list = query.list();
for(Users users : list){
System.out.println(users.getId());
}
```

7.update 数据

执行SQL语句（为什么要用SQL语句，我想是为了执行某些复杂的SQL语句吧）

```
String sql="update Table set field = 'test'"
Session session = HibernateSessionFactory.getSession();
session.createSQLQuery(sql).executeUpdate();
ts.commit();
```