

# 1、Hibernate 查询方式简单介绍

## 1、对象导航查询

(1) 在学生老师的例子中，先根据 id 查询老师，在根据老师查询所有包含的学生。

## 2、OID查询

(1) 根据 id 查询某一条记录， 返回对象。

## 3、hql 查询

(1) 使用 Query 对象，写 hql 语句实现查询。

## 4、QBC 查询

(1) 使用 Criteria 对象， 进行查询。

## 5、本地 sql 查询

(1) 使用 SQLQuery对象，使用本地SQL 实现查询。

# 2、对象导航查询

1、查询某个老师中的所有学生过程，使用对象导航查询实现

2、详细代码：

```
1      Teacher teacher = session.get(Teacher.class, 2);
2      Set<Student> students = teacher.getStudents();
3
4      for (Student student : students) {
5          System.out.println(student);
6      }
```

通过一对多关系中一这一方查询对应的多的一方为 对象导航查询。

# 3、OID查询

1、根据 id 查询记录 (即调用 session 里面的 get 方法)

## 2、参考代码

```
1 Teacher teacher = session.get(Teacher.class, 2);
```

## 4、HQL查询

1、hql : hibernate query language, hibernate 提供的一种查询语言, hql 语言与普通 sql 语言很相似。

区别: 普通 sql 操作数据库的表和字段, hql 操作实体类和属性

### 2、常用的 hql 语句

- (1)、查询所有: from 实体类名称
- (2)、条件查询: from 实体类名称 where 属性名 = ?
- (3)、顺序查询: from 实体类名称 order by 实体类属性名称 asc/desc
- (4)、投影查询: select 实体类名称1, 实体类名称2 from 实体类名称
- (5)、聚集函数的使用: select count(\*) from 实体类名称

### 3、使用 hql 查询操作的时候, 使用 query 对象

- (1)、创建 Query 对象, 写 hql 语句
- (2)、调用 Query 里面的方法得到结果

## 4.1 查询所有 (掌握)

### 1、查询所有老师记录

- (1)、创建 Query 对象, 写 hql 语句
- (2)、调用 Query 里面的方法得到结果

```
1 Query query = session.createQuery("from Teacher");
2 List<Teacher> list = query.list();
3 for (Teacher object : list) {
4     System.out.println(object.getName());
5     System.out.println();
6 }
```

## 4.2 条件查询（掌握）

1、hql 条件查询语句：

from 实体类名称 where 实体类属性名称 = ? and 实体类属性名称 = ?  
form 实体类名称 where 实体类属性名称 like ?

2、参考代码：

条件查询

```
1      String hql = "from Teacher t where t.name = ? and t.teaId = ?";
2      Query query = session.createQuery(hql);
3  //      hql的占位符的索引和jdbc不一样，jdbc是从1开始的，hql是从0开始的。
4      query.setParameter(0, "wk3");
5      query.setParameter(1, 2);
6      List<Teacher> list = query.list();
7      for (Teacher object : list) {
8          System.out.println(object);
9          System.out.println();
10     }
```

模糊查询

```
1      String hql = "from Teacher t where t.name like ?";
2      Query query = session.createQuery(hql);
3      query.setParameter(0, "%w%");
4
5      List<Teacher> list = query.list();
6      for (Teacher object : list) {
7          System.out.println(object.getTeaId());
8          System.out.println();
9      }
```

## 4.3 排序查询（掌握）

1、排序语句写法

(1) from 实体类名称 order by 实体类属性名称 asc/desc

## 2、实现代码：

### 倒序

```
1      String hql = "from Teacher t order by t.id desc";
2      Query query = session.createQuery(hql);
3
4      List<Teacher> list = query.list();
5      for (Teacher object : list) {
6          System.out.println(object.getTeaId());
7          System.out.println();
8      }
```

### 顺序

```
1      String hql = "from Teacher t order by t.id asc";//desc";
2      Query query = session.createQuery(hql);
3
4      List<Teacher> list = query.list();
5      for (Teacher object : list) {
6          System.out.println(object.getTeaId());
7          System.out.println();
8      }
```

## 4.4 分页查询（掌握）

### 1、mysql 实现分页是通过 关键字 limit 实现的

```
1  select * from tb_teacher limit 0, 3
```

### 2、hql 实现中实现分页，是通过 hibernate 的 Query 对象封装的两个方法实现的分页的操作

`Query.setFirstResult();` 设置起始位置

`Query.setMaxResults();` 设置每页记录数

以下是参考代码：

```

1      String hql = "from Teacher";
2      Query query = session.createQuery(hql);
3      query.setFirstResult(0);
4      query.setMaxResults(2);
5
6      List<Teacher> list = query.list();
7      for (Teacher object : list) {
8          System.out.println(object.getTeaId());
9          System.out.println();
10     }

```

## 4.5 投影查询（会用）

- 1、投影查询：查询表中的部分字段称之为 投影查询。
- 2、投影查询 hql 语句写法：
  - (1) **select** 实体类属性名称1, 实体类属性名称2 **from** 实体类名称;
  - (2) **select** 后面不能写 \* 是不支持的。
- 3、参考代码：

```

1      String hql = "select t.id, t.name from Teacher t";
2      Query query = session.createQuery(hql);
3
4      List<Object[]> list = query.list();
5      for (Object[] object : list) {
6          System.out.println(object[0]);
7          System.out.println();
8      }

```

## 4.6 聚集函数使用（会用）

- 1、常用聚集函数 **count**、**sum**、**avg**、**max**、**min**
- 2、hql 聚集函数语句写法
  - (1) 查询表的记录数： **select count(\*) from** 实体类名称

### 3、参考代码：

```
1      String hql = "select count(*) from Teacher t";
2      Query query = session.createQuery(hql);
3
4      //使用query.uniqueResult()获得查询结果;
5      Object uniqueResult = query.uniqueResult();
6      System.out.println((long)uniqueResult);
```