Hibernate Named Query

* Nếu sử dụng câu lệnh HQL hoặc Native SQL nhiều lần sẽ có thể lộn xộn code vì tất cả các query sẽ rải rác trong cả project.
* Hibernate Named Query là các để dùng query bằng cách cho chúng 1 tên.
* Hibernate Named Query sẽ được đinh nghĩa ở 1 nơi và sử dụng ở mọi nơi trong project.

Sử dụng Named Query trong Hibernate :

* Viết câu lệnh HQL trong xml file để gọi HQL named query.
* Hoặc có thể sử dụng annotation @*NameQuery* trong entity.
* Để viết Hibernate Named Query có thể sử dụng <query> trong Hibernate mapping file hoặc *@NameQuery* trong entity.

Lợi ích của Named Query trong Hibernate:

* Sử dụng toàn project.
* Dễ dàng bảo trì sửa chữa.

Dùng Hibernate Named Query bằng cách dùng Anntation:

* HQL dùng annotation:
  + Sử dụng *@NameQueries* *và @NameQuery* annotations.
  + *@NameQuery* được dùng để tạo câu lệnh đơn.
  + *@NameQueries* được dùng để tạo nhiều query.

Query query = session.getNamedQuery(**"findUserById"**);  
query.setString(**"name"**, **"hoang"**);  
List<User> listuser = query.list();

@NamedQueries({  
 @NamedQuery(  
 name = **"findUserById"**,  
 query = **"from User where name = :name"** )  
})

@Entity  
**public class** User {  
 @Id @GeneratedValue  
 **private int id**;  
 @Column  
 **private** String **name**;  
 @Column  
 **private** String **address**;

}

* Native SQL bằng annotation

@NamedNativeQueries({  
 @NamedNativeQuery(  
 name = **"findUserById"**,  
 query = **"select \* from user where name = :name",**

resultClass = **User.class** )  
})

@Entity  
**public class** User {  
 @Id @GeneratedValue  
 **private int id**;  
 @Column  
 **private** String **name**;  
 @Column  
 **private** String **address**;

}

Dùng Hibernate Named Query bằng xml mapping file

* HQL

<query name="findUserById">

<![CDATA[from User where name = :name]]>

</query>

* Native SQL

<sql-query name="findUserById">

<return alias="user" class="model.User"/>

<![CDATA[select \* from user where name = :name]]>

</sql-query>