

SpringJPA复杂查询

1.核心方法

- 查询所有数据 `findAll()`
- 修改 添加数据 `S save(S entity)`
- 分页查询 `Page<T> findAll(Example example, Pageable pageable)`
- 根据实体类属性查询: `findByProperty (type Property);` 例如: `findByAge(int age)`
- 删除 `void delete(T entity)`
- 计数 查询 `long count()` 或者 根据某个属性的值查询总数 `countByAge(int age)`
- 是否存在 `boolean existsById(ID primaryKey)`

2.查询关键字

~~-and~~

And 例如: ~~findByCustNameAndCustLevel(String custName, String custLevel);~~

~~-or~~

Or 例如: ~~findByCustNameOrCustPhone(String custName, String custPhone);~~

~~-between~~

Between 例如: ~~CustLevelBetween(int max, int min);~~ //客户等级在min和max之间

~~"<"~~

LessThan 例如: ~~findByCustLevelLessThan(int max);~~ //<

~~">"~~

GreaterThan 例如: ~~findBySalaryGreaterThan(int min);~~

~~-like~~

Like 例如: ~~findByUsernameLikeAndUserageLike(String user,String userage);~~

~~-not like~~

NotLike 例如: ~~findByUsernameNotLike(String user);~~

~~"!="~~

Not 例如: ~~findByUsernameNot(String user);~~

~~-in~~

In 例如: ~~findByUsernameIn(Collection userList)~~ , 方法的参数可以是 Collection 类型, 也可以是数组或者不定长参数;

~~-order by~~

OrderBy 例如: ~~findByUsernameOrderByNameAsc(String user);~~直接通过name正序排序

~~-not in~~

NotIn 例如: ~~findByUsernameNotIn(Collection userList)~~ , 方法的参数可以是 Collection 类型, 也可以是数组或者不定长参数;

-

~~Top/Limit~~

~~查询方法结果的数量可以通过关键字来限制，**first** 或者 **top**都可以使用。**top/first**加数字可以指定要返回最大结果的大小 默认为1~~

```
User findFirstByOrderByLastnameAsc();  
User findTopByOrderByAgeDesc();  
Page<User> queryFirst10ByLastname(String lastname, Pageable pageable);  
Slice<User> findTop3ByLastname(String lastname, Pageable pageable);  
List<User> findFirst10ByLastname(String lastname, Sort sort);  
List<User> findTop10ByLastname(String lastname, Pageable pageable);
```

3.详细查询语法

关键词	示例	对应的sql片段
And	findByLastnameAndFirstname	... where x.lastname = ?1 and x.firstname = ?2
Or	findByLastnameOrFirstname	... where x.lastname = ?1 or x.firstname = ?2
Is, Equals	findByFirstname, findByFirstnameIs, findByFirstnameEquals	... where x.firstname = ? 1
Between	findByStartDateBetween	... where x.startDate between ?1 and ?2
LessThan	findByAgeLessThan	... where x.age < ?1
LessThanEqual	findByAgeLessThanEqual	... where x.age <= ?1
GreaterThan	findByAgeGreaterThan	... where x.age > ?1
GreaterThanEqual	findByAgeGreaterThanEqual	... where x.age >= ?1
After	findByStartDateAfter	... where x.startDate > ? 1
Before	findByStartDateBefore	... where x.startDate < ? 1
IsNull	findByAgeIsNull	... where x.age is null
IsNotNull, NotNull	findByAge(Is)NotNull	... where x.age not null
Like	findByFirstnameLike	... where x.firstname like ?1
NotLike	findByFirstnameNotLike	... where x.firstname not like ?1
Startingwith	findByFirstnameStartingwith	... where x.firstname like ?1 (parameter bound with appended %)
Endingwith	findByFirstnameEndingwith	... where x.firstname like ?1 (parameter bound with prepended %)
Containing	findByFirstnameContaining	... where x.firstname like ?1 (parameter bound wrapped in %)
OrderBy	findByAgeOrderByLastnameDesc	... where x.age = ?1 order by x.lastname desc
Not	findByLastnameNot	... where x.lastname <> ? 1
In	findByAgeIn(Collection<Age> ages)	... where x.age in ?1
NotIn	findByAgeNotIn(Collection<Age> ages)	... where x.age not in ? 1

关键词	示例	对应的sql片段
True	findByActiveTrue()	... where x.active = true
False	findByActiveFalse()	... where x.active = false
IgnoreCase	findByFirstnameIgnoreCase	... where UPPER(x.firstname) = UPPER(?1)

-