Hibernate Tip: Use a SubQuery in a CriteriaQuery

Question:

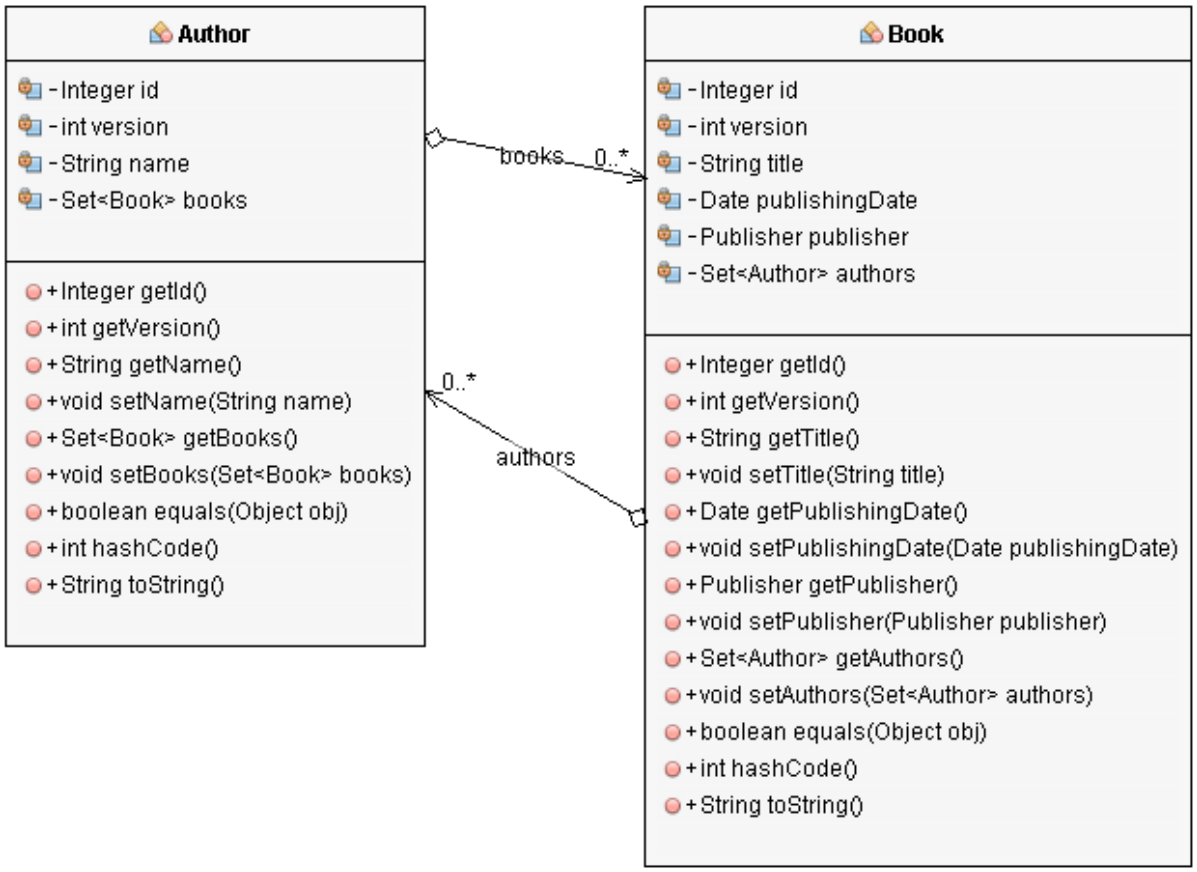
In one of our use cases, we use the Criteria API to build the query dynamically. Now I need to add a subquery. Does the Criteria API support subqueries? And how can I implement them?

Solution:

The Criteria API supports the same features as a JPQL query. So, you can use a subquery only in your WHERE but not in the SELECT or FROM clause.

Let's take a look at an example.

I use a simple model consisting of an *Author* and a *Book* entity and a many-to-many association between them.



Hibernate Tip: Use a SubQuery in a CriteriaQuery

In the following code snippet, I use the Criteria API to select all *Author*s who've written at least 3 *Book*s.

```
// create the outer query
CriteriaBuilder cb = em.getCriteriaBuilder();
CriteriaQuery cq = cb.createQuery(Author.class);
Root root = cq.from(Author.class);

// count books written by an author
Subquery sub = cq.subquery(Long.class);
Root subRoot = sub.from(Book.class);
SetJoin<Book, Author> subAuthors =
subRoot.join(Book_.authors);
sub.select(cb.count(subRoot.get(Book_.id)));
sub.where(cb.equal(root.get(Author_.id),
subAuthors.get(Author_.id)));

// check the result of the subquery
cq.where(cb.greaterThanOrEqualTo(sub, 3L));

TypedQuery query = em.createQuery(cq);
List authors = query.getResultList();
```

In the first step, I instantiate a *CriteriaQuery* which returns *Author* entities.

Then I call the *subquery* method on the *CriteriaQuery* to create a subquery that counts the *Book*s written by the *Author* which is selected by the outer query.

As you can see, I define the subquery in the same way as I create a *CriteriaQuery*. I first define the *Book* entity as the root and join it with the *Author* entity.

Then I use the *count* function to count the number of *Book*s in the SELECT clause. And after that, I compare the *id* of the *Author* entity which got selected in the outer query with the *id* of the *Author* selected in the subquery.

In the final step, I define the WHERE clause of the outer query. I want to select all *Author*s who have written at least 3 *Book*s. So, I use the *greaterThanOrEqualTo* method to check if the result of the subquery is greater or equal 3.

When you execute this *CriteriaQuery*, Hibernate generates the following SQL statement.

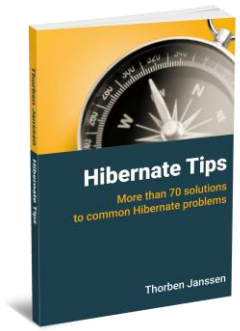
```
16:55:38,728 DEBUG [org.hibernate.SQL] -
    select
        author0_.id as id1_0_,
        author0_.firstName as firstNam2_0_,
        author0_.lastName as lastName3_0_,
        author0_.version as version4_0_
    from
        Author author0_
    where
        (
            select
                count(book1_.id)
            from
                Book book1_
            inner join
                BookAuthor authors2_
                  on book1_.id=authors2_.bookId
            inner join
                Author author3_
                   on authors2_.authorId=author3_.id
            where
                author0_.id=author3_.id
        )>=3
```

Learn more

Here are a few more Hibernate Tips using the Criteria API:

- Hibernate Tips: How to select a POJO with a Criteria Query
- Hibernate Tips: How to select multiple scalar values in a Criteria Query
- Hibernate Tips: How to call a user-defined function in a CriteriaQuery

Hibernate Tips Book



Get more recipes like this one in my book Hibernate Tips: More than 70 solutions to common Hibernate problems.

It gives you more than 70 ready-to-use recipes for topics like basic and advanced mappings, logging, Java 8 support, caching and statically and dynamically defined queries.