

Hibernate Tip: Map native query results to a POJO

Question:

My query is too complex for JPQL, and I have to use a native query. What is the easiest way to map the result of the query to a POJO?

Solution:

You can define the schema and table name with the schema and n

You can use a *@SqlResultSetMapping* to map the query result to a POJO. The following code snippet shows an example of such a mapping.

```
@SqlResultSetMapping(  
    name = "BookValueMapping",  
    classes = @ConstructorResult(  
        targetClass = BookValue.class,  
        columns = {  
            @ColumnResult(name = "title"),  
            @ColumnResult(name = "date")  
        }  
    )  
)
```

The *@ConstructorResult* annotation defines a constructor call of the *BookValue* class. The *@ColumnResult* annotations define how the columns of the result shall be mapped to the parameters of the constructor. In this example, Hibernate will perform a constructor call with the value of the title column as the first and the value of the date column as the second parameter.

Hibernate Tip: Map native query results to a POJO

When you provide the name of the `@SqlResultSetMapping` as the second parameter to the `createNativeQuery` method, Hibernate will apply the mapping to the query result. You can use it to map the results of all queries that return at least the parameters defined by the `@ColumnResult` annotations.

```
BookValue b = (BookValue) em.createNativeQuery(
    "SELECT b.publishingDate as date, b.title, b.id FROM
    book b WHERE b.id = 1", BookValueMapping")
    .getSingleResult();
```

Learn more:

@SqlResultSetMappings are a powerful feature which allows you to define complex mappings for native query results. You can read more about native queries and *@SqlResultSetMappings* in my free [Native Queries with Hibernate ebook](#).