

Hibernate Tip: Calculate entity attributes with @Formula

Question:

The value of one of my entity attribute gets calculate by an SQL function. How can I map that with Hibernate?

Solution:

You can use the *@Formula* annotation to provide an SQL snippet which Hibernate will execute when it fetches the entity from the database. The return value of the SQL snippet gets mapped to a read-only entity attribute.

I use it in the following example to calculate the age of an author.

```
@Entity
public class Author {

    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    @Column(name = "id", updatable = false, nullable =
false)
    private Long id;

    @Column
    private LocalDate dateOfBirth;

    @Formula(value = "date_part('year', age(dateOfBirth))")
    private int age;

    ...

    public int getAge() {
        return age;
    }
}
```

Hibernate Tip: Calculate entity attributes with @Formula

When Hibernate fetches an Author entity from the database, it adds the SQL snippet of the formula annotation to its SQL statement.

```
05:35:15,762 DEBUG [org.hibernate.SQL] –
select author0_.id as id1_0_,
       author0_.dateOfBirth as dateOfBi2_0_,
       author0_.firstName as firstNam3_0_,
       author0_.lastName as lastName4_0_,
       author0_.version as version5_0_,
       date_part('year', age(author0_.dateOfBirth))
       as formula0_
from Author author0_
where author0_.id=1
```

The *@Formula* annotation provides an easy way to map the result of an SQL snippet to an entity attribute. But it also has some downsides you should be aware of:

1. Hibernate executes the SQL snippet for every Author entity it fetches from the database. So better make sure, that you only use it for attributes you need in all of your use cases.
2. You need to provide a native SQL snippet to the @Formula annotation. This can affect the database portability of your application.