

Hibernate Tip: How to define schema and table names

Question:

How can I define the name of the database schema and table which will be used to persist my entity?

Solution:

You can define the schema and table name with the schema and name attributes of the *javax.persistence.Table* annotation. You can see an example of it in the following code snippet.

You just have to add the annotation to your entity class and set the name and schema attributes.

```
@Entity
@Table(name = "author", schema = "bookstore")
public class Author {...}
```

When you now use the entity, Hibernate uses the provided schema and table names to create the SQL statements.

The following code snippet persists a new *Author* entity and performs a query to get all *Author* entities with the given first name.

```
Author a = new Author();
a.setFirstName("firstName");
a.setLastName("lastName");
em.persist(a);

a = em.createQuery("SELECT a FROM Author a WHERE
firstName = 'firstName'", Author.class).getSingleResult();
```

Hibernate Tip: How to define schema and table names

As you can see in the following log output, Hibernate persists the *Author* entity to the author table in the bookstore database schema and performs the SELECT statement on the same table.

```
06:27:24,009 DEBUG [org.hibernate.SQL] – insert into  
bookstore.author (firstName, lastName, version, id) values (?,  
?, ?, ?)
```

```
06:27:24,022 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 bookstore.author author0_ where  
author0_.firstName='firstName'
```

Learn more:

If you want to dive deeper into Hibernate's mapping annotations and other advanced features, you should have a look at my [Advanced Hibernate Online Training](#).