COURSEWARE

Entities

Contents

Overview

Entities are classes that represent tables in the database.
Entity classes are annotated with `@Entity` and must have at least one field marked as an `@Id`, a *default* constructor *and* getters and setters.

@Id

Denotes that the field represents a **Primary Key**.
Can be applied to multiple columns in the case of a **composite Key**.

```
@Id
private Long id;
```

@GeneratedValue

Sets a field to **Auto Increment**, can *only* be applied to a numeric field marked as an `@Id`.

```
@Id
@GeneratedValue
private Long id;
```

GenerationType

There are a few strategies available for generating id's in Spring Boot but in order to set up an `AUTO_INCREMENT` field we will be using the `Identity` strategy.

```
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private long id;
```

@Column

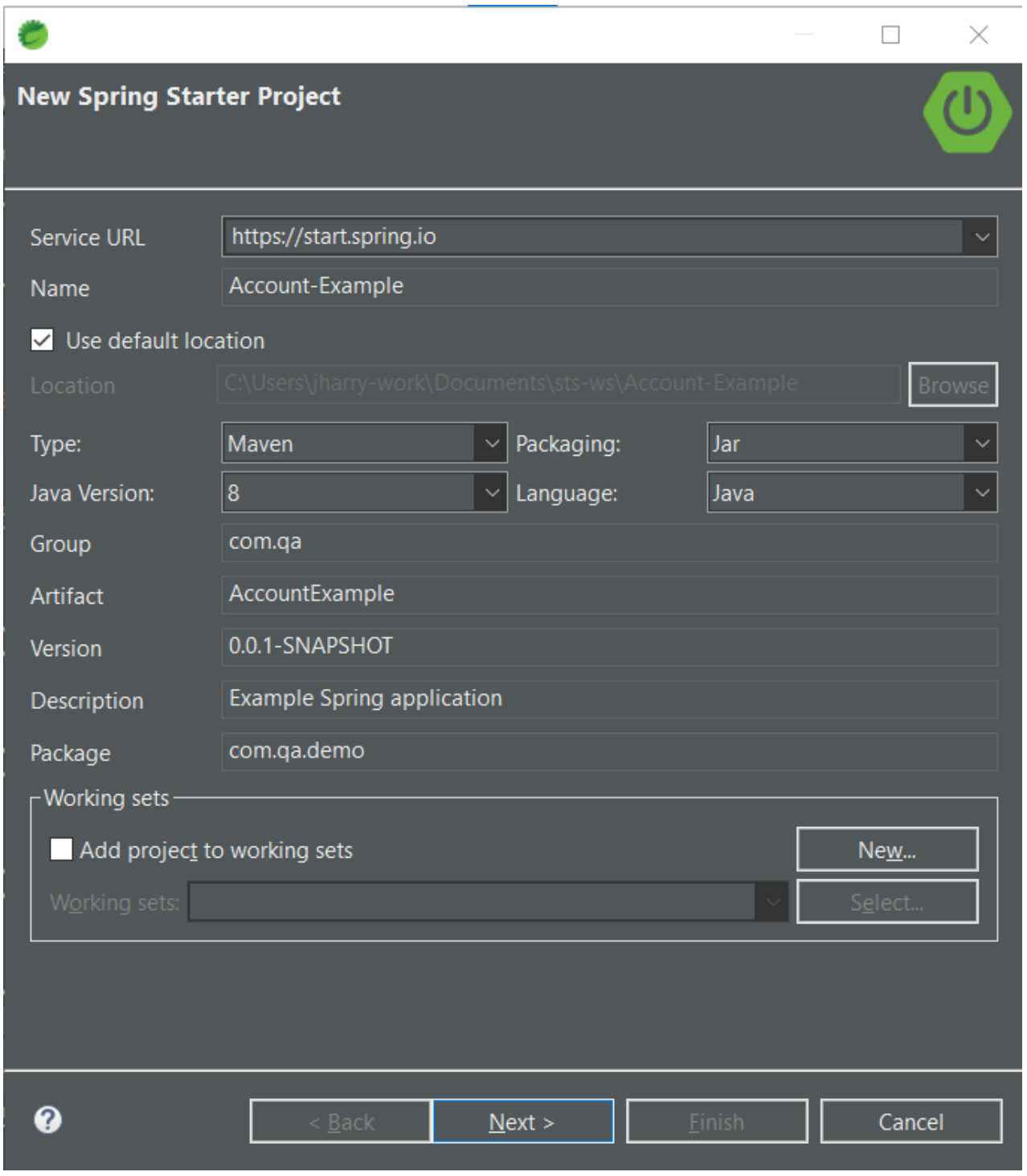
Fields in an entity are mapped to columns in that table by default (with variable names in **camelCase** being converted to **snake_case** for the column names) but the addition of `@Column` allows for the *configuration* of those fields.
In the following example the `username` field is set to be both **unique** and **not null** and the `password` field is set to be **not null** whereas the rest of the fields will default to being not-unique and nullable.

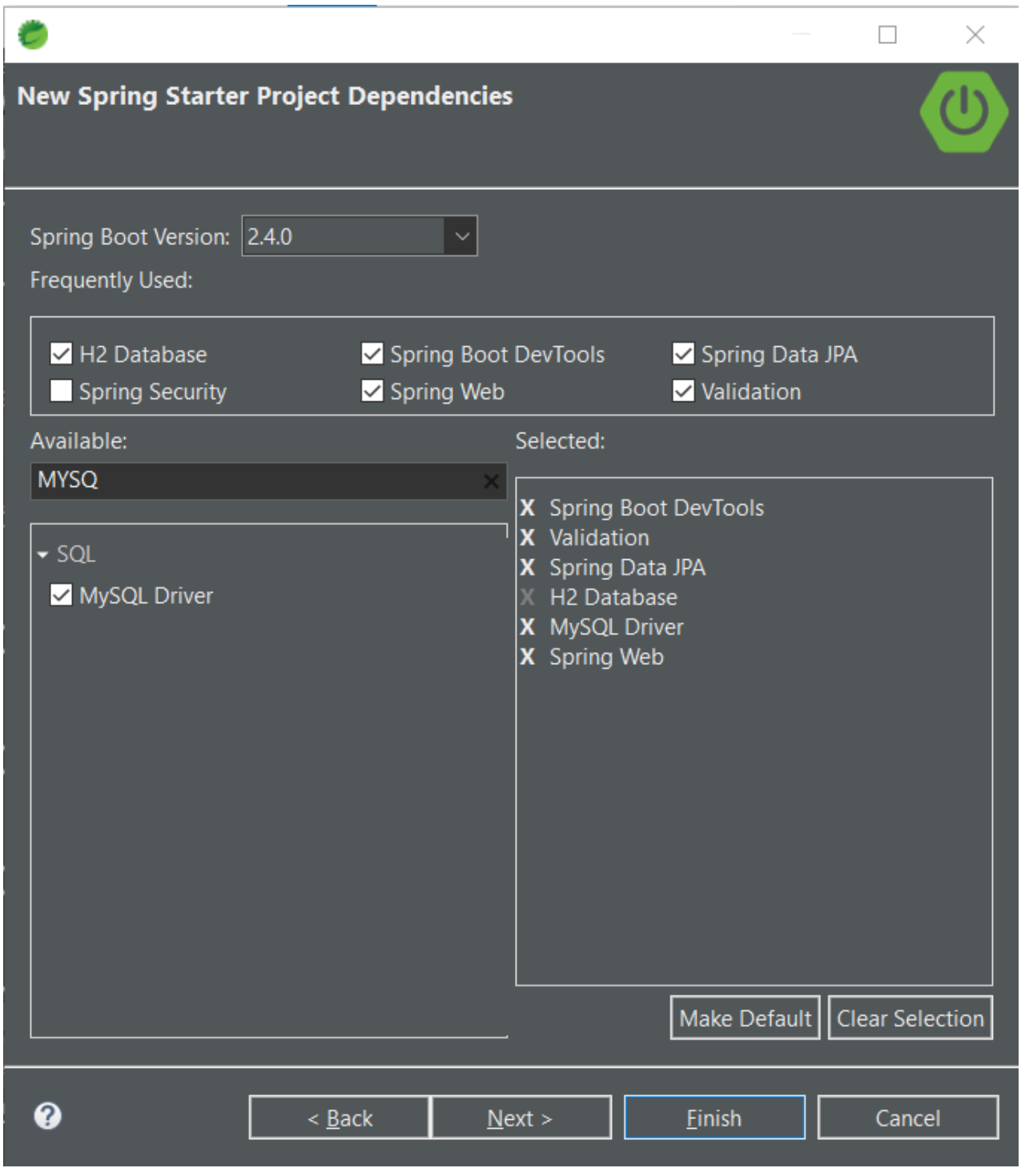
Tutorial

▶ Example Entity
The above example shows a possible way to model a user table as a Spring entity.

Exercises

Create a new **Spring Starter Project** with this configuration:





Create a *valid* `Account` entity with:

1. A String **accountNumber** field that is both **unique** and **not null**.
2. A Long **id** field that is set to `AUTO_INCREMENT`.
3. A String **name** field.

Make sure that your `Account` entity is in an appropriate package.