Setting up XAMPP:

- Download XAMPP version **5.6.33 (IMPORTANT!)** (https://www.apachefriends.org/download.html)
- Install it.

Save a new Bookshelf

insert into bookshelf (capacity, location) values (15, "Middle")

OR

insert into bookshelf (capacity, location) values (15, "Middle")

OR (SprinData equivalent query)

bookshelfRepository.save(someShelf);

Save a new Book (linked to a Bookshelf)

insert into book (author, bookshelf_id, title) values ("Mr Java", 1, "Java's greatest hits")

OR

insert into book (author, title, bookshelf_id) values ("Mr Java", "Java's greatest hits", 1)

OR (SprinData equivalent query)

bookRepository.save(someBook);

Get information about all Book

select book0_.id as id1_0_, book0_.author as author2_0_, book0_.bookshelf_id as bookshel4_0_, book0_.title as title3_0_ from book book0_

OR

select * from book

OR (SprinData equivalent query)

List<Book>findAll()

Get information about all Book by a specific author

select book0_.id as id1_0_, book0_.author as author2_0_, book0_.bookshelf_id as bookshel4_0_, book0_.title as title3_0_ from book book0_ where book0_.author="Mr Java"

OR

select * from book where book.author="Mr Java"

OR (SprinData equivalent query)

List<Book>findByAuthor(String author)

Get information about all Book with a title containing a certain text

select book0_.id as id1_0_, book0_.author as author2_0_, book0_.bookshelf_id as bookshel4_0_, book0_.title as title3_0_ from book book0_ where book0_.title like "%Java%"

OR

select * from book where book.title like "%Java%"

OR (SprinData equivalent query)

List<Book> findByTitleContaining (String text);

Get information about all Bookshelf with a capacity greater than a minimum

select bookshelf0_.id as id1_1_, bookshelf0_.capacity as capacity2_1_, bookshelf0_.location as location3_1_ from bookshelf bookshelf0_ where bookshelf0_.capacity > 9

OR

select * from bookshelf where bookshelf.capacity > 9

OR (SprinData equivalent query)

List<Bookshelf> findByCapacityGreaterThan (Integer minimum);

Get information about all Book that are placed on a certain Bookshelf (selected by its location property)

select book0_.id as id1_0_, book0_.author as author2_0_, book0_.bookshelf_id as bookshel4_0_, book0_.title as title3_0_ from book book0_ left outer join bookshelf bookshelf1_ on book0_.bookshelf_id=bookshelf1_.id where bookshelf1_.location="Top"

OR

select book.* from book left outer join bookshelf on book.bookshelf_id=bookshelf.id where bookshelf.location="Top"

OR (SprinData equivalent query)

List<Book> findByBookshelfLocation (String location);

For data table creation, Hibernate does these lines:

Hibernate: create table book (id bigint not null auto_increment, author varchar(255), title varchar(255), bookshelf_id bigint, primary key (id))

Hibernate: create table bookshelf (id bigint not null auto_increment, capacity integer, location varchar(255), primary key (id))

Hibernate: alter table book add constraint FKd8d8xgv94y9sxs6yn36uq4jb foreign key (bookshelf_id) references bookshelf (id)