Exercises

Summary

Build a full CRUD system for creating, retrieving, updating, and deleting information in an H2 database.

Less explicit instruction given today to properly prepare you for the exam next week!

Step 1

Start with a fresh Spring Boot project called ex52_fullCRUD using Spring Web, Thymeleaf, Lombok, H2, and Spring JDBC repos.

Create a new class called Appointment in a ca.sheridancollege.<yourUserName>.beans package. Inside, create the following fields:

- A private Long id
- A private String firstName
- A private String email
- a private LocalDate appointmentDate
- A private LocalTime appointmentTime

Friendly note and possibly helpful tip for later: to bind this properly with Thymeleaf as a model object, I typed the following:

```
@DateTimeFormat(pattern = "yyyy-MM-dd")
private LocalDate appointmentDate;
@DateTimeFormat(pattern = "HH:mm")
private LocalTime appointmentTime;
```

Lombok it for ease with an @Data and an @NoArgsConstructor.

Note: this part at least is similar to an exercise we did previously, obviously, but this time we're going to CRUD it with a database!

Step 2

Create a DatabaseConfig class a ca.sheridancollege.<yourUserName>.database package and annotate the class with @Configuration.

Inside, copy and paste the following code to make it go!

```
@Bean
public NamedParameterJdbcTemplate namedParemterJdbcTemplate(DataSource dataSource) {
	return new NamedParameterJdbcTemplate(dataSource);
}
```

Now create a DatabaseAccess class in the same package. @Autowire in your NamedParameterJdbcTemplate as follows:

```
@Autowired
protected NamedParameterJdbcTemplate jdbc;
```

Fill in any required methods following our previous lectures and exercises. You need a method to insertAppointment an Appointment to a database, as well as getAllAppointments, deleteAppointmentById, and getAppointmentById. Follow the lectures and exercises!

Don't forget to create a schema.sql for your database structure, and also a data.sql to fill in a row or two of sample data! Your database structure should match your Appointment POJO as closely as possible! Research H2 SQL column types! There are tons of them! Here's what may be helpful hints:

```
CREATE TABLE appointment(
	id LONG PRIMARY KEY AUTO_INCREMENT,
	firstName VARCHAR(255),
	email VARCHAR(255),
	appointmentDate DATE,
	appointmentTime TIME
);
```

```
INSERT INTO appointment(firstName, email, appointmentDate, appointmentTime) VALUES
('Frank', 'frank@frank.com', '2020-01-01', '12:00:00'),
('Sally', 'sally@gmail.com', '2020-02-02', '08:15:00'),
('Sue', 'sue@yahoo.ca', '2020-03-03', '14:30:00'),
('Tim', 'jaspreet@outlook.com', '2020-04-04', '10:15:00');
```

Step 3

Create a Controller called AppointmentController in a ca.sheridancollege.<yourUserName>.controllers package.

Inside, create methods mapped to "/", insertAppoitment, editAppointment, and deleteAppointment. Be sure to use the path variable technique we learned in class today, though I fully expect you'll need to come back and edit in several places as you go.

Be sure to use the trick we saw today where you getAppointmentById, add it to the Model, then deleteAppointmentByID, before also adding getAllAppointments to the Model for a refreshed full list.

Step 4

Create an index.html page to make it go!

Your index.html page should automatically display all entered Appointments at launch (i.e. loading of "/"), along with unique Edit and Delete buttons for each, and then provide your users a simple form where they can enter additional Appointment info following the POJO.

Style, style, style!!!

Step 5

Expect the exam to be a similar application! Do this, then ask questions about ANYTHING next class if you get or got stuck ANYWHERE – in the entire course! Happy to help, but it's really on you to make it go.

Upload a screenshot of a bunch of appointments in your HTML list. Edit one that would be obvious, and delete two of them. Take another screenshot and upload it as well. Remember to name your screenshots using your own name plus a 1 or 2 for their filenames.

Have a great midterm and a great rest of the semester! 😊