Tutorial 6 – Spring Security with JDBC

❖ **Contents:**

▪ Build JDBC-based login feature with Spring Security.

❖ **Instructions:**

1. Create a Spring Boot project with the dependencies for a typical web app and Spring Security:

  ▪ Spring Web

  ▪ Spring Data JPA

  ▪ MySQL Driver or MariaDB Driver

  ▪ Thymeleaf

  ▪ Spring Security

2. Create a `SecurityCfg.java` file for Spring Security configuration. Refer to Lecture 6 for details of how to create this file. To sum up, you need to configure the following:

  ▪ Allow everyone to access all URLs except the ones in the `/member` directory, which are only accessible for authenticated users.

  ▪ Use the Form Login method to authenticate users. Use all default URLs for login and logout.

  ▪ Disable CSRF.

3. Configure the DataSource of your Spring Boot application. Refer to Tutorial 2 or Lecture 6 for details.

4. Create necessary tables (`users` and `authorities`) in your database to store users. Refer to Lecture 6 for the SQL script to create these tables.

5. In your `SecurityCfg.java` file, add a bean of type `UserDetailsManager` which returns a `JdbcUserDetailsManager`. Sample code below (change user and password to your liking):

```java
@Bean
UserDetailsManager users(DataSource dataSource) {
    UserDetails user = User.builder()
            .username("quan")
            .password(passwordEncoder().encode("123"))
```

```
            .roles("USER")
            .build();
    JdbcUserDetailsManager users =
            new JdbcUserDetailsManager(dataSource);
    users.createUser(user);
    return users;
}
```

6. Note that the UserDetailsManager above requires a PasswordEncoder to encode password. You also need to add a PasswordEncoder bean to your configuration file:

```
@Bean
PasswordEncoder passwordEncoder() {
    return new BCryptPasswordEncoder();
}
```

❖ **Run, debug & submit your project**

Once finished, visit the login URL and try to login with the user that you created in your SecurityCfg.java file. After verifying that your code works, compress your project into a .zip file and submit the file to the tutorial's submission box.