

Assignments 1

Implement a menu driven program using Daos. Provide following functionalities.

Add Book
Update Book
Delete Book by Id
Find Book by Id
Find All Books
Find Books by Subject
Find All Subjects
Add Customer
Find Customer by Email
Find Customer by Email And Password
Change Password
Update Customer
Delete Customer by Id

Assignments 2

Create spring bean class "Arithmetic" with fields "num1" and "num2". Write business logic methods "int add()", "subtract()", "multiply()", "divide()". Use annotation config to create Arithmetic bean "arith" with fixed values e.g. num1=22, num2=7. In main(), access the bean and test all above methods.

Assignments 3

Create an interface Sender with a method `void send(double value);`. Implement the method in three bean classes i.e. TcpSender, HttpSender, and UdpSender. In each implementation, write `println()` statement to print the given value on the console. Access these bean objects (e.g. `ts = ctx.getBean(TcpSender.class)`) into main class and call their `send()` methods. What error we get if we access `ctx.getBean(Sender.class)`? Resolve the error using `@Primary` on any one of the bean i.e. `@Primary` with `@Bean` method.

Assignments 4

1. Create an interface Sender with a method `void send(double value);`. Implement the method in three bean classes (using `@Component`) i.e. TcpSender, HttpSender, and UdpSender. In each implementation, write `println()` statement to print the given value on the console. Access these bean objects (e.g. `ts = ctx.getBean(TcpSender.class)`) into main class and call their `send()` methods. What error we get if we access `ctx.getBean(Sender.class)`? Resolve the error using `@Primary` on any one of the bean.

2. In earlier Logger classwork, use `@Autowired` on Logger field in AccountImpl. If you have set logger manually in main code, comment those lines. Comment `@Primary` on bean class (if written). Observe the error and solve it.

3. Implement JDBC Daos i.e. BookDao and CustomerDao for BookShop application using Spring

Assignments 5

1. Complete BookDao and CustomerDao implementations.
2. Create a login page to input email and password. Check if user is valid or not (using CustomerDao). On welcome page display "Welcome" or "Login Failed" message.

Assignments 6

1. Complete Mobiles & Users REST services for Mobile shop.
2. Complete all REST services for Book shop

Assignments 7

1. Create categories table (id, title, description). Example categories are Electronics, Fashion, Grocery. Create products table (id, name, price, category_id). Create appropriate entity mappings. Create a spring rest application to perform crud operations on categories and products.

2. Create Mobile shop REST apis.

Assignments 8

1. Create categories table (id, title, description). Example categories are Electronics, Fashion, Grocery. Create products table (id, name, price, category_id). Create appropriate entity mappings. Create a web mvc application to show all categories. If any category is clicked, on next page display products for that category. Also provide button for creating new product. Create form to add new product. Do validation for new product.
2. Create tables users(id, firstname, lastname, email, password) and tweets(id, text, hashtag, tweet_datetime, user_id). Assume that one tweet will have only one hashtag. Create appropriate entities and repositories. Create appropriate CRUD rest apis.