**Exercise 11: Implementing Dependency Injection**

**Step 1: Repository Interface – CustomerRepository.java**

public interface CustomerRepository {

String findCustomerById(String id);

}

**Step 2: Concrete Repository – CustomerRepositoryImpl.java**

public class CustomerRepositoryImpl implements CustomerRepository {

@Override

public String findCustomerById(String id) {

// In real app, this would query a database

return "Customer[ID=" + id + ", Name=Anushka Gone]";

}

}

**Step 3: Service Class – CustomerService.java**

public class CustomerService {

private CustomerRepository repository;

// Constructor-based Dependency Injection

public CustomerService(CustomerRepository repository) {

this.repository = repository;

}

public void displayCustomer(String customerId) {

String customerDetails = repository.findCustomerById(customerId);

System.out.println("Fetched Customer: " + customerDetails);

}

}

**Step 4: Main Class – TestDependencyInjection.java**

public class TestDependencyInjection {

public static void main(String[] args) {

// Create repository object

CustomerRepository repo = new CustomerRepositoryImpl();

// Inject repository into service

CustomerService service = new CustomerService(repo);

// Use service to display customer

service.displayCustomer("C101");

}

}

**Output:**

A black screen with white text

AI-generated content may be incorrect.