**Exercise 11: Implementing Dependency Injection**

**Scenario:**

You are developing a customer management application where the service class depends on a repository class. Use Dependency Injection to manage these dependencies.

**Steps:**

1. **Create a New Java Project:**
   * Create a new Java project named **DependencyInjectionExample**.
2. **Define Repository Interface:**
   * Create an interface **CustomerRepository** with methods like **findCustomerById()**.
3. **Implement Concrete Repository:**
   * Create a class **CustomerRepositoryImpl** that implements **CustomerRepository**.
4. **Define Service Class:**
   * Create a class **CustomerService** that depends on **CustomerRepository**.
5. **Implement Dependency Injection:**
   * Use constructor injection to inject **CustomerRepository** into **CustomerService**.
6. **Test the Dependency Injection Implementation:**
   * Create a main class to demonstrate creating a **CustomerService** with **CustomerRepositoryImpl** and using it to find a customer.

Code –

package Design\_Patterns\_And\_Principles.DependencyInjectionExample;

import java.util.HashMap;

import java.util.Map;

 interface CustomerRepository {

    String findCustomerById(int id);

}

 class CustomerRepositoryImpl implements CustomerRepository {

    private Map<Integer, String> customerData;

    public CustomerRepositoryImpl() {

        customerData = new HashMap<>();

        customerData.put(1, "John Doe");

        customerData.put(2, "Jane Smith");

        customerData.put(3, "Alice Johnson");

    }

    @Override

    public String findCustomerById(int id) {

        return customerData.get(id);

    }

}

class CustomerService {

    private CustomerRepository customerRepository;

    // Constructor injection

    public CustomerService(CustomerRepository customerRepository) {

        this.customerRepository = customerRepository;

    }

    public String getCustomerName(int id) {

        return customerRepository.findCustomerById(id);

    }

}

public class DependencyInjectionDemo {

    public static void main(String[] args) {

        // Create a CustomerRepositoryImpl instance

        CustomerRepository customerRepository = new CustomerRepositoryImpl();

        // Create a CustomerService instance with the injected repository

        CustomerService customerService = new CustomerService(customerRepository);

        // Test the service

        int customerId = 2;

        String customerName = customerService.getCustomerName(customerId);

        System.out.println("Customer ID " + customerId + " is " + customerName);

    }

}