**Module 1 - Design Patterns and Principles**

**Exercise 11: Implementing Dependency Injection**

**Aim:**

To develope a customer management application where the service class depends on a repository class using Dependency Injection to manage these dependencies.

**Code:**

interface CustomerRepository {

    String findCustomerById(String id);

}

class CustomerRepositoryImpl implements CustomerRepository {

    public String findCustomerById(String id) {

        return "Customer ID: " + id + " | Name: Deepak | Status: Active";

    }

}

class CustomerService {

    private CustomerRepository repo;

    public CustomerService(CustomerRepository repo) {

        this.repo = repo;

    }

    public void displayCustomer(String id) {

        System.out.println(repo.findCustomerById(id));

    }

}

public class DependencyInjectionExample{

    public static void main(String[] args) {

        CustomerRepository repo = new CustomerRepositoryImpl();

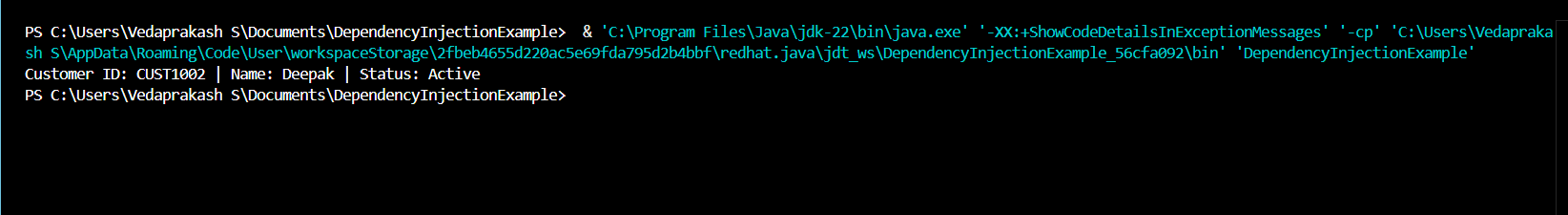
        CustomerService service = new CustomerService(repo);

        service.displayCustomer("CUST1002");

    }

}

**Output:**

****