**Exercise 4.5: Mocking Multiple Return Values**

**CODE:**

**Repository Interface: -**

package org.example;  
public interface Repository{  
 String getData();  
}

**Service Class: -**

package org.example;  
public class Service{  
 private Repository repository;  
  
 public Service(Repository repository) {  
 this.repository = repository;  
 }  
  
 public String processData() {  
 String data = repository.getData();  
 System.*out*.println("Fetched: " + data);  
 return "Processed " + data;  
 }  
}

**ServiceTest Class: -**

package org.example;  
import org.junit.jupiter.api.Test;  
import static org.junit.jupiter.api.Assertions.\*;  
import static org.mockito.Mockito.*mock*;  
import static org.mockito.Mockito.*when*;  
  
class ServiceTest {  
 @Test  
 public void testServiceWithMultipleReturnValues() {  
 Repository mockRepository=*mock*(Repository.class);  
 *when*(mockRepository.getData())  
 .thenReturn("First Mock Data")  
 .thenReturn("Second Mock Data");  
 Service service=new Service(mockRepository);  
 String firstResult=service.processData();  
 String secondResult=service.processData();  
 System.*out*.println("First: "+firstResult);  
 System.*out*.println("Second: "+secondResult);  
 *assertEquals*("Processed First Mock Data",firstResult);  
 *assertEquals*("Processed Second Mock Data",secondResult);  
 }  
}

**OUTPUT:**

