**Exercise 4.1: Mocking Databases and Repositories**

**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("Repository returned: "+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 testServiceWithMockRepository(){  
 Repository mockRepository=*mock*(Repository.class);  
 *when*(mockRepository.getData()).thenReturn("Mock Data");  
 Service service=new Service(mockRepository);  
 String result=service.processData();  
 System.*out*.println("Final result from service: "+result);  
 *assertEquals*("Processed Mock Data",result);  
 }  
}

**OUTPUT:**

