**Exercise 2: Mocking a Repository in a Service Test**

**\*\*Task:\*\*** Write a unit test for a Spring service that uses a repository to fetch data. Mock the repository dependency using Mockito.

**User.java :-**

package org.example1;  
  
import jakarta.persistence.Entity;  
import jakarta.persistence.Id;  
  
@Entity  
public class User {  
  
 @Id  
 private Long id;  
 private String name;  
  
 public User() {}  
  
 public User(Long id, String name) {  
 this.id = id;  
 this.name = name;  
 }  
  
 // Getters and setters  
 public Long getId() {  
 return id;  
 }  
  
 public void setId(Long id) {  
 this.id = id;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
}

**UserRepository.java :-**

package org.example1;  
  
import org.springframework.data.jpa.repository.JpaRepository;  
  
public interface UserRepository extends JpaRepository<User, Long> {  
}

**UService.java :-**

package org.example1;  
  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.stereotype.Service;  
  
@Service  
public class UserService {  
  
 @Autowired  
 private UserRepository userRepository;  
  
 public User getUserById(Long id) {  
 return userRepository.findById(id).orElse(null);  
 }  
}

**UserServiceTest.java :-**

package org.example1;  
  
import org.junit.jupiter.api.Test;  
import org.mockito.InjectMocks;  
import org.mockito.Mock;  
import org.mockito.MockitoAnnotations;  
  
import java.util.Optional;  
  
import static org.junit.jupiter.api.Assertions.*assertEquals*;  
import static org.mockito.Mockito.*when*;  
  
public class UserServiceTest {  
  
 @Mock  
 private UserRepository userRepository;  
  
 @InjectMocks  
 private UserService userService;  
  
 public UserServiceTest() {  
 MockitoAnnotations.*openMocks*(this);  
 }  
  
 @Test  
 void testGetUserById() {  
 User user = new User(1L, "Naresh");  
 *when*(userRepository.findById(1L)).thenReturn(Optional.*of*(user));  
  
 User result = userService.getUserById(1L);  
 *assertEquals*("Naresh", result.getName());  
 }  
}

**Output :-**

