**Exercise 6.1:** **Mocking a Service Dependency in a Controller Test**

**CODE:**

**User Class(Entity): -**

package com.example61;  
  
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;  
 }  
  
 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;  
 }  
}

**UserRespository Interface: -**

package com.example61;  
  
import org.springframework.data.jpa.repository.JpaRepository;  
  
public interface UserRepository extends JpaRepository<User,Long> {  
}

**UserService Class: -**

package com.example61;  
  
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);  
 }  
}

**UserController Class: -**

package com.example61;  
  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.http.ResponseEntity;  
import org.springframework.web.bind.annotation.GetMapping;  
import org.springframework.web.bind.annotation.PathVariable;  
import org.springframework.web.bind.annotation.RequestMapping;  
import org.springframework.web.bind.annotation.RestController;  
  
@RestController  
@RequestMapping("/users")  
public class UserController {  
  
 @Autowired  
 private UserService userService;  
  
 @GetMapping("/{id}")  
 public ResponseEntity<User> getUser(@PathVariable Long id) {  
 return ResponseEntity.*ok*(userService.getUserById(id));  
 }  
}

**UserTest Class: -**

package com.example61;  
  
import org.junit.jupiter.api.Test;  
import org.mockito.Mockito;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.boot.test.autoconfigure.web.servlet.WebMvcTest;  
import org.springframework.boot.test.mock.mockito.MockBean;  
import org.springframework.test.web.servlet.MockMvc;  
  
import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.*get*;  
import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.*jsonPath*;  
import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.*status*;  
  
@WebMvcTest(UserController.class)  
public class UserControllerTest {  
  
 @Autowired  
 private MockMvc mockMvc;  
  
 @MockBean  
 private UserService userService;  
  
 @Test  
 public void testGetUserById() throws Exception {  
 User user = new User(1L, "Srija");  
  
 Mockito.*when*(userService.getUserById(1L)).thenReturn(user);  
  
 mockMvc.perform(*get*("/users/1"))  
 .andExpect(*status*().isOk())  
 .andExpect(*jsonPath*("$.id").value(1L))  
 .andExpect(*jsonPath*("$.name").value("Srija"));  
 }  
}

**OUTPUT:**

