**Exercise 5.5: Test Controller POST Endpoint**

**CODE:**

**User Class(Entity): -**

package org.example;  
  
import jakarta.persistence.Entity;  
  
@Entity  
public class User {  
 private Long Id;  
 private String name;  
 private String email;  
  
 public User() {  
 }  
  
 public User(Long id, String name, String email) {  
 Id = id;  
 this.name = name;  
 this.email = email;  
 }  
  
 public Long getId() {  
 return Id;  
 }  
  
 public void setId(Long id) {  
 Id = id;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
  
 public String getEmail() {  
 return email;  
 }  
  
 public void setEmail(String email) {  
 this.email = email;  
 }  
}

**UserRepository Interface: -**

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

**UserService Class: -**

package org.example;  
import org.springframework.stereotype.Service;  
  
@Service  
public class UserService {  
  
 private UserRepository userRepository;  
  
 public UserService(UserRepository userRepository) {  
 this.userRepository = userRepository;  
 }  
 public User saveUser(User user){  
 return userRepository.save(user);  
 }  
}

**UserController Class: -**

package org.example;  
  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.http.ResponseEntity;  
import org.springframework.web.bind.annotation.PostMapping;  
import org.springframework.web.bind.annotation.RequestBody;  
import org.springframework.web.bind.annotation.RequestMapping;  
import org.springframework.web.bind.annotation.RestController;  
  
@RestController  
@RequestMapping("/users")  
public class UserController {  
 @Autowired  
 private UserService userService;  
 @PostMapping  
 public ResponseEntity<User> createUser(@RequestBody User user) {  
 return ResponseEntity.*ok*(userService.saveUser(user));  
 }  
}

**UserControllerTest Class: -**

package org.example;  
  
import com.fasterxml.jackson.databind.ObjectMapper;  
import org.junit.jupiter.api.Test;  
import org.junit.jupiter.api.extension.ExtendWith;  
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.http.MediaType;  
import org.springframework.test.context.junit.jupiter.SpringExtension;  
import org.springframework.test.web.servlet.MockMvc;  
  
import static org.mockito.ArgumentMatchers.*any*;  
import static org.mockito.Mockito.*when*;  
import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.*post*;  
import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.\*;  
  
@ExtendWith(SpringExtension.class)  
@WebMvcTest(UserController.class)  
public class UserControllerTest {  
  
 @Autowired  
 private MockMvc mockMvc;  
  
 @MockBean  
 private UserService userService;  
  
 @Test  
 public void testCreateUser() throws Exception {  
  
 User inputUser = new User(null, "Sushma", "sushma@example.com");  
 User savedUser = new User(1L, "Sushma", "sushma@example.com");  
  
 *when*(userService.saveUser(*any*(User.class))).thenReturn(savedUser);  
  
 mockMvc.perform(*post*("/users")  
 .contentType(MediaType.*APPLICATION\_JSON*)  
 .content(new ObjectMapper().writeValueAsString(inputUser)))  
 .andExpect(*status*().isOk())  
 .andExpect(*jsonPath*("$.id").value(1L))  
 .andExpect(*jsonPath*("$.name").value("Sushma"))  
 .andExpect(*jsonPath*("$.email").value("sushma@example.com"));  
  
 System.*out*.println("testCreateUser passed.");  
 }  
}

**OUTPUT:**

