**Exercise 5: Test Controller POST Endpoint**

**Controller.java :-**

@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.java :-**

package com.example.demo;

import com.example.demo.controller.UserController;

import com.example.demo.model.User;

import com.example.demo.service.UserService;

import com.fasterxml.jackson.databind.ObjectMapper;

import org.junit.jupiter.api.Test;

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.web.servlet.MockMvc;

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.\*;

@WebMvcTest(UserController.class)

public class UserControllerTest {

@Autowired

private MockMvc mockMvc;

@MockBean

private UserService userService;

@Autowired

private ObjectMapper objectMapper;

@Test

public void testCreateUser() throws Exception {

// Prepare input and mock output

User inputUser = new User();

inputUser.setName("Naresh");

inputUser.setEmail("naresh@example.com");

User savedUser = new User();

savedUser.setId(1L);

savedUser.setName("Naresh");

savedUser.setEmail("naresh@example.com");

// Mock the service call

when(userService.saveUser(inputUser)).thenReturn(savedUser);

// Perform POST /users

mockMvc.perform(post("/users")

.contentType(MediaType.APPLICATION\_JSON)

.content(objectMapper.writeValueAsString(inputUser)))

.andExpect(status().isOk())

.andExpect(jsonPath("$.id").value(1))

.andExpect(jsonPath("$.name").value("Naresh"))

.andExpect(jsonPath("$.email").value("naresh@example.com"));

}

}

Output :-

