**Exercise 5.7: Test Custom Repository Query**

**CODE:**

**User Class (Entity): -**

package org.example;  
import jakarta.persistence.\*;  
  
@Entity  
@Table(name = "Users")  
public class User {  
 @Id  
 @GeneratedValue(strategy = GenerationType.*IDENTITY*)  
 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;  
import java.util.List;  
  
public interface UserRepository extends JpaRepository<User,Long>{  
 List<User> findByName(String name);  
}

**UserRepositoryTest Class: -**

package org.example;  
import org.junit.jupiter.api.Test;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.boot.test.context.SpringBootTest;  
import org.springframework.test.context.TestPropertySource;  
import java.util.List;  
import static org.junit.jupiter.api.Assertions.*assertEquals*;  
import static org.junit.jupiter.api.Assertions.*assertTrue*;  
  
@SpringBootTest  
@TestPropertySource(locations = "classpath:application-test.properties")  
public class UserRepositoryTest {  
  
 @Autowired  
 private UserRepository userRepository;  
  
 @Test  
 public void testFindByName() {  
 User user1=new User(null, "Sushma", "sushma1@example.com");  
 User user2=new User(null, "Sushma", "sushma2@example.com");  
 User user3=new User(null, "Ravi", "ravi@example.com");  
  
 userRepository.save(user1);  
 userRepository.save(user2);  
 userRepository.save(user3);  
  
 List<User> result = userRepository.findByName("Sushma");  
  
 *assertEquals*(2, result.size());  
 *assertTrue*(result.stream().allMatch(user -> user.getName().equals("Sushma")));  
  
 System.*out*.println("testFindByName passed: Found " + result.size() + " users named Sushma.");  
 }  
}

**OUTPUT:**

