**Exercise 7: Handling Void Methods with Exceptions**

**Scenario:** You need to test a void method that throws an exception.

**NotificationService.java :-**

package org.example1;  
  
public interface NotificationService {  
 void sendEmail(String recipient);  
}

**NotificationServiceTest.java :-**

package org.example1;  
  
import org.junit.jupiter.api.Test;  
  
import static org.mockito.Mockito.\*;  
import static org.junit.jupiter.api.Assertions.*assertThrows*;  
  
public class NotificationServiceTest {  
  
 @Test  
 public void testSendEmail\_ThrowsException() {  
 // Step 1: Create mock object  
 NotificationService mockService = *mock*(NotificationService.class);  
  
 // Step 2: Stub the void method to throw an exception  
 *doThrow*(new RuntimeException("Mail server down"))  
 .when(mockService).sendEmail("naresh@example.com");  
  
 // Step 3: Verify that the exception is thrown  
 *assertThrows*(RuntimeException.class, () -> {  
 mockService.sendEmail("naresh@example.com");  
 });  
  
 // Step 4: Verify interaction  
 *verify*(mockService).sendEmail("naresh@example.com");  
 }  
}

**pom.xml :-**

<project xmlns="http://maven.apache.org/POM/4.0.0"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0  
 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
 <groupId>org.example1</groupId>  
 <artifactId>void-method-exception</artifactId>  
 <version>1.0-SNAPSHOT</version>  
  
 <properties>  
 <maven.compiler.source>17</maven.compiler.source>  
 <maven.compiler.target>17</maven.compiler.target>  
 </properties>  
  
 <dependencies>  
 <!-- JUnit 5 -->  
 <dependency>  
 <groupId>org.junit.jupiter</groupId>  
 <artifactId>junit-jupiter</artifactId>  
 <version>5.10.0</version>  
 <scope>test</scope>  
 </dependency>  
  
 <!-- Mockito -->  
 <dependency>  
 <groupId>org.mockito</groupId>  
 <artifactId>mockito-core</artifactId>  
 <version>5.11.0</version>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>junit</groupId>  
 <artifactId>junit</artifactId>  
 <version>4.13.2</version>  
 <scope>test</scope>  
 </dependency>  
 </dependencies>  
  
 <build>  
 <plugins>  
 <plugin>  
 <groupId>org.apache.maven.plugins</groupId>  
 <artifactId>maven-surefire-plugin</artifactId>  
 <version>3.1.2</version>  
 </plugin>  
 </plugins>  
 </build>  
</project>

**Output :-**

