Test Cases for Practice

1. Write a Java unit test case to assert that a given method returns the expected value.

2. Write a Java test case to verify that a specific exception is thrown when a method is called with invalid input.

3. Write a Java test case that implements a JUnit test with setup and teardown methods to prepare and clean up resources for multiple test cases.

4. Write a Java program that implements parameterized test to verify that a method behaves correctly for different input values.

5. Write a Java test case that fails if a method takes longer than a specified time to execute.

6. Write a Java test case that is intentionally ignored, and explain the scenarios where test skipping might be useful.

7. Write a Java program that uses assertions with custom error messages to provide meaningful feedback when a test fails.

8. Write a Java program to explore strategies for testing private methods in a class.

9. Write a Java program that tests a singleton class, ensuring it behaves as expected in a multi-threaded environment.

10. Write a Java program that create tests that verify the interaction between different components or modules in your application.