



# Test.isRunningTest() Method in Apex

# What is Test.isRunningTest() ?

When we are setting up test data from Apex test method, we need a way to disable the triggers that will fire. It might cause "LimitException: 'Too many SOQL queries: 101'". In this case, the triggers are not the target of the test case, hence this scenario will cause the test method to fail. It is not necessary to disable the trigger for every test case, instead we can use isRunningTest().

# What is Test.isRunningTest() ?

Test.isRunningTest() is a method in Salesforce Apex that allows you to determine whether the current code is being executed as part of a test method or not. It returns a Boolean value (true or false) indicating if the code is running within the context of a test class.



This method is useful when certain logic or functionality should only be executed (or skipped) during test execution. It allows you to handle situations where you may need to bypass certain operations like callouts, data-heavy operations, or external service integrations when running tests.

```
public class LongRunningService {  
    public void processData() {  
        if (!Test.isRunningTest()) {  
            // Only execute this complex logic in non-test scenarios  
            System.debug('Performing a long-running data processing operation...');  
  
            // Simulating a long operation (e.g., API call, heavy calculations, etc.)  
            Integer sum = 0;  
            for (Integer i = 0; i < 1000000; i++) {  
                sum += i;  
            }  
            System.debug('Operation complete. Sum: ' + sum);  
        } else {  
            // Bypassing the logic for test execution  
            System.debug('This is a test run, skipping long-running process...');  
        }  
    }  
}
```