# TEST 1

## Error conditions handled smoothly

In most cases tests should include confirmation that the macro handle invalid or error conditions smoothly, with a clear message to the user, and an appropriate return value to the calling program.

For example, given an invalid data set name, a macro should set the global macro variable CONTINUE to 0 (See [Project Programming Guidelines](http://www.phusewiki.org/wiki/index.php?title=WG5_P02_Programming_Guidelines) for further discussion).

## Checks

1. MISSING required parameter, **return 0**
   1. Missing DS
   2. Missing VAR
   3. Missing DS and VAR
2. repeat a. with INVALID required parameter, **return 0**
3. Invalid optional WHR= clause, **return 0**
   1. invalid variable specified
   2. variable type mismatch in WHR clause, using NUMERIC var
   3. variable type mismatch in WHR clause, using CHAR var

# TEST 2 – not yet implemented

## Title of this macro-specific test

Brief description and justification of this test.

For example, illustrate the need for this test using an example from clinical trials analysis.

## Checks

1. Quick description of this scenario
   1. Describe a successful result
   2. Or describe an expected failure result

# TEST 3 – not yet implemented

## Confirm expected log messages

In most cases tests will include failure scenarios, which should produce log messages. The following log messages are EXPECTED due to test design (testing failure conditions):

WARNING: (TEST\_*macro-name*) User must ensure PhUSE/CSS utilities are in the AUTOCALL path.

ERROR: (*macro-name*) Result is FAIL. Here are a few helpful details.

ERROR: (*macro-name*) Result is FAIL. Something else went wrong here.

ERROR: Errors printed on pages 12,15,20.