Macros – the Language of SAS

Please read our Terms and Conditions and our Privacy Policy.

Duration: 14-16 hours (plus additional time for exercises)

Learning Overview:

Do you run your programs more than once? The Macros course teaches you techniques to make your programs smarter and easier to maintain.

There are mini quizzes and comprehensive exercises throughout to help assess and reinforce your learning.

Learning Outcomes:

By the end of this course you will be able to:

1. Explain how base and macro code is tokenised and processed
2. Describe how macro variables are stored and resolved
3. Utilise macro variables and functions to make your programs smarter
4. Create and store macro programs to simplify repetitive tasks
5. Implement conditional iterative macro processing to help you generate dynamic, data driven code in the Language of SAS

Delivery Schedule:

Week 1

On day 1 we invite join a short online welcome to meet your instructor, introduce you to our Digital Learning platform and get you started on the eLearning modules of this course.

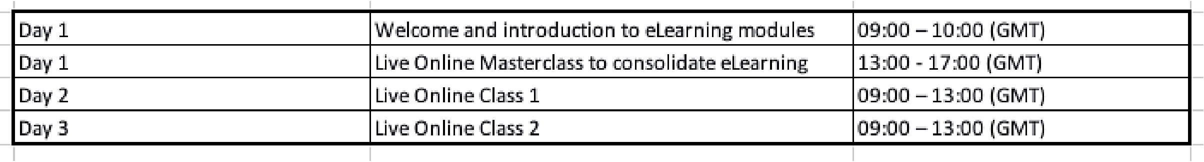
You will need 4 hours to complete the 6 eLearning modules with additional time required for exercises.

We ask you to complete these modules before joining us for the online masterclass on day 2.

This is followed by 3 x 4 hour live online classes to complete your learning.

In addition, you will also need to set aside time to complete the Hands-On exercises.

Each day finishes with an opportunity for you to join an open Q&A session with the instructor if you need any additional support.



Day 1	Welcome and introduction to eLearning modules	09:00 – 10:00 (GMT)
Day 1	Live Online Masterclass to consolidate eLearning	13:00 - 17:00 (GMT)
Day 2	Live Online Class 1	09:00 – 13:00 (GMT)
Day 3	Live Online Class 2	09:00 – 13:00 (GMT)

Pre-requisites:

To get the most out of this course it is expected that you should have attended the Fundamentals – the Language of SAS and the Intermediate – the Language of SAS courses (or comparable courses of study).

If not, we would advise that you have at least six months experience of developing code in the language of SAS, including:

- Use of Global statements e.g. LIBNAME FILENAME and OPTIONS
- Use of functions to transform character data e.g. SCAN, FIND and SUBSTR
- Use of functions to transform numeric data e.g. CEIL, INT, FLOOR and ROUND
- Use of functions to transform dates e.g. DAY, MONTH, YEAR and INTCK
- KEEP and DROP to select variables
- IF and WHERE statements to select observations
- Conditional Processing with IF-THEN-ELSE
- Iterative DO loops
- Conditional DO loops e.g. DO WHILE and DO UNTIL
- Analyse data with the FREQ, MEANS and TABULATE procedures
- Combine data horizontally and vertically using MERGE and SET
- Summarise data using BY Groups
- Understand the two phases of DATA Step processing

For the hands-on practice activities in the course, you will need access to an environment that runs the programming Language of SAS. On our courses, we signpost you to some of the free tools available.

Check out the link below to review system requirements:

- SAS® OnDemand for Academics

Learning Modules:

Introduction to Macro processing (eLearning)

Learning Objective: Explain basic Macro Processing in the Language of SAS.

- Compilation and Execution phases
- Tokenisation
- Macro Triggers
- The %PUT Statement
- Displaying macro variable values

Introduction to Macro variables (eLearning)

Learning Objective: Describe six different features of Macro Variables.

- Macro Variable Storage
- Macro Variable Resolution
- User-Defined Macro Variables
- The %LET Statement
- The SYMBOLGEN Option
- Delimiting Macro Variables

Macro Functions (eLearning)

Learning Objective: Identify three different types of Macro Functions.

- Character Functions – review
- Using Macro Functions
- Macro Functions for Numbers
- The %SYSFUNC and other functions including: %INDEX, %SCAN, %SUBSTR, %UPCASE, %BQUOTE, %STR, %NRBQUOTE, %NRSTR, %EVAL and % SYSEVALF.

Macro Programs (live online class)

Learning Objective: *Create and store macro programs to simplify repetitive tasks*

- Compilation
- Macro Calls
- Generating Code
- The MPRINT Option
- Storage
- Autocall Macros
- Best Practices
- Macro Program Parameters

Global and Local Symbol Tables (live online class)

Learning Objective: Utilise Macro Variables on the Global and Local Symbol Tables

- The Global Symbol Table
- The Local Symbol Table
- Nested Macros
- Creating and Updating Macro Variables
- Resolving Macro Variables

DATA Step and PROC SQL (live online class)

Learning Objective: Creating Macro Variables in DATA Step and Proc SQL

- Creating Macro Variables in DATA Step
- Compare CALL SYMPUTX and %LET
- Indirect Macro Variable References
- Create Macro Variables in PROC SQL
- Store multiple values in a Macro Variable

Conditional Processing (live online class)

Learning Objective: Implement Conditional Macro Processing

- %IF -%THEN - %ELSE
- The MLOGIC Option
- Conditionally Returning a Program Step
- Conditionally Returning a Statement
- Conditionally Returning a Partial Statement
- %DO - %END
- The &INCLUDE Statement

Iterative Processing (live online class)

*Learning Objective: G*enerate dynamic, data driven code - using Conditional Iterative Processing

- Iterative Processing in a Macro Program
- Generate Program Code Iteratively – Dynamic Code
- Perform Conditional Iterative Processing