

SAS

Programming

Course Content



SAS Programming

SAS Base

- Use FORMATTED, LIST and COLUMN input to read raw data files
- Use INFILE statement options to control processing when reading raw data files
- Use various components of an INPUT statement to process raw data files including column and line pointer controls, and trailing @ controls
- Combine SAS data sets using the DATA step

Creating Data Structures

- Create temporary and permanent SAS data sets
- Create and manipulate SAS date values
- Use DATA Step statements to export data to standard and comma delimited raw data files
- Control which observations and variables in a SAS data set are processed and output

Managing Data

- Investigate SAS data libraries using base SAS utility procedures
- Sort observations in a SAS data set
- Conditionally execute SAS statements
- Use assignment statements in the DATA step
- Modify variable attributes using options and statements in the DATA step
- Accumulate sub-totals and totals using DATA step statements
- Use SAS functions to manipulate character data, numeric data, and SAS date values
- Use SAS functions to convert character data to numeric and vice versa
- Process data using DO LOOPS

-
- Process data using SAS arrays

Generating Reports

- Generate list reports using the PRINT and REPORT procedures
- Generate summary reports and frequency tables using base SAS procedures
- Enhance reports through the use of labels, SAS formats, user-defined formats, titles, footnotes and SAS System reporting options
- Generate HTML reports using ODS statements

Handling Errors

- Identify and resolve programming logic errors
- Recognize and correct syntax errors
- Examine and resolve data errors

SAS Advanced Programming

Accessing Data Using SQL

- Generate detail reports by working with a single table or joining tables using PROC SQL and the appropriate options
- Generate summary reports by working with a single table or joining tables using PROC SQL and the appropriate options
- Construct sub queries within a PROC SQL step
- Compare solving a problem using the SQL procedure versus using traditional SAS programming techniques
- Access Dictionary Tables using the SQL procedure
- Demonstrate advanced PROC SQL skills by creating and updating tables, updating data values, working with indexes using the macro interface/creating macro variables with SQL, defining integrity constraints, SQL views and SET operators

Macro Processing

- Creating and using user-defined and automatic macro variables within the SAS Macro Language
- Automate programs by defining and calling macros using the SAS Macro Language
- Understand the use of macro functions
- Recognize various system options that are available for macro debugging and displaying values of user-defined and automatic macro variables in the SAS log

Advanced Programming Techniques

- Demonstrate advanced data set processing techniques such as updating master data sets, transposing data, combining/merging data, sampling data using generation data sets, integrity constraints and audit trails
- Reduce the space required to store SAS data sets and numeric variables within SAS data sets by using compression techniques, length statements or DATA step views
- Develop efficient programs by using advanced programming techniques such as permanent formats and array processing
- Use SAS System options and SAS data set options for controlling memory usage
- Control the processing of variables and observations in the DATA step
- Create sorted or indexed data in order to avoid unnecessary sorts, eliminate duplicate data and to provide more efficient data access and retrieval
- Use PROC DATASETS to demonstrate advanced programming skills (e.g. renaming columns, displaying metadata, creating indexes, creating integrity constraints, creating audit trails)



Noida Office

**Webtrackker Technology (P)Ltd
c-67 Sector 63 Noida,India**

Other Office

**F -1 Sector 3 (Near Sector
16 metro station)Noida, India.**