

# Getting Started with Clinical SAS Programming Self-Study Course on [SASSavvy.com](http://SASSavvy.com)

From: Sunil Gupta, [Sunil@GuptaProgramming.com](mailto:Sunil@GuptaProgramming.com)

## EXECUTIVE SUMMARY

For pharmaceutical companies, there is a push to prepare SAS programs for clinical study reporting, analysis and validation. Clinical, statistical and regulatory professionals need to save time and resources by learning how to do things right the first time instead of just doing something. The variety of topics covered in this course use real-world examples to provide solutions based on a combination of processes and techniques. Students can expect to directly apply methods learned to fulfill reporting needs as well as address validation requirements. Students will benefit from my project management and hands-on experience of over seven successful FDA submissions.

This unique and comprehensive course will serve as a good starting point to understand the many programming options available to apply SAS software in regulatory submissions. This course complements the current list of courses in the regulatory, validation and technology sections.

## AUDIENCE

This course is ideal for pharmaceutical/clinical trial professionals including SAS Statistical Programmers, Statisticians, Clinical Data Managers, Clinical Research Analysts, Medical Affairs Personnel, Quality Assurance Specialists and Regulatory Affairs associates. Anyone directly or indirectly responsible for the creation, content or validation of summary tables, data lists and graphs used to support drug efficacy and safety in a regulatory submission will benefit from this course.

This course requires some SAS programming experience and an understanding of the clinical trials process and the regulatory environment.

## COURSE DESCRIPTION

This course provides a good overview of the processes and techniques required by SAS Statistical Programmers to meet regulatory submission deadlines. This in-depth course shows SAS programming examples for developing a clinical reporting system to create publication-quality summary tables, data lists, and graphs as well incorporating validation steps throughout the process. Get your SAS technical and validation questions answered and learn efficient tips for producing a quality regulatory submission.

## COURSE CONTENT

### A – Understanding the environment and setting the stage

- Chapter 1: Regulatory Environment: Requirements, Standards and Clinical Data
  - Section 1: US Code of Federal Regulations (CFR) Title 21 Part 11 requirements
  - Section 2: Creating the required documentations for effective impact

- Section 3: FDA expectations and QA Audits
- Section 4: Clinical data flow and structure
- Chapter 2: Overview of Regulatory Processes and SAS Techniques
  - Section 1: About this Course
  - Section 2: Process for creating and validating output
  - Section 3: Various techniques to address requirements
  - Section 4: Sample Clinical Study – Summary Table, Data Lists, Graph
  - Section 5: Analysis of potential set backs
- Chapter 3: Understanding and Applying the QC Plan
  - Section 1: Differences between Quality Control and Reviewer/Quality Assurance
  - Section 2: Applying methods to validate programs from QC Plan
  - Section 3: Completing tasks with Validation Documentation

## **B - Producing Summary Tables, Data Lists and Graphs**

- Chapter 4: Programming standards, conventions and training for improved productivity
  - Section 1: Program Index and Directory Structure for better organization
  - Section 2: Software Development Life Cycle (SDLC) for accurate, reliable, and validated results
  - Section 3: Information in the Program Header
  - Section 4: Anatomy of a SAS Application Program
- Chapter 5: Creating Summary Tables, Data Lists, and Graphs
  - Section 1: Developing Summary Tables
  - Section 2: Developing Data Lists
  - Section 3: Developing Graphs
  - Section 4: Clinical Trials Reporting Templates
- Chapter 6. Identifying data-related issues using edit checks and validation macros
  - Section 1: Focus on generating output instead of writing SAS code
  - Section 2: Easier to read SAS code that would traditionally be lengthy
  - Section 3: Power and flexibility of Proc SQL for queries and validation
  - Section 4: Data integrity with edit checks as PDF file
  - Section 5: Automate communication of updated data sets and output files

## **C- Appendix**

- A. Handling Errors and Debugging Programs
  - Section 1: Recognizing and correcting syntax and non-syntax errors
  - Section 2: Examining and resolving data errors
  - Section 3: Program debugging techniques
- B. Enterprise Guide Basics
- C. Style Definitions and Table Templates using Proc Template