

# Good Programming Practice

## Table of Contents

<i>Good Programming Practice .....</i>	<i>1</i>
Introduction.....	2
Header and Revision History.....	3

## Introduction

The Good Programming Practices are defined in order to:

- Ensure the clarity of the code and facilitate code review.
- Save time in case of maintenance, and ease the transfer of code among programmers or companies.
- Minimise the need for code maintenance by robust programming.
- Minimise the development effort by the development and re-use of standard code and by the use of dynamic (easily adaptable) code;
- Minimise the resources needed at execution time (improve the efficiency of the code);
- Reduce the risk of logical errors.
- Meet regulatory requirements regarding validation and 21CFRPart11 compliance.

Note: As often, the various guidelines provided hereafter may conflict with one another if applied in too rigorous a way. Clarity, efficiency, re-usability, adaptability and robustness of the code are all important, and must be balanced in the programming practice.

Regulatory Requirements	
Validation	Reading the <a href="#">21CFRPart11</a> actual documentation is helpful. While it is long and written in government speak, you can read it and interpret it for yourself. <a href="#">21 CFR Part 11</a>

Readability and Maintainability	
Language	English is an international language and study protocols, study reports for practical reasons (regulatory authorities, inlicensing, outlicensing, partnerships, mergers) are mostly written in English, therefore it is recommended to write the SAS code and comments in English.

## Header and Revision History

Include a header for every program (template below)

```
*****,  
* Program name   :  
*  
* Author        :  
*  
* Date created   :  
*  
* Study          : (Study number)  
*                (Study title)  
*  
* Purpose        :  
*  
* Template       :  
*  
* Inputs         :  
*  
* Outputs        :  
*  
* Program completed: Yes/No  
*  
* Updated by     : (Name) – (Date):  
*                (Modification and Reason)  
*****,
```

- In addition to your name or initials, use your login ID to identify yourself in the header. This is so there is no ambiguity on the identity of each programmer.
- Update the revision history at each code modification made after the finalisation of the first version of a program.

Note: When you copy a program from another study, you become the author of this program, and you should clear the revision history. You can specify the origin of the program under the “Template” section of the header. Text is available under the [Creative Commons Attribution-ShareAlike License](#) ; additional terms may apply. See Terms of Use for details.