# Introduction to Coding Conventions

SAS FANS Sweden

Network Meeting – Programming, 2021-02-26

Erik Strömgren, SAS

# Coding conventions

From Wikipedia, the free encyclopedia

**Coding conventions** are a set of guidelines for a specific programming langua that recommend programming style, practices, and methods for each aspect a program written in that language. These conventions usually cover file organization, indentation, comments, declarations, statements, white space, naming conventions, programming practices, programming principles, programming rules of thumb, architectural best practices, etc. These are guidelines for software structural quality. Software programmers are highly recommended to follow these guidelines to help improve the readability of t source code and make software maintenance easier. Coding conventions are only applicable to the human maintainers and peer reviewers of a software project. Conventions may be formalized in a documented set of rules that an entire team or company follows,[1] or may be as informal as the habitual codi practices of an individual. Coding conventions are not enforced by compilers.

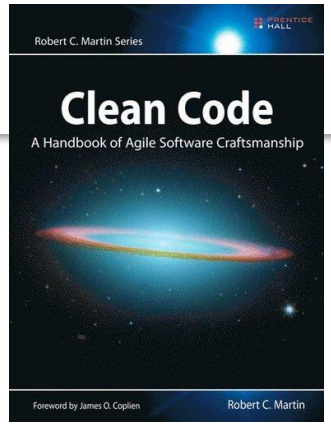


SAS…?

From Wikipedia, the free encyclopedia

**Refactoring**

Refactoring refers to a software maintenance activity where source code is modified to improve readability or improve its structure. Software is often refactored to bring it into conformance with a team's stated coding standards after its initial release. Any change that does not alter the behavior of the software can be considered refactoring. Common refactoring activities are changing variable names, renaming methods, moving methods or whole classes and breaking large methods (or functions) into smaller ones.


Agile software development methodologies plan for regular (or even continuous) refactoring making it an integral part of the team software development process.

From Wikipedia, the free encyclopedia

There are a large number of coding conventions; see *Coding Style* for numerous examples and discussion. Common coding conventions may cover the following areas:

- Comment conventions
- Indent style conventions
- Line length conventions
- Naming conventions
- Programming practices
- Programming principles
- Programming style conventions

Coding standards include the CERT C Coding Standard, MISRA C, High Integrity C++.

# Naming conventions

- Length of identifiers
  - `ap, avgprc, averageprice, productaverageprice`
- Letter case and numerals
  - `cat2, CAT2, Category2, _2Cat`
- Multiple-word identifiers
  - `averagePrice, AveragePrice, average_price`

| Multiple-word identifier formats | |
|---|---|
| Formatting | Name(s) |
| twowords | flat case[12][13] |
| TWOWORDS | upper flat case[12] |
| twoWords | (lower) camelCase, dromedaryCase |
| twoWords | camelCase |
| TwoWords | PascalCase,Upper Camel Case,StudlyCase[14] |
| two_words | snake_case, pothole_case |
| TWO_WORDS | SCREAMING_SNAKE_CASE, MACRO_CASE, CONSTANT_CASE |
| two_Words | camel_Snake_Case |
| Two_Words | Pascal_Snake_Case |
| two-words | kebab-case, dash-case, lisp-case |
| two\|words | doner\|case |
| TWO-WORDS | TRAIN-CASE, COBOL-CASE, SCREAMING-KEBAB-CASE |
| Two-Words | Train-Case,[12] HTTP-Header-Case[15] |

# Poll Q1: How do you feel about a coding convention in your team for SAS code?

- Don’t care. As long as the code works… Move along!
- Nah… I like my own code style… no need for others to have opinions…
- Hm… would be interesting to see how others code…
- We should definitely talk about it more in my team…
- Very important! Critical for my organization!
- <none of these options>

# Poll Q2: Does your team have an agreed upon (documented or not) coding convention for writing new SAS code?

- No. I don’t have a team. I’m the only one working with my SAS code.
- No.
- Yes. But no one follows it.
- Yes. But only some of us follow it.
- Yes. And most actually follow it.
- Yes. And our code review process checks it is actually followed.
- <none of these options>

# Poll Q3: When writing new SAS code, how would you name a dataset with employee status codes?

- `data work.empsttscds;`
- `data work.EmpSttsCds;`
- `data work.EMPSTTSCDS;`
- `data work.employeestatuscodes;`
- `data work.EmployeeStatusCodes;`
- `data work.EMPLOYEESTATUSCODES;`
- `data work.employee_status_codes;`
- `data work.EMPLOYEE_STATUS_CODES;`
- <not like any of these>

# Poll Q4: When writing new SAS code, how would you, in an employee dataset, name a variable containing employee annual salary?

- `sal = source.MonthlySalary * 12;`
- `SAL`
- `Salary`
- `annualsal`
- `AnnualSalary`
- `EmployeeAnnualSalary`
- `EMPLOYEEANNUALSALARY`
- `employee_annual_salary`
- `EMPLOYEE_ANNUAL_SALARY`
- <not like any of these>

# Poll Q5: When writing new SAS code, how would you name a macro variable containing an employee type code?

- `%let type =C3;`
- `%let TYPE`
- `%let Type`
- `%let typecode`
- `%let TypeCode`
- `%let EmployeeTypeCode`
- `%let EMPLOYEETYPECODE`
- `%let employee_type_code`
- `%let EMPLOYEE_TYPE_CODE`
- <not like any of these>

# Poll Q6: When writing new SAS code, how would you name a macro extracting the rows with a specific employee type into a separate dataset?

- `%macro extract (inds, outds, emptype);`
- `%macro EXTRACT`
- `%macro Extract`
- `%macro extractemployees`
- `%macro ExtractEmployees`
- `%macro ExtractEmployeesOfType`
- `%macro EXTRACTEMPLOYEESOFTYPE`
- `%macro extract_employees_of_type`
- `%macro EXTRACT_EMPLOYEES_OF_TYPE`
- <not like any of these>

# Q & A