# Coding Standards Document

The coding standards document provides information on the format and structure code produced at Spoon is expected to have.

Any queries about the Coding Standards Document should be referred to the Software Manager.

Contents

[Coding Standards Document 1](#_Toc379028184)

[Version History 1](#_Toc379028185)

[Names 1](#_Toc379028186)

[Identifier Names 1](#_Toc379028187)

[Standard Java Naming Conventions 2](#_Toc379028188)

[Comments 2](#_Toc379028189)

[Header Comments 2](#_Toc379028190)

[Body Contents 2](#_Toc379028191)

[Code Structure 3](#_Toc379028192)

[Brackets 3](#_Toc379028193)

[Declarations 3](#_Toc379028194)

[Initialisation 3](#_Toc379028195)

## Version History

|  |  |
| --- | --- |
| Version | Description |
| 1.0 | Created 01/02/2014 |
| 1.1 | Removed requirement for version history within code. Added example of line comments within a method. Added @Params and @Returns to method comments |

## 

## Names

### Identifier Names

All Identifier names should directly describe the identifiers function or content. For example if the program deals with customer accounts, then the chosen names for the Identifiers could include customerName (a variable to store a customer’s name) or withdrawFunds (a method to withdraw from the account).

### Standard Java Naming Conventions

|  |  |  |
| --- | --- | --- |
| **Identifier Type** | **Rules** | **Examples** |
| Package | All letters are to be in lowercase. | package controlinterface  package calculator |
| Class | Noun to represent a class and it has to be in CamelCase whereby each new word begins with a capital letter. | class Customer  class Account  class VideoFile |
| Interface | A name that describes what the class does. It has to be in CamelCase. | interface Comparing  interface StreamVideo |
| Method | Use verb to describe what the method does. It has to be in mixed case whereby it is similar to CamelCase except that the first letter of the name is in lowercase. | int calculateBalance()  void displayUserInterface() |
| Variable | The name should represent the value of the variable. It has to be in mixed case. | string lastName  int accountBalance |
| Constant | The name has to be in uppercase separated by underscores if needed. | static final int MAX\_WIDTH |

## Comments

### Header Comments

At the beginning of each new class include;

Programmer’s name

Date created

Description of the class

### Body Contents

Use comments to provide descriptions of code that are not available within the code itself. Before commenting always consider refactoring your code first to make it clearer.

## Code Structure

### Brackets

Brackets are to begin after an argument with a single space and to end in line with the beginning of the statement or method name.

Example

methodName() {

if (condition) {

statements;

} else {

Statements;

}

}

### Declarations

All field declarations are to be made at the start of a block of code. Only one declaration per line.

### Initialisation

Local variables should be initialised where they are declared.

Full Sample Code Structure

/\*

\*Programmer: Steve Thorpe

\*Date: 24/02/2014

\*Description of Class: Sample class intended to demonstrate coding standards required for the SWEng project

\*

\*/

public class SampleClass extends Application {

/\*\* Sample method creating a button with text from the argument and adding

\*it to a JFX stackpane

\*@Param text: The input text string

@Return methodFinished: Returns true if the method is finished

\*/

public boolean testMethod(String text) {

Button button;

StackPane pane;

button = new Button(text)

pane = new StackPane();

//Adds button to the pane, getChildren() returns a list of the nodes currently add to pane.

pane.getChildren().add(button);

methodFinished = true

return methodFinished

}

}