Coding Standards

# Introduction

This document serves as the coding style guide for all Java source code projects created by clearAvenue, LLC.  Providing a uniform style makes it easier for others to read and understand the source code.

These standards cover formatting aesthetics and other programming conventions.

The attached file contains the Eclipse java formatting profile that conforms to these coding standards.



# Source Files

## File Names

The source file name must include the top-level class name appended with the following extension:

|  |
| --- |
| .java |

## Special Characters in Files

### Tabs

The source code will use Tabs for indentation.  The Tab size should be 4 characters.

## Formatting

### Braces

#### Optional Braces

Braces for all control statements are to be used, even when the body is empty or if there is only a single statement.

### Blocks

The following styles are used for blocks:

* All opening braces are on the same line as the definition followed by a line break
* Place a line break after the closing brace if it ends the body of a method, constructor, or class

#### Empty Blocks

Empty blocks may be concise unless it is part of a multi-block statement: if/else-if/else or try/catch/finally.

Example:

|  |
| --- |
| void doNothing() {} |

### Column Size

Column size should be set to 180 characters.

### Line Wrapping

* Prefer wrapping outer expression (keep nested expression on one line)
* For Binary expressions, wrap before the operator
* For 'multi-catch' statements, wrap before the '|' operator

### Whitespaces

* Separate control statements with a space between the reserved word and the opening parenthesis
* Before any open curly brace
* On both sides of any binary, ternary, or operator-like symbols
* Between the type and variable declaration

### Variable Declarations

#### One variable per declaration

Declare only one variable at a time.   Do not use:  int x, y;

#### Declare where needed

Declare/initialize variables as close to where needed as possible

#### No C-style array declarations

Place the square brackets as part of the type, not the variable:     String[] names is good, but not String names[]

## Naming

### Package names

Package names are all lowercase, with no underscores  
Example:

Valid: com.clearavenue.databaseobjects

Invalid: com.clearavenue.DB\_Objects

### Class names

Class names are all UpperCamelCase and are normally nouns or noun phrases.

Test classes should end with the phrase ‘Test’.   For example: NameTest

### Method names

Method names are all lowerCamelCase and are normally verbs or verb phrases.

Example:

Valid: getUsername(), printOrders()

Invalid: SubmitTicket(), DISPLAY()

### Constant names

Constant names are all UPPER\_CASE, with words separated with underscores.

It must be declared as static final.

Example:

static final IP\_ADDRESS

### Field names

Field names are all lowerCamelCase

Example:

public int numberOfIssues

### Parameter names

Parameter names are all lowerCamelCase.    Try to avoid single character parameter names.

String userId, String streetAddress

### Local Variable names

Local variable are all lowerCamelCase

int loopCounter

### Type Variable names

Type variables are all singe capital letters

T genericList

# Eclipse Settings

## Source > Clean Up actions

The following details are done during the Clean Up action:

* Change non static accesses to static members using declaring type
* Change indirect accesses to static members to direct accesses (accesses through subtypes)
* Remove unused imports
* Add missing '@Override' annotations
* Add missing '@Override' annotations to implementations of interface methods
* Add missing '@Deprecated' annotations
* Remove unnecessary casts
* Remove unnecessary '$NON-NLS$' tags

## Save actions

The following details are done during a save of a source file:

* Format all lines of source code based on format setting in this style guide as shown in the [clearAvenueEclipseFormatter.xml](https://clearavenue.atlassian.net/wiki/download/attachments/1015889/clearAvenueEclipseFormatter.xml?version=1&modificationDate=1435261267019&api=v2) file
* Organize imports
* Remove 'this' qualifier for non-static field accesses
* Convert control statement bodies to block
* Convert 'for' loops to enhanced 'for' loops
* Add final modifier to private fields
* Remove unused imports
* Add missing '@Override' annotations
* Add missing '@Override' annotations to implementations of interface methods
* Add missing '@Deprecated' annotations
* Remove unnecessary casts
* Remove unnecessary '$NON-NLS$' tags
* Remove trailing white spaces on all lines
* Correct indentation