

KaBook Store	DEL 5
Coding Standards	Date: 21/05/2020

# Frisbee

## Coding Standards

### 1 Introduction

Coding standards makes sure that all the developers working on the project are following specified specifications..

### 2 Description

Using a coding standard, they make source code more readable and the software project more maintainable. The code can be easily understood and proper consistency is maintained. They also help catch bugs and mistakes that are disguised as seemingly harmless code.

### 3 Coding Standards Specifications

- **1- Camel Case:**

Camel case combines words by capitalizing all words following the first word and removing the space, as follows:

Example: userLoginCount

- **2- Pascal Case:**

Pascal case combines words by capitalizing all words and removing the space, as follows:

Example: UserLoginCount

- **3- Snake Case:**

Snake case combines words by replacing each space with an underscore “\_” and, in the all caps version, all letters are capitalized as follows:

Example: user\_login\_count

Example(All caps): USER\_LOGIN\_COUNT

- **Pascal Case** will be used for **class** names.
- **Camel Case** will be used for **method** names, **method argument** names and **local variable** names.
- **Snake Case** will not be used in **identifiers**.
- Names that begin with a number will not be used .
- All **class variables** will be declared at the top of the class, static variables coming first.
- Using **Abbreviations** will be avoided except for **Abbreviations** commonly used as names like ID, NO etc.
- Using single character **variable** names like i, n etc. will be avoided.
- **Pascal Case** will be used for **table** names in MySQL.
- **Pascal Case** will be used for **file** names.

KaBook Store	DEL 5
Coding Standards	Date: 21/05/2020

In the backend code Model-View-Controller Pattern used, files related to Model stored in Model-Package and according to this rule View-Controller related files stored in related package. Package names started with uppercase. In the front end code spring-boot project file organization used.

In a project that is developed by many developer, commenting is an important case to make developers understand each others code. For this project, our aim is to inform each of us by commenting. For example, a variable can be declared by someone and other ones might spend a lot of time to understand this variable is used for what. In this case, other developers will be informed by commenting to the line of variable declaration. The same thing is applicable for other issues such as functions, classes etc. However this situation might cause lack of readability of the code. So, just the crucial points of the code should be commented. Otherwise, there will be too many comments and it will be difficult to find what developers looking for.

Code conventions are important to programmers for a number of reasons, first of all 80% of the lifetime cost of a piece of software goes to maintenance. Software maintenance in software engineering is the modification of a software product after delivery to correct faults, to improve performance or other attributes. Second one is hardly any software is maintained for its whole life by the original author and Code conventions improve the readability of the software, allowing engineers to understand new code more quickly and thoroughly. Also if our source code is a good sample, rest of the our project will be well packaged and clean as any other codes we create. Important topics to be considered at coding conventions :

- 1- File Names
  - 1.1 – File Suffixes
  - 1.2 – Common File Names
- 2- File Organization
  - 2.2 – Beginning Comments
  - 2.3 – Package and Import Statements
  - 2.4 – Class and Interface Declarations
- 3- Indentation
  - 3.1 – Line Length
  - 3.2 – Wrapping Lines
- 4- Comments
  - 4.1 – Implementation Comment Formats
    - 4.1.1 – Block Comments
    - 4.1.2 – Single – Line Comments
    - 4.1.3 – Trailing Comments
    - 4.1.4 – End-Of-Line Comments
  - 4.2 – Documentation Comments
- 5- Declarations
  - 5.1 – Number Per Line

KaBook Store	DEL 5
Coding Standards	Date: 21/05/2020

5.2 – Initialization

5.3 – Placement

5.4 – Class and Interface Declarations

6- Statements

6.1 – Simple Statements

6.2 – Compound Statements

6.3 – Return Statements

6.4 – If, if-else, if else-if else Statements

6.5 – For Statements

6.6 – While Statements

6.7 – Switch Statements

6.8 – Do-While Statements

6.9 – Try-Catch Statements

7- White Space

7.1 – Blank Lines

7.2 – Blank Spaces

8- Naming Conventions

“Important Note : Topics listed above are taken from "Oracle" and we tried to adhere to these topics while developing our project..

White spaces may seem less important than other standards. However, correct usage increases the readability among developers. In this project, we can list the important points about this criterion as follows.

1. In operations such as mathematical operations and assignment operations-in both Java and JavaScript codes- a space will be placed before and after operators.
2. Annotations in Java codes will be written directly above the name of the function or class.
3. One blank line will be left between each function. (Java and JavaScript.)
4. A blank line will be left before the value to be returned. (Java and JavaScript.)
5. A space will be left between the imports and the line where the class starts. (Java and JavaScript.)
6. There will be only one statement on each line. (Java and JavaScript.)
7. A space will be left after the commas separating the parameters. (Java and JavaScript.)