

Sports Center Management System	Version 1.0
Coding Standard	26/04/2018

Sports Center Management System – Coding Standard

Table of Contents

Introduction

1. Coding Standards
 - 1.1. Indentation
 - 1.2. Comments
 - 1.2.1. Block Comments
 - 1.2.2. Single-Line Comments
 - 1.2.3. Trailing Comments
 - 1.3. Classes, Functions, and Methods
 - 1.4. Source Files
 - 1.5. Variable Names
 - 1.6. Use of Braces

2. Coding Guidelines
 - 2.1. Length
 - 2.2. Spacing
 - 2.3. Wrapping Lines
 - 2.4. Variable Declarations
 - 2.5. Program Statements
 - 2.6. Use of Parentheses
 - 2.7. Coding for Efficiency vs. Coding for Readability
 - 2.8. Meaningful Error Messages
 - 2.9. Reasonably Sized Functions and Methods

3. Resources

Sports Center Management System	Version 1.0
Coding Standard	26/04/2018

Introduction

This document contains the coding conventions for SportSupport , based on the Java language coding standards presented in the [Java Language Specification](#), from Sun Microsystems Incorporated. Coding standards pertain to how the developer writes code. Coding guideline is just suggestions for developer. Usage of coding guidelines is not mandatory.

1. Coding Standards

1.1. Indentation

Four spaces should be used as the unit of indentation.

1.2. Comments

1.2.1. Block Comments

Block comments are used to provide descriptions of files, methods, data structures and algorithms. Block comments may be used at the beginning of each file and before each method. A block comment should be preceded by a blank line to set it apart from the rest of the code.

1.2.2. Single Line Comments

Short comments can appear on a single line indented to the level of the code that follows.

```

18     public @ResponseBody String addNewMember (@RequestParam String username
19         , @RequestParam String password) {
20         // @ResponseBody means the returned String is the response, not a view name
21         // @RequestParam means it is a parameter from the GET or POST request
22
23         member n = new member(username,password);

```

1.2.3. Trailing Comments

Very short comments can appear on the same line as the code they describe.

Sports Center Management System	Version 1.0
Coding Standard	26/04/2018

```

20     private String password;
21     private String statue; //This is for user's statue. It can be banned, owner, active, inactive, non-member.
22     private String status; //It can be gold, platinum, standard.
23     private String mail;
24     private int referenceNumber; //This is for a member's friend count that came with member.
25     private int branchAuthority; //This count for user's entered another branches except member's registered branch
26     private Date age;

```

1.3. Classes, Functions, and Methods

Keep all classes, functions, and methods reasonable size.

1.4. Source Files

Each Java source file contains a single public class or interface. Java source files should definitely have the following features :

- Package and Import statements
- Class Declaration
- Method Declarations if exists

1.5. Variable Names

Variable names should be meaningful and conform to the rules of JDK 8 .

```

16     private Integer id;
17     private String name;
18     private String surname;
19     private String username;
20     private String password;

```

1.6. Use of Braces

Functions and classes should surrounded by curly braces "{" and "}" .

2. Coding Guidelines

2.1. Length

Avoid lines longer than 80 characters. Lines longer than this may not be displayed properly on some terminals and tools.

Sports Center Management System	Version 1.0
Coding Standard	26/04/2018

2.2. Spacing

We set our spacing rules to the general coding rules. These are:

- a. A keyword followed by a parenthesis should be separated by a space.
- b. A blank space should appear after each comma in an argument list.
- c. All binary operators except “.” should be separated from their operands by spaces. Blank spaces should never separate unary operators such as unary minus, increment (“++”), and decrement (“—“) from their operands.
- d. Casts should be made followed by a blank space.

2.3. Wrapping Lines

When an expression will not fit on a single line, break it according to these general principles:

- Break after a comma.
- Break before an operator.
- Prefer higher-level breaks to lower-level breaks.
- Align the new line with the beginning of the expression at the same level on the previous line.

2.4. Variable Declarations

There must be one definition per line.

2.5. Program Statements

Each line should contain at most one statement. Also should not use nested statements.

2.6. Use of Parentheses

Parentheses should be used with care. Operations should be clearly visible.

2.7. Meaningful Error Messages

We used too many error messages in our project to improve the user's efficiency and we tried to direct the user with that attitude. We tried to write these messages as simple and clear as possible.

Sports Center Management System	Version 1.0
Coding Standard	26/04/2018

2.8. Reasonably Sized Functions and Methods

We tried to avoid long codes, we applied this attitude for improve readability and code security. In the case of very long codes, there might be problems in applying subtasks.

3. Resources

<https://docs.oracle.com/javase/tutorial/java/javaOO/variables.html>

<https://docs.oracle.com/javase/tutorial/java/index.html>

http://www.nws.noaa.gov/oh/hrl/developers_docs/General_Software_Standards.pdf