

Правила оформления кода, принятые на курсе "Разработчик на Spring Framework"

1. Имя группы проекта пишется маленькими буквами с разделением слов точкой. Например: "ru.otus".
2. Имя проекта пишется через дефис маленькими буквами в kebab-case.
3. Пакеты именуются так же как имя группы проекта.
4. Имена классов именуются с большой буквы в CamelCase.
5. Имена переменных, полей классов, параметров методов и самих методов пишутся с маленькой буквы в lowerCamelCase.
6. Имена констант пишутся большими буквами в SCREAMING_SNAKE_CASE.
7. Названия должны нести смысловую нагрузку и желательно не сокращаться (кроме общепринятых сокращений типа Impl, Config, App).
8. Имя класса/поля/переменной/метода должны соответствовать содержимому. Например, лучше не стоит даже временно класть значение счетчика в переменную обозначающую возраст.
9. Если класс принадлежит к определенному слою (Service, Dao, Transformer, Dto) то это нужно отразить в его названии. Исключением являются доменные объекты. Model и Domain (а так же Data и Object) не нужно добавлять в название.
10. Интерфейсам нужно давать "нормальное" имя без приставок, показывающих принадлежность к интерфейсам, а к имени реализации добавлять или (лучше) добавить в название особенности реализации (например, PersonDaoJdbc для интерфейса PersonDao).
11. Классы должны лежать в пакетах. Главный класс, точка входа в приложение, лежит в корневом пакете (имя группы + еще что-то, например: "ru.otus.homework"), а все остальные классы должны лежать в подпакетах.
12. Все поля классов должны быть приватными и быть доступными только через методы (геттеры и сеттеры).
13. Если значение поля не меняется в процессе выполнения, оно должно быть final.
14. Константы должны быть static final. Без static это не константы.
15. Хелперы (утилитные классы со статическими методами) должны быть final и иметь приватный конструктор. Но в spring лучше использовать полноценные singleton-бины. Без статики.
16. Код должен быть отформатирован так же, как это бы произошло при нажатии сочетания клавиш Alt + Ctrl + L в IDEA (пробелы после запятых, по краям от арифметических знаков и т.д.).
17. Желательно писать блок {} даже для if-а и for-а (для for-а это обязательно) с телом в одну строку. Так же это блок должен находиться на следующей строке после оператора. Исключением является, если тело if содержит return.
18. Если несколько строк кода выполняют законченное действие, то они должны быть вынесены в метод.
19. Если один или несколько методов не относятся напрямую к логике класса, то они должны быть вынесены в отдельный класс (например, сервис).

Данные правила подготовлены на основе следующих соглашений:

<https://www.oracle.com/technetwork/java/javase/documentation/codeconvtoc-136057.html>

<https://maven.apache.org/guides/mini/guide-naming-conventions.html>

<https://google.github.io/styleguide/javaguide.html>

<https://github.com/spring-projects/spring-framework/wiki/Code-Style>