

Javaプログラミング実習

17. コーディング規約

株式会社ジードライブ

今回学ぶこと

- コーディング規約とは
- コーディング規約の例
- Eclipseのフォーマット機能

コーディング規約とは

- コードの可読性を高めバグの少ないプログラムを記述するためのルール
- コーディング規約の種類
 - ネーミングに関する規約
 - コードのフォーマットに関する規約
 - 多くの人が間違える問題に関する規約
- 世の中には様々な人や団体が作成したコーディング規約がある

コーディング規約文書の例

- Java コーディング標準
 - <http://objectclub.jp/technicaldoc/java/>
 - 永和システムマネジメント社内の「オブジェクト倶楽部」が作成したドキュメント
- Java コーディング規約
 - <https://future-architect.github.io/coding-standards/documents/forJava/>
 - フューチャー株式会社が作成したドキュメント

コーディング規約の例

出典：Javaコーディング標準, オブジェクト倶楽部

- 定数
 - 大文字をアンダーバー(_)でつないだもの
 - UPPER_CASE_WITH_UNDERSCORE
- メソッド名
 - 最初小文字で、あとは区切りを大文字
- 英語と日本語
 - すべての識別子の名前は英語を基本とし、別に、日英対応用語辞書を作成してプロジェクトの全ライフサイクルでメンテナンスすること

コーディング規約の例

出典：Javaコーディング標準, オブジェクト倶楽部

- スコープが狭い名前
 - スコープが狭い変数名は、型名を略したものを使って良い
 - 例： `ServletContext sc = getServletContext();`
- 意味がとれる名前
 - 変数名から役割が読み取れる名前を好め
 - 悪い例： `copy(s1, s2)`
 - 良い例： `copy(from, to)`
- 無意味な名前
 - Info, Data, Temp, Str, Buf という名前は再考を要する

コーディング規約の例

出典：Javaコーディング標準, オブジェクト倶楽部

- import
 - import では、* をなるべく使わない。同じパッケージから3つ以上のクラスを import する場合は、* を使う
- 宣言と初期化
 - ローカル変数は、初期値と共に宣言せよ
 - 理由：変数の値に関する仮定を最小化する

プログラムサイズ

- プログラムサイズの計測方法は様々あるが、一般的な傾向としてプログラムのブロックサイズが大きくなるほどバグの発生頻度が高まると言われている
- プログラムのブロックサイズの例
 - 1行の文字数を80字以内にする
 - 1メソッドの行数を20行以内にする

Eclipseのフォーマット機能

- Eclipse では Ctrl + Shift + F でフォーマットをすることができる
 - 行を選択している場合、選択行のみがフォーマットされる
- Eclipseでは、アノテーションを使いフォーマットされる範囲を制御できる
 - 設定でOff/Onのアノテーションを有効にする必要がある

フォーマットのON/OFF設定

- 設定ウィンドウ内の
Java > コード・スタイル > フォーマッター
で編集ボタンを押す



フォーマットのON/OFF設定

- Off/Onタグを使用可能にする
 - @formatter:off と記述された行以降はフォーマットされない
 - @formatter:on と記述することで、その行以降はフォーマットが有効になる

▼ Off/On タグ

off/on タグはソース・ファイル上のどのコメントでもフォーマッター

- 各ファイルの先頭で、フォーマッターを使用可能にします。
- フォーマッターが off タグを見つけるたびに、その後のコメントお
- フォーマッターが on タグを見つけるたびに、その後のコメントお

☒ off/on タグを使用可能にする

off タグ:	@formatter:off
on タグ:	@formatter:on

フォーマットのON/OFF設定

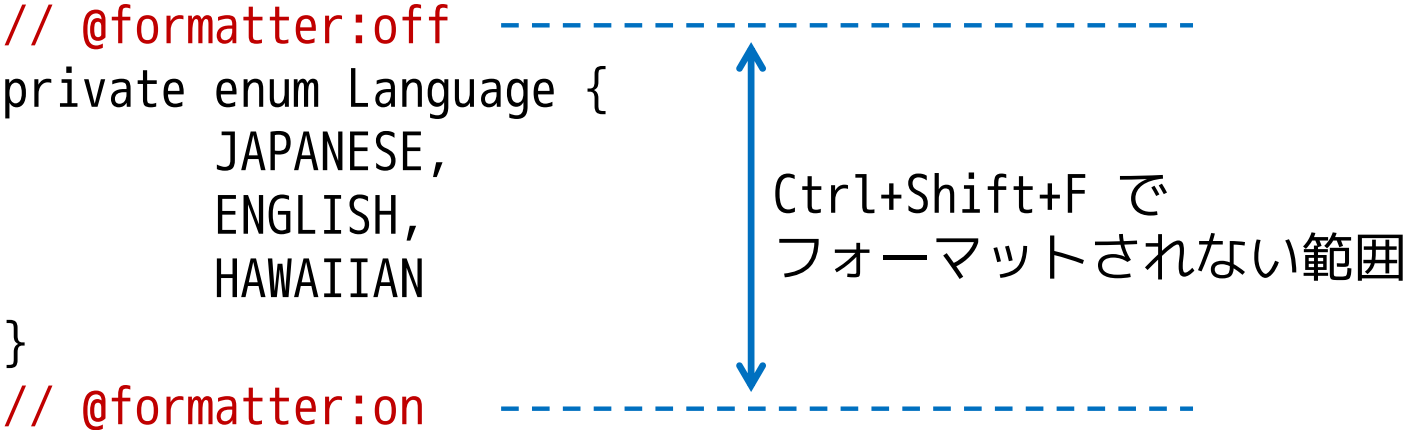
- Off/Onタグの利用例
 - Off/Onタグは行コメントとして記述する

```
package practice14_2;

public class Greeting {

    // @formatter:off -----
    private enum Language {
        JAPANESE,
        ENGLISH,
        HAWAIIAN
    }
    // @formatter:on -----

    private static void greet(Language type) { ....
```



Ctrl+Shift+F で
フォーマットされない範囲