

卒業研究プログラミング コーディング規約

麻生情報ビジネス専門学校

情報システム専攻科 3 年 A クラス チームコスモ

目次

1. 目的
 - ① コーディング規約の必要性
2. ファイルについて
 - ① 文字コード
3. 命名規則
 - ① ファイル名
 - ② クラス名
 - ③ メソッド名
 - ④ 変数名
 - ⑤ パッケージ名
 - ⑥ 定数名
4. コーディングスタイル
 - ① インデント
 - ② 中かっこ
 - ③ メソッドの長さ
 - ④ 命令の長さ
 - ⑤ コメント
5. 禁止事項
 - ① 大文字小文字で変数を区別しない
 - ② 適切な改行を入れる
 - ③ カッコの省略をしない

コーディング規約

1. 目的

① コーディング規約の必要性について

チーム内の人間がコーディング規約を守ることで、バランスの取れたソースコードに仕上げる事が可能になる。

バランスの取れたコードを記述することで、可読性の向上やチーム内での矛盾、作成者個人にしかわからない記述を減少させることができ、プロジェクトをスムーズに進めることが可能となる。

コーディング規約

1 ファイルについて

① 文字コード

ソースファイルの文字コードはすべて UTF-8 とする。

コーディング規約

2 命名規則

① ファイル名

Java ファイルはクラス名と同じファイル名とする（大文字小文字も一緒にする）

例）Main クラスのファイルは Main.java とする（main.java は不可）

② クラス名

先頭文字は大文字とし、文節は大文字で表す

例）メインクラス

・ MainActivity

・ MainServlet

③ メソッド名

・ 先頭を含め基本を小文字とし、言葉の区切りのみ大文字にする。

例）

fileSend()

checkVersion()

addItem()

showActionBar()

・ 先頭は動詞から始める下記の例以外にも様々あるので各サイトを参考にする。

（参考：<https://bit.ly/2kiYUUt>）

例）

動詞＋名刺

getDate()

deleteDate()

showError()

コーディング規約

④ 変数名

- ・先頭文字は小文字にし、言葉の区切りは大文字にする
- ・長すぎる変数名にしない
- ・意味のある変数名にする。(for の `int i = 0;` などは無効とする)

例)

○`DateBeans dateBeans = null;`

×`DateBeans DateBeans = null;`

×`DateBeans d = null;`

○`for(int i = 0 ; i < 10; i++) {`

⑤ パッケージ名

- ・全てを小文字とし、記号などは使わない
- ・役割ごとに作成する

⑥ 定数名

- ・定数名は全て大文字とし、文字の区切りはアンダーバー (`_`) を使用する。
- ・宣言には必ず `final` をつける
- ・`public` な定数に関しては `static` をつける
- ・名前から何の定数か想像できる名前にする

コーディング規約

3 コーディングスタイル

① インデント

インデントは4文字で統一する。(tab 設定)

② 中かっこ

- ・中かっこの開始の前には改行をいれない

例)

正-----

```
if(parm == 1){  
    print();  
}
```

誤-----

```
if(parm == 1)  
{  
    print();  
}
```

③ 命令の長さ

1行が画面に収まらない場合は改行して見やすくする。

コーディング規約

④ コメント

- ・コメントは処理の上部に記入すること。

例)

正-----

```
//String 型の変数 str を生成する
```

```
String str = “string 型変数”;
```

誤-----

```
String str = “string 型変数”;
```

```
//String 型の変数 str を生成する
```

- ・メソッドのコメントはメソッド上部、かつ下記の形式で記述する

例)

```
/**
```

```
 * メソッドの説明
```

```
 * @param 引数
```

```
 * @return 戻り値
```

```
 */
```

```
public String sample (int number, String message){
```

```
}
```