

# Chapter 1

## オブジェクト指向プログラミング の初歩

上級プログラミング 1 では主にLinux環境を使います（Windows環境を使うときは指示します。）

### 1.1 まずは動かしてみよう

#### (1)パスの確認

ターミナルエミュレータのシェル上で次のコマンドを実行せよ。

```
which java
```

この結果が`/usr/bin/java`であったとすると、

```
echo $PATH
```

また、結果として`/usr/bin`がパスに含まれていることを確認し、

```
ls /usr/bin | grep java
```

とした結果に`java`コマンド（実行用）と`javac`コマンド（コンパイル用）が含まれていることを確認せよ。（Linuxコマンドの意味を確認せよ。）

## (2) プログラムを動かしてみる

次のプログラムをemacsを使って次のプログラムをテキストファイルに打ち込みセーブせよ。ただし、ファイル名は「HelloWorld.java」とする。

```
class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello world");  
    }  
}
```

ファイルが出来たら、次のコマンドを実行してコンパイルせよ。

```
javac HelloWorld.java
```

どんな名前のファイルが生成されたか確認せよ。さらに次のコマンドを実行し、その結果を確認せよ。

```
java HelloWorld
```

【とても大事!!】javaコマンドの後にはファイル名ではなく、クラス名がくることに注意せよ。

## (3) Windows環境で動かしてみる

Windows環境にコンパイル済みのファイル（ソースファイルではない!!）をコピーし、コマンドプロンプトを開き、ファイルを置いたディレクトリまでカレントディレクトリを移動した上で次のコマンドを実行せよ。どうなるか。

```
java HelloWorld
```

## (4) クラスを作ってみる

次の要件を満たすクラスを作成せよ。

- クラス名をSaifuとする。（ファイル名はSaifu.javaとする）
- int型のokaneフィールドを備える。（1000で初期化する）
- 引数int型、戻り値void型のsetOkaneメソッドを備える。setOkaneメソッドは引数の値をokaneフィールドに代入する。

- 引数なし、戻り値int型のgetOkaneメソッドを備える。getOkaneメソッドはokaneフィールドの値を返す。

フィールドやメソッドの定義には、戻り値の型の指定の前にpublicをつけること（今後、この意味を説明します）<sup>1</sup>。これをコンパイルせよ。また次のコマンドを実行したらどうなるか？

```
java Saifu
```

## (5)実行用のクラスを作ってみる

次の要件を満たすクラスを作成せよ。

- クラス名をJikkou01とする。（ファイル名はJikkou01.javaとする<sup>2</sup>）
- mainメソッドを備え、次の処理を行う。
  - SaifuクラスのオブジェクトmySaifuを生成する。
  - getOkaneメソッドを使ってmySaifuオブジェクトの現在のokaneフィールドの値を取得する。
  - 取得した値に5000を加えてsetOkaneメソッドを使ってokaneフィールドにセットする。
  - okaneフィールドの値を画面に表示せよ。例えば、okaneの値を画面に表示する処理は次のように行うこと<sup>3</sup>。

```
System.out.println("okane="+mySaifu.getOkane())
```

## (6)発展問題

- 引数に与えられた文字列数が1つの場合は、その文字列を表示し、引数が2つ以上の場合は「引数はひとつだけです。」というメッセージを表示するアプリケーションを作成せよ。
- He said “it is cold today” . という英文を表示せよ。ただし、ダブルクォーテーションマークも表示すること。

<sup>1</sup>今回はクラス変数やクラスメソッドを作るわけではないので、staticはmainメソッドの前以外では（とりあえず）つけないこと。今後これについても説明します。

<sup>2</sup>1ファイルにクラスは一つ。クラス名とファイル名（拡張子除く）を同じにする癖をつけよう。

<sup>3</sup>Javaでは文字列に値を足し算すると文字列になることを利用している。