

## Chapter 2

# オブジェクト指向プログラミング の初歩（つづき）

以下の指示に従って、クラス、オブジェクトを作成・実行せよ。なお、文中では個別に指示しないが、変更を行ったら適宜コンパイルすること。なお、XXクラスのオブジェクトのことを単にXXオブジェクトと呼ぶことがある。

### 2.1 基本データ型と制御構文、配列

次のクラスを作成し、実行せよ。

- クラス名はTest02Kihon
- フィールドにはfloat型の変数aveを用意する。(0.0で初期化すること)
- mainメソッドを作成し、float型の配列（配列名data）を宣言する。ただし、要素数は6つとする。先に宣言を行ない、領域を確保すること(newを使え)。
- この配列dataのそれぞれの要素に2.3、3.4、1.5、30.9、2.1、9.2を代入せよ。その後、以下の処理をするものとする。
  - for文を使って要素の値の平均を求め、aveに格納せよ。ただし、今後、要素数を変更することを見越して、終了条件にはdataの要素数(lengthを使え)を用いよ。

- aveが5以上であれば more と、小さければ less と画面に表示せよ(System.out.println(文字列)とすると、引数の文字列が画面に表示されることを使え)。

## 2.2 mainメソッドの引数

次のクラスを作成し、実行せよ。

- クラス名はTest02Length
- メソッドとしてはmainメソッドのみ。以下ののようにTest02Lengthクラス実行時の引数（この例の場合は This, is, a, penという4つの文字列）をmainメソッドの引数から取得し、その文字列の長さを、4, 2, 1, 3 などと画面に出力せよ（出力の仕方はお好みで）。

```
java Test02Length This is a pen
```

## 2.3 Stringクラス

次の要件を満たすDogクラスおよびTest02Aクラスを作成せよ。（文字列=Stringオブジェクトの操作に慣れる）

- まず次のクラスを作成せよ。

```
class Dog{
    String name="HACHI";
    void setName(String s){
        name=s;
    }
    String getName(){
        return name;
    }
}
class Test02A{
    public static void main(String[] args){
    }
}
```

- Test02A クラスのmainメソッドにおいて、Dogクラスのakitaオブジェクトを生成せよ。
- akitaオブジェクトのnameフィールドの値となる文字列（今の段階では”HACHI”）を取得し、その文字列に”-KOU”を追加<sup>1</sup>してakitaオブジェクトにセットしなおす処理をTest02Aクラスのmainメソッドに追加せよ。（setNameメソッドとgetNameメソッドを使うこと。）ただし、この方法は2種類ある（メソッドを用いる方法と演算子を用いる方法）通常は、演算子を用いることが多いので、本問では演算子を使う方法を試せ。
- akitaオブジェクトのnameフィールドが保持する文字列の長さを表示する処理<sup>2</sup>をmainメソッドの上記の処理部分のあとに追加せよ。
- さらに、akitaオブジェクトのnameフィールドが保持する文字列を小文字にして表示する処理をmainメソッドの上記の処理部分のあとに追加せよ。
- akitaオブジェクトのnameフィールドが保持する文字列に含まれるHAをPOに変換して表示する処理をmainメソッドの上記の処理部分のあとに追加せよ。
- akitaオブジェクトのnameフィールドのうち、ハイフン”-”よりも前の部分を取り出し、表示する処理をmainメソッドの上記の処理部分のあとに追加せよ。

ヒント：<http://java.sun.com/javase/ja/6/docs/ja/api/java/lang/String.html>を参考に方法を調べよ。

## 2.4 StringクラスとStringBufferクラス

Stringクラスのみを使う場合とStringBufferクラスを用いる場合で以下の処理の実行時間を比較したい。時間の計測はおおよそでよい。

- クラス名はTest02Bとする。

---

<sup>1</sup>ハイフン”-”を忘れないように。

<sup>2</sup>System.out.println(Stringオブジェクト)でStringオブジェクト(=文字列)が画面に表示されることを思いだそう。また、文字列と数値を足し算すると値が連結された文字列になることに注意

- mainメソッドを用意し、その中で文字列”AAA”の後に文字列”BCDEF”を10000回繰り返し追加する処理を行うプログラムとせよ（”BCDEF”の追加にはfor文を用いよ）。ただし、この文字列は表示せず、一連の処理が終わったら画面に”end”という文字を出すこと。
- Stringクラスのみを使う場合には演算子+を用いる方法を用い、StringBufferクラスを用いる場合には、以下の処理を行うものとする。

```
StringBuffer strBuff=new StringBuffer("AAA");
for(int i=0; i<10000; i++){
    strBuff.append("BCDEF");
}
String str=strBuff.toString();
```