

3章 いくつかの細かいルール

Chapter

03

3-0 この章で学ぶこと

Javaのプログラムは、クラスというものを書けばいいんだ、ということを前の章では学びました。クラスの中を書くものは、データとメソッドです。メソッドには、コンピュータにやらせたい処理内容を記述します。

そこでさっそく、読者の中には、よーし、MyDatabaseクラスを書いてみよう、とか、GetMyHomePlanクラスを書いて最良のマイホーム・ゲット作戦を立てよう、とか、デジカメの写真を整理するためのMyPhotoAlbumクラスを書こう、などなどと意気込んでいるかたもおられるでしょう。好奇心と創造的意欲は、あらゆる種類の勉強のための最良のエンジンです。

しかしJavaプログラミングの玄関の敷居をちょっとまたいだけ、という現段階では、いきなり本格的なクラスの開発に取り掛かっても、わからないことが多すぎるので、遅々として進まないことにきつとなるでしょう。

この章では、Javaのプログラムを書いていくうえで必要な、Javaプログラムの細かい基本ルールをいくつか説明します。

Javaにかぎらず、コンピュータ関連のルールは、コンピュータが有限機械であることに関連している、という理解を根底に持ってください。コンピュータは、CPUの速度も、メモリの容量も、つねに一定の有限です。そういう有限の機械に、

なるべく効率良く仕事をさせるためには、ルールという「枠」を設定し、その枠の中でコンピュータを使うほうが得策なのです。

そこで、コンピュータのプログラミング言語には、人間の言葉のような自由や可変性がありません。

3-1 名前と文字

プログラムを書いていくとき、あなたは、さまざまな名前を発明し、名前を駆使します。たとえば、人の年齢なら age とか、住所なら address、商品の価格なら price、といった名前を付けるでしょう。

クラスに名前を付けますし、クラスの中のデータやメソッドにも名前を付けます。メソッドの中で new して作るオブジェクトにも、名前を付けます。たとえばこんなふうに：

```
//赤みの強い茶色のColorオブジェクトを作り、ebiChaと名付ける
Color ebiCha = new Color(116, 69, 67);
```

Java プログラムでは、名前用に使える文字の種類が、次のように制限されています：

名前に見える文字

- ・名前の冒頭用の文字：アルファベット（半角英文字）と `_` と `$`
- ・名前の2文字目以降用の文字：名前の冒頭用の文字（上述）とアラビア数字（0から9まで）

すなわち、名前の最初の文字は、AからZまで、またはaからzまでの文字、または `_`（アンダスコア文字、下線文字）、または `$`（ドル記号）です。そして名前の2文字目以降の文字は、最初の文字に許される文字プラス、0から9までの数字です。

名前の冒頭の文字として、数字の使用が認められていないのは、たぶん、数字に出会うとコンパイラは、「これは数値か？、名前か？」という判断をいちいちしなければならぬからでしょうね。

記号文字として `_` と `$` だけが認められているのは、この二つが、`+` `-` `*` `/`（加減乗除）のような演算子や、そのほかのJava独特の意味のある記号（`.` `[]` `<` `>` `()` `::` などなど）に含まれない文字だからでしょう。

なお、文字も記号も数字も、すべて、いわゆる半角文字です。

00

01

02

03

04

05

06

07

08

09

10

名前の冒頭以外の文字としては、数字の使用も認められています。ですからたとえば、次のような名前は違法です：

```
36thInfantryDivision
2banmeNoMoji
@nifty
#shohinTanka
He&She
AreYouLoneIy?
```

しかし、次のような名前は OK です：

```
_end
start_
$price
$203highland
room365
and_I_love_her
```

ここまでを読んだ読者は、プログラムの中の名前に日本語の文字（漢字や仮名）を使えないことに不満を抱いたかもしれません。

実は、本当は、いわゆるユニコード（Unicode）という、文字コードの世界的な規格の中で定義されている文字ならば、日本語文字にかぎらず、**どんな文字でも使える**んです。たとえば int 型（整数型）の変数の名前を、次のように日本語にすることは、

```
int 裕子の年齢; //整数タイプの変数を宣言する
裕子の年齢 = 25; //値を代入する
int 雅夫の年齢 = 裕子の年齢 + 4; //整数タイプの変数を宣言し値を初期化する
```

完全に OK です。

しかし、Java プログラミングを教える立場としては、私は上のルール（英語アルファベットを中心とするかぎられた範囲の文字）に固執します。なぜならば、日本に住んで、コンピュータの上には日本語を使える OS 環境が当然のようにあって、自分自身も日本語の読み書きに堪能、となり近所の人も日本語はぺらぺら、という状況の中では日本語を使いたくなるのは当然ですが、でも、名前に日本語を駆使しているプログラムは、それだけで、よその国の人が理解できないきわめてローカルな物になってしまうのです。

よその国の人が理解できない、という意味は、その人たちが日本語を知らないだけでなく、ほかの国のコンピュータの上であなたが書いたソースファイルを見ると、日本語の文字の部分が非常に奇怪な表示になってしまうのです。いや、具体

的にどうなるかは知りませんが、とにかく珍妙な記号や空白の羅列になることでしょう。

ですから、非常に限られた、英語のアルファベットといくつかの記号文字と数字、という文字集合にあえて限定して、プログラムを書いていったほうがよいのです。それらの文字は、別の言い方をすると、キーボードの上に刻印されている文字、記号、そして数字です。これらの文字ならば、世界中どここの国のコンピュータの上でも読めるし、また、誰もが入力できるのです。

以前私は、ギリシアの人が自分の国のレンタル・ビデオ屋さんのために書いたソフトのソースコードを、インターネットのフォーラム(掲示板)の上で見たことがあります。私のコンピュータはギリシアの文字を正しく表示できませんから、そのソースコードは私のコンピュータの上ではものすごく見づらかったです。そのギリシア人は私に、自作のプログラムに対する意見を求めてきたのですが、「お願いだからプログラム中の名前は英語のアルファベット文字を使ってくれ!」と悲鳴のようなメールを私は彼に送りました。

「私は日本人だから日本語の文字を使ってくれ」という願いは、グローバルなネットワークであるインターネットの上ではきわめて不合理でしょう。でも、上のようなお願いならばむしろ、世界中の人に理解され同情されることでしょう。

3-2 プログラマが使えない名前= Javaのキーワード(予約語)

単純なありふれた英語の単語なのに、プログラマが自分のプログラムの中で名前として使えない言葉がいくつかあります。たとえば、なにかの名前に `this` なんて使っていると、途端にコンパイラ `javac` は文句を言います。しかもその文句(エラー・メッセージ)は、「`this`はJavaの予約語だ、キーワードだ。だから使えな!」というわかりやすいメッセージではありませんから、初心者は戸惑います(後日、機会があるときに、実際に `this` という名前を試してみてください)。

Javaが特定の目的のために使うので、プログラマが名前として自由に使えない言葉のことを、キーワードとか、予約語といいます。

そういう、“プログラマにとっての禁句”は、次に示すように、ぜんぶで51個あります。中には、`int`とか`class`とか`public`や`new`など、あなたが本書の中ですで見えたものもありますね。

00

01

02

03

04

05

06

07

08

09

10