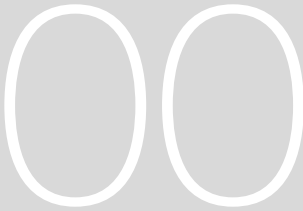


0章

Javaを使うための準備

Chapter



0-0 Java とは何か？ ——それはプログラミング言語である

あなたはこれから、本書をガイド役として使って、Javaの勉強をしていきます。しかし、Javaの勉強とは、もうすこし具体的に言うと、何を勉強することでしょうか？。それよりもまず、そもそも、Javaとは一体何でしょうか？。

Javaを使う立場は、大きく二つの種類に分かれます。ひとつは、自分のコンピュータや会社のコンピュータの上で、Javaで作られたプログラムを動かす（使う）立場です。この立場から見ると、「Javaはプラットフォームである」という言い方ができます。プラットフォームとは、電車の駅のプラットフォームでご存じのように、『台』という意味です。ここではとくに、コンピュータを仕事に使うための台=環境を指しています。

プラットフォームとしてJavaを使う場合には、Java全体の中の、“（すでにできあがっている）Javaプログラムを動かす部分”がコンピュータの上にインストールされていなければなりません。

もうひとつの立場は、Javaでプログラムを作っていく（=書いていく）立場です。あなたをはじめ、本書の読者はこの立場に属するかたのはずです。Javaを使ってプログラムを作る立場の者にとってJavaは、「プログラミング言語」です。プログラミング言語とは、プログラミングのために使用する言語という意味です。Java

は言語です。プログラミングは、言語、すなわち言葉を使って、プログラムと呼ばれる文を書くことです。

Javaでは、たとえば、こんな文を書きます：

```
int age;           //ageという名前の整数を宣言する (たぶん誰かの年齢を表すのだろう)
age = 19;         //ageの値を19にする
```

上のわずか2行を見てもおわかりのように、プログラミング言語の言葉や文は、一応なんとか、「人間が読んでわかる言葉や文」です。これに対してコンピュータが理解して実行できる命令は、人間には理解できない2進数の列です。

あなたが学ぶ第一外国語や第二外国語、すなわち英語、中国語、アラビア語などの言語はすべて、人間同士がコミュニケーションするための言語です。

しかしJavaのようなプログラミング言語は、コンピュータに命令を伝えるための言語です。ですからときどき、プログラミング言語は「コンピュータ言語」とも呼ばれます。しかし上で述べたように、人間が読めるプログラミング言語の文をコンピュータがそのまま理解して実行することは不可能です。

人間の言語は、人間の心の中や脳の中にあります。では、コンピュータ言語＝プログラミング言語は、どこに、どんな形であるのでしょうか？。

実はプログラミング言語は、次の三つの部分でできています：

プログラミング言語の構成

- ① 言語のルール (使用する言葉や文字の種類、それらの文法や意味など)
 - ② そのプログラミング言語で書いた、人間が読めるプログラムを、コンピュータの命令に翻訳または通訳するプログラム
 - ③ 上の②の段階で翻訳または通訳されたプログラムをコンピュータに与えて実行させるプログラム。
-

完成品としてのJavaプログラムは、上の②の過程を経た——コンピュータの命令に翻訳ずみの——命令の集まりです。そこで「Javaはプラットフォームである」と見る立場の人にとって必要なのは、上の③のプログラムだけです。

しかし、プログラミング言語としてJavaを使用し、プログラミング言語としてのJavaを学んでいくあなたには、①②③のすべてが必要です。たとえば、ここでは早速、intという言葉はJavaでは整数 (integer) という意味だ、というルールを

00

01

02

03

04

05

06

07

08

09

10

知りました。本書では、Javaでプログラミングを書くために必要ないちばん基本的なルールを勉強します。

そしてまた本書では、学んだルールに基づいて小さな簡単なプログラムをいくつも書いていきます…上の①の学習と応用です。そして、書いたプログラムをコンピュータの命令集に翻訳する作業も、本書で何度も経験します…上の②の過程です。またもちろん、コンピュータの命令へと翻訳されたプログラムを、実際にコンピュータの上で動かしてみる経験もします…それは上の③ですね。

なにか一つでも外国語をまあまあマスターされたかたは、「言葉は慣れである」という認識を持っておられるでしょう。とにかく、たくさん本を読んだり、多くの人と会って話をするのが、その言葉に対する慣れを作り、そして慣れが、あなたの外国語の実力を築きます。

プログラミング言語も、それと同じです。たくさんプログラムを書き、それらの翻訳と実行の失敗や成功を何度も繰り返すことによって、徐々に徐々に、プログラミングの実力がついていきます。

習うより慣れろ、とよく言いますが、プログラミングの場合は、言語のルールなどを学ぶことと、それを実際に使ってみることへの慣れ、つまり、習うことと慣れることの両方が重要です。言い換えるとプログラミングの学習は、数をこなすことと、そして多くの経験（失敗や成功）の意味を理論的に理解すること、この二つが車の両輪のように必要です。本書は、あなたの脳の中に、この強力な両輪ができていく過程をガイドします。

実際にやること（書くこと）と、やったことの結果と意味を理解すること、あなたの学習過程全体を通じて、この両方を充実させてください。

0-1 Javaプログラミングの三段跳び

Javaでプログラムを作る過程——プログラムを書いてそれを動かすまでの過程——は、次のような三つのステップです：

Java プログラミングの流れ

①ソースプログラムを書く

テキストエディタ（通称“エディタ”）などを使って*1、ソースプログラム、つまり「人間が読んで理解できるプログラム」を書きます。たとえば、あるソースプログラムの中に次のような文があると：

```
System.out.println("ありがとう");
```

`System.out.println`は“画面に文字を表示せよ”というJavaの命令だ、ということがわかれば、この文が画面に「ありがとう」というメッセージを表示するプログラムであることが、人間にもわかります。

このように、人間が読んで分かるJavaの文を書きつらねた文書のことを、Javaのソースプログラム、と呼びます。ソース (source) は、最初の、とか、起源の、とか、元々の、という意味です。つまり、コンピュータのプログラムのいちばん最初の形、スタート時点の形だから、ソースプログラムと言うんですね。トンカツにかけるソースは、sourceではなくsauceでした。

②ソースプログラムをコンパイルする

ソースプログラムは、人間には理解できますがコンピュータがそれを読んで理解し実行することはできません。そこで、ソースプログラムをコンピュータの命令に翻訳する作業が必要です。この翻訳作業のことを、コンパイル (compile) といいます。たとえば上の、`System.out.println("ありがとう");`というJavaの文をコンパイルすると、人間が見ても理解できない二進数の羅列へと変わります。その無表情な冷たい二進数の羅列こそが、コンピュータがプログラムとして実行できる命令の集まりです。

コンパイル後のこの二進数の羅列のことを、ソースプログラムに対して「何プログラム」と呼ぶのか？。それはいろんな呼び方がありますが、初心者がそれらの用語を一々覚える必要はありません。本書ではとりあえず、英語のexecutable（イグゼキュタブル、実行可能なもの）の意識として「実行形式」とか「実行プログラ

※1 Windowsには「メモ帳」や「ワードパッド」などのエディタプログラムが最初からあります。低機能なエディタですが、Javaの勉強の、ごくごく最初のうちはこれらを使ってもかまいません。エディタを使い慣れていないかたは、一太郎やワードのようなワープロソフトを使ってソースプログラムを書き、保存するときに“テキスト形式で保存”を指定すれば、結果は、エディタを使ったときと同じ形式の、いちばんシンプルなテキスト文書ができあがります。