

1 章 まず Java の初体験を

Chapter

01

1-0 この章の目的

この第1章では、0章で述べた、Javaプログラミングの3つのステップ（ソースプログラムを書く、コンパイルする、実行する）を、ひととおりすべて経験していただきます。现阶段ではJava言語についてわからないことだらけでしょうが、とりあえず何もわからなくてもかまいません。とにかくここでは、Javaプログラミングの1から10までをすべてひととおり、最初から最後までやってみます。

海で初めて泳ごうとする人は、最初は海をとて怖いと感じるでしょう。そこで、初めて海水浴をする人は、浅いところで足を海水につけてみたり、お風呂ぐらいの深さのところを全身を海水につけたりして「水に慣れる」訓練をします。そうするとだんだん海が怖くなくなり、泳ぎを覚える意欲が湧いてきます。

というわけでこの章では、Javaプログラミングという海の、とても浅いところに足を浸して、二三步、水の中を歩いてみてください。そうすると、Javaでプログラミングをする作業のスタートからゴールまでの全体像が、あなたの脳にくっきりと刻印されるでしょう。それが、この章の目的です。

1-1 コマンド入力に関する準備知識

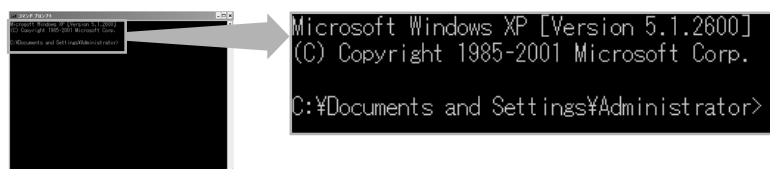
まず、簡単な Java プログラムの、ソースファイルを作りましょう。

0章の終わりのほうで述べたように、エディタやワープロの立ち上げ方は、どんな方法でもかまいません。

しかし、どんな方法を採用したとしても、これからコンパイラ `javac` やインタプリタ `java` を使っていくための準備として、MS-DOS プロンプト (Windows 98 / Me まで)、またはコマンドプロンプト (Windows XP)、またはターミナルプログラム (Linux の上の `gnome-terminal` など) を起動してください。デスクトップ画面の中に、これらのプログラムのショートカットやランチャ (Linux の場合) を設けておくと、今後便利です。



❏ 図 1-1 MS-DOS プロンプト (Windows 95, 98)



❏ 図 1-2 コマンドプロンプト (Windows XP)



❏ 図 1-3 ターミナルプログラム (Linux の上の `gnome-terminal`)

00

01

02

03

04

05

06

07

08

09

10

Windows XPのコマンドプロンプトについては、次のURLに超親切なガイドがありますから、このサイトの、すくなくとも最初の2～3ページを見ることをおすすめします*1。

今後ずっと本格的なプログラミングをやっていきたいかた向けに、私個人がおすすめするやり方は、ワープロではなくエディタを使うことです。エディタのほうが動作が軽快で、ソースプログラムを書いたり編集するための便利な機能を豊富に持っています。

もうひとつ、私がおすすめる方法は、マウスを使ってソフトや文書を表示しているアイコンをクリックするやり方ではなく、MS-DOSプロンプトやコマンドプロンプト、またはターミナルプログラム（Linuxの場合）をまず立ち上げて、その中で、エディタの名前と文書（Javaのソースファイル）の名前を、コマンドとして入力することです。たとえば次のように：

[コマンド入力の例]

```
C:\learnjava> hidemaru MyFirstJavaApp.java
```

上に示した例では、C:\learnjava> まだが、最初から画面に出ているプロンプト（入力催促記号）です。今現在あなたが、C:\learnjavaというディレクトリにいること＝あなたのカレントディレクトリがC:\learnjavaであることを、それは示しています。そしてその次のhidemaruが、エディタソフトの名前です*2。そして最後のMyFirstJavaApp.javaが、これからあなたが新たに書く、または編集する、ソースファイルの名前です。ソフトの名前とファイル名はスペースで区切ります。

なお、事前に、エディタソフトのあるディレクトリ（たとえばC:\Program Files\Hidemaru）を、PATHに含めておく必要があります。環境変数PATHの設定の仕方は0章に書かれていますから、その箇所を参考にしてください。PATHの設定はたとえば次のようになるでしょう：

[PATHの設定]

```
[Windows 98/Meの場合：AUTOEXEC.BATの中]
SET PATH=%PATH%;C:\j2se1.4.2_08\bin;C:\Program Files\Hidemaru
```

※1 Windows XPのコマンドプロンプトを勉強するならここがお勧め！
<http://pcweb.mycom.co.jp/special/2002/xpcommand/>。

※2 Windows用に「秀丸」という優れたエディタソフトが実在します… <http://hide.maruo.co.jp/software/hidemaru.html>。ただし私は個人的には、Windowsユーザにも、“エディタのボルシェ”と言われるVim（gvim）（<http://www.vim.org/download.php#pc>でgvim63.exeをダウンロード）を強力におすすめします。<http://www.kaoriya.net/doc/TextEditorCriticism.html>を見てみましょう。

```
[Linuxの場合：.bash_profileの中]
PATH=$PATH:/usr/java/j2se1.4.2_08/bin:/usr/local/Vim
export PATH
```

ご覧のように、コマンドを入力する方法は、アイコンの上をマウスでクリックする方法に比べると、かなり面倒です。しかしプログラミングを勉強していくときには、この面倒な方法に慣れたほうが、のちのちお得なのです。なぜかと言うと、コンパイラ `javac` もインタプリタ `java` も、やはりコマンドを入力して使うからです。

もっと広い意味で、あなたのコンピュータ技術が本格的に向上するためにも、コマンド入力に慣れるべきです。サーバなど、高度なプログラムも、コマンド入力によって動かすことが多いからです。

しかも本書では、Javaプログラミングの学習の最初の半年はIDEを使うな、と厳命しました(0章)から、コマンドの使い方を通じて、Javaの基本的な部分を理解していただかななくてはなりません。IDEという超親切なソフトを最初から使うと、逆に、Java言語とプログラミングの基礎がわからないまま年月が経過し、あとになって、基礎を理解していないことがあなたを困らせる結果になるからです。

そしてまた、ひとつのプログラムが完成するまであなたは、三種類のコマンドを何回も何回も繰り返します：

【書く・編集する→コンパイルする→実行する、の繰り返し】

```
C:¥learnjava> hidemaru MyFirstJavaApp.java
C:¥learnjava> javac MyFirstJavaApp.java
C:¥learnjava> java MyFirstJavaApp
```

— 何度も何度も繰り返す

上のコマンドの例では、最初の `hidemaru` コマンドでソースファイルの作成または編集(エディット(edit)作業)をやり、次の `javac` コマンドでそのソースファイルをコンパイルし、三つ目の `java` コマンドででき上がったプログラムを実行しています。実行結果にヘンなところがあったら、またまた上の三つの繰り返しをします。あるいは `javac` コンパイラがあなたのソースプログラムの書き方の欠陥に気づいてコンパイルを拒否したら、最後の `java` コマンドまで進まずに、上の二行(`hidemaru` コマンドによるエディットと `javac` コマンドによるコンパイル)の繰り返しになります。

ただし最初に `hidemaru MyFirstJavaApp.java` でエディタを起動したら、ソースファイルを書き直すたびに[保存]メニューをクリックしていると、エディタは終了しませんから、その後ずっとソースファイルの編集を続行できます。そこで、実

00

01

02

03

04

05

06

07

08

09

10