

# 「見ひらきで学べる Java プログラミング」 正誤表

古井陽之助, 神屋郁子, 下川俊彦, 合志和晃

2021/06/08

## 本資料について

本資料は「見ひらきで学べる Java プログラミング」(近代科学社, 2019) の正誤表です。本資料の著作権は著者が所有します。

## 改版履歴

- 2019/09/10 公開。
- 2019/10/25 2.4 節 (p.14) Ricty Diminished Discord の URL 変更を反映。
- 2020/05/26 初版第 2 刷での修正を反映。
- 2020/07/22 2.4 節 (p.14) Ricty Diminished Discord の URL 変更を反映。
- 2021/01/21 2.2 節 (p.11) 図 2.4 の誤りとその訂正について追記。
- 2021/06/08 5.2 節 (p.60) 図 5.1 の誤りとその訂正について追記。

©2019 古井陽之助, 神屋郁子, 下川俊彦, 合志和晃.

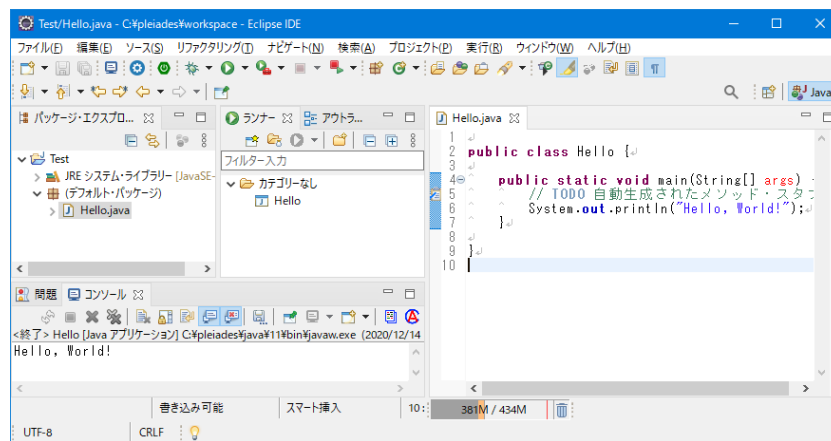
## 全体について

コマンドプロンプトにおいて javac コマンドを用いるときには、本資料の「2.6 節 (p.19)」の項目を参照してください。また、この修正の詳細については本資料の末尾にある付録 A を参照してください。

## 第2章 はじめてのプログラミング

(以下の項目は初版第2刷でも未修正)

2.2 節 (p.11) 図 2.4 図中左側のパッケージ・エクスプローラーに src というフォルダのアイコンが表示されていますが、この節の手順にしたがって操作すればこのアイコンは現れません。正しい図は次のとおりです。



2.4 節 (p.14) 側注 Ricty Diminished Discord の URL が変更されました。最新の URL には本書のサポートサイトからリンクしていますので、そちらでご確認ください。

(以下の項目は初版第2刷で修正済み)

2.3 節 (p.13) 上から3行目「今度はたくさん表示されていますが」の「今度は」を削除。

2.4 節 (p.14) 側注 Ricty Diminished Discord の URL が次のものに変更されました。

[http://www.yusa.lab.uec.ac.jp/~yusa/ricty\\_diminished.html](http://www.yusa.lab.uec.ac.jp/~yusa/ricty_diminished.html)

2.5 節 (p.16) 最後の行「(元と同じ行数に収めるため、段落は分けなくてOK)」は削除。

2.6 節 (p.19) コマンドプロンプトにおける javac の実行方法を

javac によるコンパイル処理

```
javac ファイル名.java
```

としていますが、このとき javac や java の実行結果に文字化けが見られる場合には次の 1., 2. のいずれかを試してください。

1. javac の文字コード指定を -encoding で UTF-8 に変更する。詳細は付録 A.1 を参照。

javac によるコンパイル時の文字コード指定

```
javac -encoding UTF-8 ファイル名.java
```

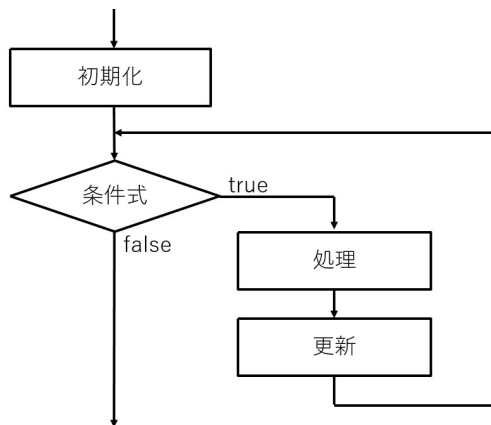
2. Java ソースファイルの文字コードをシフト JIS にして保存しなおす。詳細は付録 A.2 を参照。

文字コードの詳しい説明については本資料の付録 A を参照してください。

## 第5章 繰り返し

(以下の項目は初版第2刷でも未修正)

- 5.2 節 (p.60) 図 5.1 「更新」から出た矢印が「初期化」の前に接続するのは誤りです。正しい接続先は「初期化」と「条件式」の間です。正しい図は次のとおりです。



## 第9章 インスタンスメソッドと修飾子

(以下の項目は初版第2刷で修正済み)

- 9.2 節 (p.126) 最後の行「リスト 8.4 節 (8.3 節)」を「リスト 8.4 (8.3 節)」に修正。

- 9.10 節 (p.142) 章末問題 問2のリスト 9.15 の1行目にあるクラス名 Pr09\_2 を Pr09\_02 に修正。

```
1 public class Pr09_02 {
```

## 第10章 コンストラクタと多重定義

(以下の項目は初版第2刷で修正済み)

- 10.8 節 (p.158) 章末問題 問3の中央付近「フィールド fuel」を「フィールド fuel」に修正。

## 付録 A. JDK とテキストエディタによる開発における文字コード指定

### A.1 javac の文字コード指定を `-encoding` で UTF-8 に変更する

`javac` や `java` の実行結果において文字化けが起きた場合、`javac` によるコンパイル時に文字コードとして UTF-8 を指定すればおそらく解決するはずです。とりあえず対処するだけならこの方法が簡単です。

javac によるコンパイル時の文字コード指定

```
javac -encoding UTF-8 ファイル名.java
```

たとえば、図 2.11 の場合には次のようにしてください。

```
C:\Users\taro> javac -encoding UTF-8 Hello.java
```

### A.2 Java ソースファイルの文字コードをシフト JIS にして保存しなおす

別の解決方法です。テキストエディタでソースファイル（ファイル名.java）を開き、文字コードとしてシフト JIS を指定して、保存しなおしてください。

**メモ帳を用いる場合** テキストエディタとして Windows 付属のメモ帳を用いる場合には、次の手順でソースファイルの文字コードを修正することができます。

1. メモ帳でソースファイルを開く。
2. メニューから [ファイル]-[名前を付けて保存] を選択する。
3. 文字コードとして「ANSI」を選択する。
4. 保存する。

**TeraPad を用いる場合** テキストエディタとして TeraPad<sup>1</sup>を用いる場合には、次の手順でソースファイルの文字コードを修正することができます。

1. TeraPad でソースファイルを開く。
2. メニューから [ファイル]-[文字/改行コード指定保存] を選択する。
3. 文字コードとして「SHIFT-JIS」を選択する。
4. 保存する。

このようにして保存しなおしたソースファイルを `javac` でコンパイルするときには `-encoding UTF-8` は付けなくてください。

文字化けが起きるたびにソースファイルを保存しなおすのは面倒ですから、あらかじめテキストエディタの既定の文字コードもシフト JIS に変更しておくといでしょう。その方法はテキストエディタごとに異なります（できないこともあります）。TeraPad については TeraPad のインストール手順<sup>2</sup>を参照してください。

<sup>1</sup>TeraPad. <https://tera-net.com/>

<sup>2</sup>TeraPad のインストール手順. [https://mhirakijava.github.io/support/terapad\\_install.pdf](https://mhirakijava.github.io/support/terapad_install.pdf)

### A.3 解説

Java コンパイラ (javac) の文字コード指定と、実際の Java ソースファイルの文字コードとが食い違っていると、どこかで文字化けが生じます。

日本語 Windows でよく使われる文字コードはシフト JIS (厳密にいうとその一種の MS932) と UTF-8 です。一方、日本語 Windows 環境における javac の既定の文字コードはシフト JIS です。ソースファイルの文字コードも同じシフト JIS であれば文字化けは起きません。したがって、文字化けが起きるとすればその原因はソースファイルの文字コードが UTF-8 になっていることだろうと見当がつかます。

特に、Windows 10 バージョン 1903 (May 2019 Update) からは、メモ帳の既定の文字コードがシフト JIS から UTF-8 に変更されました。このため、メモ帳で Java ソースファイルを作ると文字化けが発生しやすいという状況になっています。

これに対処するには、javac の文字コードを UTF-8 に変える (上記の A.1) か、ソースファイルの文字コードをシフト JIS に変える (上記の A.2) かということになります。

ちなみに、文字化けの現れかたは 2 パターンあるようです。

- javac が文字コード変換できなくてエラーになる。
- javac は文字コードの食い違いに気づかないが、java の実行結果などに文字化けとして現れる。

### A.4 補足: 統合開発環境の場合

Eclipse のような統合開発環境は原則的にエディタとコンパイラの文字コードを合わせてくれます。統合開発環境を普通に使っているぶんには文字化けは起こりません。

もし統合開発環境の設定において文字コードを UTF-8 としてプログラミングをおこなっていて、特に問題がなければ、この設定を MS932 に変更したりソースファイルをシフト JIS で保存しなおしたりする必要はありません。