ま　え　が　き

本書はプログラミング初級者を対象として，読者がプログラミング，Java 言語，オブジェクト指向の基礎を理解し，活用できることを目的とした「アクティブラーニングで学ぶ Java プログラミングの基礎」シリーズの後編である。シリーズ前編の『アクティブラーニングで学ぶ Java プログラミングの基礎 1』では，変数，条件文，繰り返し文，配列などについて解説した。それらの内容は，Java 言語に限定したものではなく，手続き型プログラミング言語全般に共通する概念・構文であり，Java 言語を題材として取り上げてプログラミングの基礎について記述した。本書ではそれに引き続いて，クラス，継承といったオブジェクト指向言語である Java 言語の真髄を中心に学ぶことになる。

オブジェクト指向プログラミングが真価を発揮するのは，対象とするアプリケーションが大規模化し，持続的に展開される開発現場においてであり，数行のサンプルプログラムでそれを実感して理解するのは容易ではない。本書では，新しい概念を説明するためのサンプルプログラムを，それに留まることなく読者の理解を確実なものにするため「アクティブラーニング」を随所に取り入れた。アクティブラーニングとは書籍や教科書を単に読んで理解するだけでなく，読者が主体的・積極的に自ら物事を学ぶことであり，知識を定着させ，応用力を身につけるために重要な学びの手法である。また，「コーヒーブレイク」では初級者が陥りやすい誤りやより深い理解のための知識をできる限り平易に解説した。なお，「アクティブラーニング」の解答例を Web ページからダウンロードすることができるので，ぜひ自習することをお勧めする（詳細は p.138 参照）。

本書は比較的短いまとまりを章に区切って構成している。一つの章をほぼ 1 週間で理解し学習を進めることにより半年足らずで学習が完了する。また，必要に応じて「復習」を行う章を入れ，最後には理解度を測るための模擬試験を置いて，理解の確認と定着を図った。

本書の著者は，大学 1 年生の講義・演習を担当する若手の講師陣により構成されている。著者らは研究活動を通してプログラミング技術を現実の問題を解決するツールとして活用する研究者である一方で，担当講義の中で実際に学生に接して，理解が難しい概念をどう伝えたらよいかを日々工夫して解説をしている。本書はそういったノウハウを結集し，新しい学習者のために再構成を行ったものである。

読者は，本書を活用して，Java 言語，オブジェクト指向プログラミングを理解し，つぎのステップに進むべき基礎力を身につけていただきたい。また，自ら学ぶ力を習得することに

よって，プログラミングの世界での応用力を養ってくれることを期待している。

最後に，宇田隆哉講師をはじめとする東京工科大学の教員からなる本書の執筆陣一同の協力に心から感謝する。また，本書の出版にあたりコロナ社にも感謝の意を表する。

2014 年 8 月

大野　澄雄

●編者・執筆者一覧●

○編者

大野（おおの）　澄雄（すみお）　（東京工科大学）

○執筆者（執筆順）

宇田（うだ）　隆哉（りゅうや）（東京工科大学）：1，2，11，15，16 章

三橋（みつはし）　郁（かおる）　（東京工科大学）：3 章

柴田（しばた）　千尋（ちひろ）（東京工科大学）：4，5 章

荻谷（おぎや）　光晴（みつはる）（東京工科大学）：6，7，8 章

伊東（いとう）　拓（たく）　（東京工科大学）：9 章

井上（いのうえ）　亮文（あきふみ）（東京工科大学）：10 章

長名（おさな）　優子（ゆうこ）（東京工科大学）：12 章

渡邊（わたなべ）　紀文（のりふみ）（東京工科大学）：13 章

田所（たどころ）　裕康（ひろやす）（東京工科大学）：14 章

政倉（まさくら）　祐子（ゆうこ）（東京工科大学）：17 章

（2014 年 8 月現在）

目　　次

1. 『アクティブラーニングで学ぶ Java プログラミングの基礎 1』の復習

2. クラス

3. クラスの機能

4. クラス変数とクラスメソッド

5. クラスライブラリの利用（入門編）

6. クラスライブラリの利用（応用編）

7. 変数の代入

8. オブジェクト型の配列

9. 継　　承

10. パッケージ

11. 復 習 1

12. 抽象クラスとインタフェース

13. 例 外 処 理

14. 入　出　力

15. 復　習　2

16. 復　習　3

17. 模擬試験問題

1 『アクティブラーニングで学ぶ Java プログラミングの基礎 1』の復習

この章では Java プログラミングの基礎となる条件文，論理演算子，繰り返し文，配列について概略を説明するのみとし，詳細は割愛する。詳細については『アクティブラーニングで学ぶ Java プログラミングの基礎 1』を参照してほしい。

1.1 条　件　文

1.1.1 if～else 文

if 文は else 文と組み合わせて使用される。if～else 文の構造を以下に示す。

```
if(条件)  {
    文 1;
    文 2;
    …
}
else  {
    文 A;
    文 B;
    …
}
```

条件が満たされた場合，文 1，文 2，…が実行され，条件が満たされなかった場合には文 A，文 B，…が実行される。else 文を書かない状態でも実行は可能であるが，条件を満たさなかった場合にどうなるのかを明示しないままにしておくのはよくない。if 文を書くときには else 文も書くことを心がけよう。なお，else を書き忘れたのではないことが明示的にわかる場合には else は省略してもよいし，本書でもそのような書き方になっている。

if～else 文の例をプログラム 1-1 に示す。

プログラム 1-1 (if～else 文の例 (IfElseSample.java))

```
01 class IfElseSample {
02     public static void main(String[] args) {
03         int x = 1;
04         char answer = 'N';
05
06         if(x == 1) {
07             answer = 'Y';
08         }
```

int 型の変数 x を宣言すると同時に 1 を代入する。

char 型の変数 answer を宣言すると同時に N を代入する。

x の値が 1 であれば「Y」が，そうでなければ「N」が表示されるようになっている。

```
09          else {
10              answer = 'N';
11          }
12
13          System.out.println(answer);
14      }
15  }
```

アクティブラーニング **1.1**

プログラム 1-1 の 04，06，07，10 行目を変更し，x の値が 5 より大きければ 1 が，そうでなければ 0 が表示されるようにせよ。なお，answer を int 型の変数にすること。

［04 行目］

［06 行目］

［07 行目］

［10 行目］

1.1.2　if ～ else if ～ else 文

if ～ else if ～ else 文では，if～else 文に else if 文が加わる。if ～ else if ～ else 文の構造を以下に示す。

```
if(条件 1)  {
    文 1;
    文 2;
    …
}
else  if(条件 2)  {
    文 a;
    文 b;
    …
}
else  if(条件…)  {
    …
}
else  {
    文 A;
    文 B;
    …
}
```

条件 1 が満たされた場合，文 1，文 2，…が実行され，条件 1 が満たされなかった場合には，条件 2 が満たされるかどうかの判定に移る。条件 2 が満たされた場合，文 a，文 b，…が実行され，条件 2 が満たされなかった場合には，次の条件が満たされるかどうかの判定に

移る。else if 文は複数記述できる。どの条件も満たされなかった場合には，文 A，文 B，…が実行される。

if ～ else if ～ else 文の例をプログラム 1-2 に示す。

プログラム 1-2 (if ～ else if ～ else 文の例 (IfElseIfElseSample.java))

```
01 class IfElseIfElseSample {
02     public static void main(String[] args) {
03         int x = -1;
04         char answer = 'N';
05
06         if(x == 1) {
07             answer = 'Y';
08         }
09         else if(x < 0) {
10             answer = 'F';
11         }
12         else {
13             answer = 'N';
14         }
15
16         System.out.println(answer);
17     }
18 }
```

x の値が 1 であれば「Y」が表示される。

そうでなくてもし x の値が 0 未満であれば「F」が表示される。

そうでなければ「N」が表示される。

アクティブラーニング 1.2

プログラム 1-2 の 04，06，07，10，13 行目を変更し，x の値が 5 より大きければ 1 が，0 未満であれば −1 が，そうでなければ 0 が表示されるようにせよ。なお，answer を int 型の変数にすること。

［04 行目］

［06 行目］

［07 行目］

［10 行目］

［13 行目］

1.1.3 switch 文

switch 文では，条件の代わりに式が評価される。switch 文の構造を以下に示す。

```
switch(式)  {
case 値 1:
    文 1;
    文 2;
```

```
    …
    break;
case 値 2:
    文 a;
    文 b;
    …
    break;
default:
    文 A;
    文 B;
    …
    break;
}
```

式の値が値 1 となる場合,「case 値 1:」の行に処理が移り，文 1，文 2，…が実行される。式の値が値 2 となる場合,「case 値 2:」の行に処理が移り，文 a，文 b，…が実行される。case の行が現れるかどうかに関係なく，break 文が現れるまでこれらの処理は継続される。つまり，もし「case 値 2:」の直前の break 文がなければ，式の値が値 1 だとしても，文 1，文 2，…に続いて文 a，文 b，…も実行され,「default:」の行の直前の break 文で switch 文の処理が終了する。式の値と一致する case が存在しない場合には,「default:」の行に処理が移り，文 A，文 B，…が実行される。

switch 文の例をプログラム 1-3 に示す。

プログラム 1-3 (switch 文の例 (SwitchSample.java))

```
01 class SwitchSample {
02     public static void main(String[] args) {
03         int x = 1;
04         char answer = 'N';
05
06         switch(x) {
07         case 1:
08             answer = 'Y';
09             break;
10         case 2:
11             answer = 'Y';
12             break;
13         default:
14             answer = 'N';
15             break;
16         }
17         System.out.println(answer);
18     }
19 }
```

x の値を評価し，1 であれば 07 行目，2 であれば 10 行目，それ以外であれば 13 行目に処理が移る。

x の値が 1 であれば「Y」が表示される。

x の値が 2 であれば「Y」が表示される。

x の値がそれ以外であれば「N」が表示される。

アクティブラーニング **1.3**

プログラム 1-3 の 04，07，08，10，11，14 行目を変更し，x の値が 0 であれば 1 が，−1 であれば −1 が，それ以外であれば 0 が表示されるようにせよ。なお，answer を int 型の変数にすること。

［04 行目］

［07 行目］

［08 行目］

［10 行目］

［11 行目］

［14 行目］

この変更を行った場合，03 行目の x の値を 0 や −1 に変更し，出力がどのように変わるか説明せよ。

［出力がどのように変わるか］

1.2 論 理 演 算 子

Java でよく使われる論理演算子をつぎに示す。

(1) &&　両辺の論理積（AND）を演算する。両辺が真（true）であれば演算結果も真になる。

(2) ||　両辺の論理和（OR）を演算する。両辺のどちらか片方が真（true）であれば演算結果も真になる。

(3) !　右側の値を否定（NOT）する。真（true）であれば演算結果は偽（false）に，偽であれば真になる。

論理演算子の例をプログラム 1-4 に示す。

プログラム 1-4 (論理演算子の例 (LogicalOperatorSample.java))

```
01 class LogicalOperatorSample {
02     public static void main(String[] args) {
03         int x = 1;
04         int y = 0;
05         char answer = 'N';
06
```

```
07          if(x == 1 && y == 0) {
08              answer = 'Y';
09          }
10          else {
11              answer = 'N';
12          }
13
14          System.out.println(answer);
15      }
16  }
```

x の値が 1，なおかつ y の値が 0 であれば「Y」が，それ以外であれば「N」が表示されるようになっている。

アクティブラーニング **1.4**

プログラム 1-4 の 05，07，08，11 行目を変更し，x の値が 5 より大きいか，または y の値が 3 未満であれば 1 が，それ以外であれば 0 が表示されるようにせよ。なお，answer を int 型の変数にすること。

［05 行目］

［07 行目］

［08 行目］

［11 行目］

1.3 繰り返し文

1.3.1 for 文

for 文の構造を以下に示す。

```
for(式 1; 式 2; 式 3)  {
    文 1;
    文 2;
    …
}
```

式 1 は，for 文に処理が移った時点で実行される式である。式 2 は，for 文のループを行うかどうか判定する条件式である。条件が満たされた場合，ループ部分の文 1，文 2，…が実行される。式 3 は，for 文のループが毎回終了する直後に実行される式である。ループが終了するたびに式 2 の条件が評価され，条件が満たされた場合，ループ部分の文 1，文 2，…が実行される。

for 文の例をプログラム 1-5 に示す。

プログラム **1-5** (for 文の例 (ForSample.java))

```
01 class ForSample {
02     public static void main(String[] args) {
03         int sum = 0;
04
05         for(int i=0; i<10; i++) {
06             sum += i;
07         }
08
09         System.out.println(sum);
10     }
11 }
```

0 から 9 までの整数を sum に足し，その結果が表示される。

加算代入演算子が用いられており，「sum = sum+i;」と同じ意味を持つ。

アクティブラーニング **1.5**

プログラム 1-5 の 03，05，06，09 行目を変更し，1 から 5 までの整数の積が表示されるようにせよ。なお，05 行目では int 型の変数 product を宣言すること。

［03 行目］

［05 行目］

［06 行目］

［09 行目］

1.3.2 while 文

while 文の構造を以下に示す。

```
while(条件)  {
    文 1;
    文 2;
    …
}
```

条件が満たされた場合，ループ部分の文 1，文 2，…が実行される。ループが終了するたびに条件が評価され，条件が満たされた場合，ループ部分の文 1，文 2，…が実行される。

while 文の例をプログラム 1-6 に示す。

プログラム **1-6** (while 文の例 (WhileSample.java))

```
01 class WhileSample {
02     public static void main(String[] args) {
03         int sum = 0;
04         int i = 0;
05
06         while(i<10) {
```

0 から 9 までの整数を sum に足し，その結果が表示されるようになっている。

索　　引

—— 編者略歴 ——

1988 年　東京大学工学部電気工学科卒業
1993 年　東京大学大学院工学系研究科博士課程修了
　　（電子工学専攻），博士（工学）
1993 年　東京理科大学助手
1999 年　東京工科大学講師
2002 年　東京工科大学助教授
2010 年　東京工科大学教授
　　現在に至る

アクティブラーニングで学ぶ **Java プログラミングの基礎 2**

Foundations of Java Programming 2 — An Active Learning Approach —

2014 年 10 月 20 日　初版第 1 刷発行　★

検印省略

編　者　大野澄雄（おおのすみお）
発行者　株式会社　コロナ社
　　代表者　牛来真也
印刷所　三美印刷株式会社

112–0011　東京都文京区千石 4–46–10
発行所　株式会社　コロナ社
CORONA PUBLISHING CO., LTD.
Tokyo Japan
振替 00140–8–14844・電話(03)3941–3131(代)
ホームページ http://www.coronasha.co.jp

ISBN 978–4–339–02487–6（新井）　（製本：愛千製本所）
Printed in Japan