

# 新わかりやすい Java 入門編

## 第二版



単行本: 518 ページ  
出版社: 秀和システム (2019/11/4)

## 1. プログラミングと Java 言語

### 1.1 Java 言語とは

### 1.2 開発ツールの準備

1. Windows へのインストール
2. MacOS へのインストール

### 1.3 Eclipse による初めてのプログラム

1. プロジェクトとパッケージ
2. Eclipse を起動する
3. プロジェクト、パッケージ、クラスの作成
4. プログラムを書く
5. プログラムを実行する
6. Eclipse を終了する

### 1.4 実行の仕組み

### 1.5 まとめ

### 1.6 通過テスト

## 2.プログラムの書き方

### 2.1 プログラムの成り立ち

1. パッケージ文
2. クラス宣言
3. メソッド
4. 命令文

### 2.2 プログラムのフォーマット

1. インデントの付け方

### 2.3 コメント文

### 2.4 コンソールへの出力

### 2.5 まとめ

### 2.6 通過テスト

## 3.データ型と変数

### 3.1 データ型とリテラル

### 3.2 変数

1. 変数とは

### 3.3 JShell でプログラムを試す

### 3.4 変数名の付け方

### 3.5 まとめ

### 3.6 通過テスト

## 4.変数の使い方

### 4.1 変数宣言

1. 変数宣言の方法

### 4.2 いろいろな変数宣言

### 4.3 自動型変換

1. 整数から浮動小数点数への自動型変換
2. 整数リテラルの自動型変換
3. 小さな整数型から大きな整数型への自動型変換
4. char 型の整数型への自動型変換

### 4.4 まとめ

### 4.5 通過テスト

## 5.演算子と演算

- 5.1 基本的な演算子
- 5.2 インクリメント、デクリメント演算子
- 5.3 四則演算の演算子
- 5.4 文字列の連結
- 5.5 変数と文字列連結
- 5.6 キャスト演算子
- 5.7 代入演算子
- 5.8 複合代入演算子
- 5.9 まとめ
- 5.10 通過テスト

## 6.標準クラスのメソッドを使う

- 6.1 Math クラスのクラスメソッド
  - 1. 平方根の計算
  - 2. Math クラスの API
- 6.2 Input クラスのメソッド
  - 1. 数値の入力
  - 2. 文字列の入力
  - 3. Input クラスのその他の機能
- 6.3 文字列のメソッド
  - 1. 文字列のインスタンスメソッド
  
  - 2. インスタンスメソッドの使い方 (1)
  - 3. インスタンスメソッドの使い方 (2)
- 6.4 printf メソッド
  - 1. 浮動小数点数の出力
- 6.5 まとめ
- 6.6 通過テスト

## 7.配列と for 文

- 7.1 配列の作り方
  - 1. 配列の作成

- 2. 配列要素を表示する
- 7.2 for 文の書き方
- 7.3 カウンタ変数
- 7.4 配列要素の表示
  - 1. for 文で表示する方法
  - 2. タブを使って、横に並べて表示する方法
  - 3. プログラムの構造を図 (SPD) で表す
- 7.5 まとめ
- 7.6 通過テスト

## 8.for 文の使い方

- 8.1 配列の要素を合計する
  - 1. length ? 配列の要素数
  - 2. 配列要素の値を合計する
- 8.2 複数の for 文を使う
- 8.3 for 文の仕組み
  - 1. for 文を構成する要素
  - 2. 逆順に処理する for 文
  - 3. { }のない for 文
  - 4. for 文の要素の省略
- 8.4 ネストした for 文
- 8.5 拡張 for 文
- 8.6 まとめ
- 8.7 通過テスト

## 9.条件を書くための演算子

- 9.1 関係演算子と関係式
  - 1. 関係演算子
  - 2. 関係式の値
- 9.2 文字と文字列
  - 1. 文字の関係式
  - 2. 文字列の比較
- 9.3 論理演算子
  - 1. 論理演算子の使い方

- 2. 短絡評価
- 9.4 条件演算子
- 9.5 まとめ
- 9.6 通過テスト

## 10.while 文

- 10.1 while 文とは
- 10.2 While 文の使い方
- 10.3 電卓プログラム
- 10.4 While 文の書き方について
- 10.5 do
- 10.6 まとめ
- 10.7 通過テスト

## 11.if 文

- 11.1 if 文
- 11.2 if 文 (else の省略)
- 11.3 3つ以上に場合分けする
- 11.4 数値の範囲による場合分け
- 11.5 繰り返しと if 文
- 11.6 まとめ
- 11.7 通過テスト

## 12.分岐とジャンプによる制御

- 12.1 switch 文とは
- 12.2 コロンを使う switch 文
- 12.3 switch 式
- 12.4 break と continue
  - 1. 繰り返しを中止する
  - 2. 繰り返しで後続の処理をスキップする
  - 3. break、continue と多重ループ
- 12.5 まとめ
- 12.6 通過テスト

## 13. 配列

### 13.1 配列の作り方と使い方

1. 配列の作り方
2. その他の作成方法
3. 配列の仕組み
4. 配列要素へ値の代入

### 13.2 Arrays クラスの利用

1. 並び替え、検索、初期化、コピー、出力
2. ストリーム処理とラムダ式

### 13.3 「配列」の配列

1. 配列の配列とは
2. 拡張 for 文による配列の表示
3. 「配列」の配列を new で作成する
4. 配列要素へのアクセス

### 13.4 まとめ

### 13.5 通過テスト

## 14. ビット演算

### 14.1 ビット演算の準備

1. 2進数、10進数、16進数への相互変換
2. リテラルの表記法

### 14.2 シフト演算子

### 14.3 ビット演算子（ビットごとの論理演算）

1. 論理演算とは
2. ビット演算（ビットごとの論理演算）

### 14.4 まとめ

### 14.5 通過テスト

## 15. メソッド

### 15.1 メソッドの作り方

1. メソッドの作成
2. メソッドの使い方

### 15.2 戻り値や引数のないメソッド

## 15.3 引数と戻り値の受け渡し

1. 引数と戻り値の型変換
2. 配列の受け渡し
3. 参照の受け渡し

## 15.4 メソッドのオーバーロード

## 15.5 プログラムを分けるためのメソッド

## 15.6 ま と め

## 15.6 通過テスト

# 16.オブジェクトの作り方

## 16.1 オブジェクトを定義する

1. データの集まりをオブジェクトにする
2. 基本機能の組み込み

## 16.2 インスタンスを作る

1. インスタンスの作成
2. new 演算子とコンストラクタの働き

## 16.3 インスタンスメソッドを使う

1. メンバ参照演算子
2. ゲッターの使い方
3. セッターの使い方

## 16.4 まとめ

## 16.5 通過テスト

# 17.クラスの仕組み

## 17.1 メソッドを追加する

1. メソッドの作成
2. toString メソッド

## 17.2 クラスの仕組み

1. カプセル化
2. スタティックメンバとは
3. インスタンスメンバとは
4. インスタンスメンバとスタティックメンバの混在

## 17.3 コンストラクタの仕組み

1. コンストラクタのオーバーロード

2. this によるコンストラクタ呼び出し
3. デフォルトコンストラクタ

#### 17.4 まとめ

#### 17.5 通過テスト

## 18. クラスの継承

### 18.1 クラス図

### 18.2 クラスの継承

1. サブクラスを作る
2. 継承を確認する
3. サブクラスでの toString メソッドの自動生成

### 18.3 継承の規則

1. Is-A の関係
2. 継承できないクラス
3. 継承できないメンバ
4. super() の省略

#### 18.4 まとめ

#### 18.5 通過テスト

## 19. 継承関係

### 19.1 継承ツリー

1. 継承ツリーとは
2. Object クラス
3. コンストラクタの連鎖

### 19.2 アクセス修飾子

1. 基本的なアクセス修飾子
2. 継承にかかわるアクセス修飾子
3. クラスの要素とアクセス修飾子

### 19.3 複数のクラス

#### 19.4 まとめ

#### 19.5 通過テスト

## 20.参照とポリモーフィズム

### 20.1 参照とは

1. 参照と参照型
2. 参照の受け渡し
3. Java 言語の型

### 20.2 ポリモーフィズム (多態性)

1. 参照のアップキャスト
2. instanceof 演算子
3. ポリモーフィズム (多態性)

### 20.3 オーバーライド

1. オーバライドとは
2. オーバーライドメソッドの起動
3. オーバーライドとアップキャスト

### 20.4 サブクラスでのオーバーロード

### 20.5 ま と め

### 20.6 通 過 テ ス ト

## 21.抽象クラスとインタフェース

### 21.1 抽象クラス

1. オーバーライドとアップキャスト
2. 抽象クラスとは
3. 抽象クラスのクラス図
4. 抽象クラスにして継承する

### 21.2 インタフェース

1. インタフェースの作成
2. クラスへの実装
3. クラス図の書き方
4. インタフェースとポリモーフィズム
5. インタフェース間の継承
6. インタフェースの仕様の拡張

### 21.3 ま と め

### 21.4 通 過 テ ス ト

## 22. ファイル入出力と例外処理

### 22.1 ファイルに出力する

1. ファイル処理に使用する標準クラス
2. テキストファイルの出力

### 22.2 例外処理の書き方

### 22.3 ファイルから入力する

### 22.4 まとめ

### 22.5 通過テスト

## 補足資料

- 1 例外クラスの概要
- 2 演算子一覧表
- 3 ユニコード表

## 索引