

- ・暗記事項を参考にして予習ノートの作成
- ・予習範囲の熟読
- ・暗記事項の暗記
- ・教科書の演習問題を理解

上記を済ませてから練習問題に取り組んでください。

最終ページに解答を載せています。

## 予習範囲

P.11~P.18 の 2.3 節の前までです。ただし、ソースコード 2.6, 2.7, および, 2.8 は講義中に演習を行います。

## 暗記事項

【暗記事項 1】 から【暗記事項 28】 までは、すべて、「A: 正しい」が正答となります。予習確認テストでは表現を変えて「B: 間違っている」が答えになることもあります。

ただ覚えるだけでなく、教科書中の当該事項を説明した文章を読んで理解につとめてください。

- 【暗記事項 1】 クラスの定義には予約語 `class` を用いる
- 【暗記事項 2】 クラス名は予約語 `class` の後に書く
- 【暗記事項 3】 クラスの構成要素は、フィールド、メソッド、およびコンストラクタである。
- 【暗記事項 4】 変数を用いるためには宣言しないといけない
- 【暗記事項 5】 変数は基本データ型変数と参照型変数に分けられる
- 【暗記事項 6】 変数はフィールドと局所変数に分けられる
- 【暗記事項 7】 局所変数のことをローカル変数と呼ぶことがある
- 【暗記事項 8】 フィールドと局所変数は、宣言の位置によって区別できる
- 【暗記事項 9】 基本データ型変数ではない変数は参照型変数である。
- 【暗記事項 10】 基本データ型変数の型の種類は `boolean`, `byte`, `char`, `short`, `int`, `long`, `float`, `double` の 8 種類である。
- 【暗記事項 11】 `boolean` 型の変数には `true`(真)または `false`(偽)の値が入る。
- 【暗記事項 12】 `char` は非負の整数である。
- 【暗記事項 13】 `float` は 32bit の実数である。
- 【暗記事項 14】 基本データ型変数を宣言する書式は、「変数の型名 変数名;」である。
- 【暗記事項 15】 参照型変数を宣言する書式は、「クラス名 変数名;」である。
- 【暗記事項 16】 参照型変数の宣言における「クラス名」と  
基本データ型変数の宣言における「基本データ型変数の型名」をまとめて  
「型名」や「変数の型」と呼ぶ。
- 【暗記事項 17】 クラス名は通常、大文字から始める。
- 【暗記事項 18】 変数名は通常、小文字から始める。
- 【暗記事項 19】 参照型変数はオブジェクトのメモリ空間上の位置を記憶するためのものである。
- 【暗記事項 20】 オブジェクトは参照型変数を介してアクセスすることができる。
- 【暗記事項 21】 参照型変数の初期値は `null` であり、オブジェクトを参照していない状態を表す。
- 【暗記事項 22】 Java 言語における命名の慣習として、次のものがある：

「英単語などの意味のある単語を1つ以上組み合わせる」

【暗記事項 23】 Java 言語における命名の慣習として、次のものがある：

「2語以上の英単語をつなげる場合は、2語目以降の単語の先頭文字を大文字にする」

【暗記事項 24】 new 演算子を用いることによりオブジェクトを生成することができる。

【暗記事項 25】 new 演算子を用いたオブジェクト生成の書式は「new クラス名(引数リスト)」である  
(教科書3章に記載されています)。

【暗記事項 26】 new 演算子によるオブジェクトの生成直後、コンストラクタが呼び出される。

【暗記事項 27】 `Rectangle rectangle = new Rectangle(2,1,5,6);`

のように一行で書いた場合と、

```
Rectangle rectangle;
```

```
rectangle = new Rectangle(2,1,5,6);
```

のように2行で書いた場合は同じ意味となる。

【暗記事項 28】 `Rectangle rectangle = new Rectangle(2,1,5,6);`

のように書けば、

- 1) クラス `Rectangle` の参照型変数 `rectangle` を宣言し(左辺),
- 2) new 演算子によりクラス `Rectangle` のオブジェクトを生成し(右辺),
- 3) 参照型変数 `rectangle` に生成したオブジェクトの位置を記憶させる,  
ということの意味している。

## 2.1 節 クラス

クラスは、1つの機能・役割を担うように記述する



可読性が高まるだけでなく、プログラムの修正（変更・追加）が容易になる。

```
クラスの定義方法 語 1  
  
アクセス修飾子 class クラス名 2 {  
    _____ の宣言, _____ の定義 3  
}
```

(注) クラス名は通常, \_\_\_\_\_ から始める 17

## 2.2 節 基本データ型変数と参照型変数

変数を用いるためには \_\_\_\_\_ する必要がある 4

そして、変数は以下のように分類できる。

{ \_\_\_\_\_ 型変数 と \_\_\_\_\_ 型変数 5  
 \_\_\_\_\_ と \_\_\_\_\_ 変数 6 ( \_\_\_\_\_ 変数) 7

⇒ 「(2)として宣言された(1)」と表現される

基本データ型は 種類しかない 10

基本データ型変数では \_\_\_\_\_ 変数は参照型変数 9

(1)

(2)

宣言の \_\_\_\_\_ によって区別 8

### 基本データ型変数

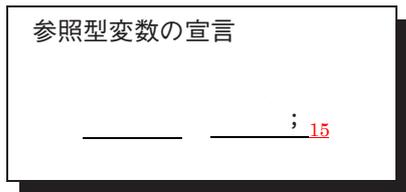
	型名	データの種類	範囲または大きさ
種類 10	boolean	真偽値	_____, _____ 11
	byte	整数	-128 ~ 127
	char	整数	0 ~ 65535 _____ の整数 12
	short	整数	-32768 ~ 32767
	int	整数	$-2^{31} \sim 2^{31} - 1$
	long	整数	$-2^{63} \sim 2^{63} - 1$
	float	実数	_____ ビット 13
	double	実数	64ビット

基本データ型変数とはデータを入れる箱のようなもの

```
基本データ型変数の宣言  
  
_____ _____ ; 14
```

## 参照型変数

オブジェクトのメモリ空間上の \_\_\_\_\_ を記憶しているもの 19



⇨ \_\_\_\_\_ は参照型変数を介して  
アクセスできる 20

## 変数の型

基本データ型変数の宣言「基本データ型変数の型名」

参照型変数 の宣言「クラス名」

⇨ 「\_\_\_\_\_」や「\_\_\_\_\_」と呼ぶ 16

宣言の左側

## 変数名の命名規則

- 変数名は通常, \_\_\_\_\_ から始める 18
- Java 言語における命名の慣習として,
  - 英単語などの \_\_\_\_\_ を1つ以上組み合わせる 22
  - 2単語以上の英語をつなげる時, 2語目以降の単語の先頭の文字は \_\_\_\_\_ にする 23

## 参照型変数とオブジェクト

参照状況を変化させる方法は3通り

(1) 新たにオブジェクトを生成する

\_\_\_\_\_ 演算子を用いる 24

オブジェクト生成の書式

new クラス名(引数リスト) 25

予約語

生成するオブジェクトが属するクラス

そのクラスのコンストラクタが呼び出される

1. オブジェクトを生成する
2. コンストラクタを呼び出す
3. 生成したオブジェクトの位置を返す

- new 演算子によるオブジェクトの生成直後, \_\_\_\_\_ が呼び出される 26
- new という文字列は予約語 (変数名やクラス名などに使えない) なので, new という単語がソースコードにあれば, そこでオブジェクトが生成されていると考えてよい

(2) 参照型変数にオブジェクトを参照させなくする

- 参照型変数の初期値は \_\_\_\_\_ であり, どのオブジェクトを参照していない状態を表す 21
- オブジェクトを参照している (オブジェクトの場所を覚えている) 参照型に, オブジェクトを参照させなくするには null を代入する

(例) `rectangle = null;`

(3) 他の参照型変数が参照しているオブジェクトを参照する

代入演算(=)を用いる (下のソースコードの3行目).

3行目の実行後, 参照型変数 `rectangle1` と `rectangle2` は同じオブジェクトを参照する

```
Rectangle rectangle1 = new Rectangle(2,1,5,6);  
Rectangle rectangle2;  
rectangle2 = rectangle1;
```

## 練習問題

確認テストも、この練習問題と同じ方法で、解答してください。最終ページに答えを載せています。

解答が複数ある場合は、ハイフン で繋いで答えること。

例: A と B と C を解答したい場合、A-B-C と解答欄に記入する。答えがない場合は-1 と解答すること。

行番号を解答するとき、左詰め0 は取ること

例: 001 行目を解答するときは 1 を解答すること

以下のソースコードについて【練習 01】から【練習 04】を解答しなさい

```
1 public class VarDefinition {
2     float var = 0;
3 }
```

【練習 01】 クラス名を答えなさい

【練習 02】 宣言された変数の個数を答えなさい

【練習 03】 変数の型名を答えなさい

【練習 04】 変数名を答えなさい

以下のソースコードについて【練習 05】から【練習 08】を解答しなさい

```
1 public class LocalVars {
2     public static void main(String[] args) {
3         VarDefinition varDefinition = new VarDefinition();
4         double a;
5     }
6 }
```

【練習 05】 このソースコードで定義されたクラスのクラス名を答えなさい

【練習 06】 宣言された変数の個数を答えなさい

以下、解答が複数ある場合は、変数が宣言されたソースコードの行数の昇順で、ハイフン繋ぎで答えること。

例: int-char-long

【練習 07】 宣言された変数のすべてについて、型名を答えなさい

【練習 08】 宣言された変数のすべてについて、変数名を答えなさい

以下のソースコードについて【練習 09】から【練習 13】を解答しなさい

```
01 public class ClassWithMethod {
02     byte data;
03     String name;
04     public ClassWithMethod(byte arg1, String arg2) {
05         data = arg1;
06         name = arg2;
07     }
08     public void printName() {
09         boolean flag = false;
10         System.out.println(name);
11         if (data == 2) {
12             flag = true;
13         }
14         System.out.println(flag);
15     }
16 }
```

```
15 }
16 }
```

【練習 09】 このソースコードで定義されたクラスのクラス名を答えなさい

【練習 10】 クラス内で宣言された変数の個数を答えなさい

【練習 11】 宣言された変数のすべてについて、宣言があった箇所の行番号を答えなさい

【練習 12】 宣言された変数のすべてについて、型名を答えなさい

【練習 13】 宣言された変数のすべてについて、変数名を答えなさい

以下のソースコードについて【練習 14】 から【練習 18】 を解答しなさい

```
01 public class Confusable {
02     long l1, l2 = 3;
03     public Confusable(long l1, long l2, long l3) {
04         this.l1 = l1;
05         this.l2 = l2;
06         System.out.println(l3);
07     }
08
09     public boolean compare(boolean l3) {
10         boolean ans = false;
11         if (l3 == true) {
12             if (l1 < l2) ans = true;
13         }
14         else {
15             if (l1 >= l2) ans = true;
16         }
17         return ans;
18         // 普通、 return (l1 < l2) == l3; と一行で書く。
19     }
20
21     public Confusable getInstance() {
22         Confusable confusable;
23         confusable = new Confusable(0, 1, 2);
24         return confusable;
25         // 普通、 return new Confusable(0, 1, 2); と一行で書く。
26     }
27
28     public static void main(String[] args) {
29         Confusable confusable = new Confusable(3, 2, 4);
30         confusable.compare(false);
31     }
32 }
```

【練習 14】 このソースコードで定義されたクラスのクラス名を答えなさい

【練習 15】 クラス内で宣言された変数の個数を答えなさい

【練習 16】 宣言された変数のすべてについて、宣言があった箇所の行番号を答えなさい

【練習 17】 宣言された変数のすべてについて、型名を答えなさい

【練習 18】 宣言された変数のすべてについて、変数名を答えなさい

以下の問いに答えなさい。ただし、クラス `LocalVars` は、本 PDF の 5 ページ目のものを用いる

(解答には**セミコロン**を記述しないこと)

【練習 19】クラス `LocalVars` の参照型変数 `localVars` を宣言しなさい

【練習 20】クラス `LocalVars` のオブジェクトを生成するコードを書きなさい

【練習 21】クラス `LocalVars` の参照型変数 `localVars` を宣言し、オブジェクトを生成し参照させなさい

この授業では、「厳密に」もしくは「まわりくどく」問題文を書いています。実習科目の課題 PDF では、下の  
ように、ざっくりと書いていることが多いです。下の問題は対応する問題と同じことを言っています。

【練習 19'】`LocalVars` 型の変数 `localVars` を宣言しなさい

【練習 20'】`LocalVars` のオブジェクトを生成する

【練習 21'】`LocalVars` の変数を宣言し初期化する

変数名に自由度を持たせている課題なら【練習 20'】のような表現で課題スライドに書いていても【練習 21】の  
ようなコードを書くことを期待されていることがあります。気を付けましょう

## 練習問題の解答

【練習 01】 VarDefinition

【練習 02】 1

【練習 03】 float

【練習 04】 var

【練習 05】 LocalVars

【練習 06】 3

【練習 07】 String[]-VarDefinition-double

【練習 08】 args-varDefinition-a

【練習 07】と【練習 08】について main メソッドの引数も含めていることに留意してください。また、変数名の varDefinition が小文字から始まっていて、型名（これは参照型変数なのでクラス名）の VarDefinition は大文字で始まっていることにも気を付けましょう

【練習 09】 ClassWithMethod

【練習 10】 5

【練習 11】 2-3-4-4-9

【練習 12】 byte-String-byte-String-boolean

【練習 13】 data-name-arg1-arg2-flag

【練習 14】 Confusable

【練習 15】 10

【練習 16】 2-2-3-3-3-9-10-22-28-29

【練習 17】 long-long-long-long-long-boolean-boolean-Confusable-String[]-Confusable

【練習 18】 l1-l2-l1-l2-l3-l3-ans-confusable-args-confusable

【練習 19】 LocalVars localVars

【練習 20】 new LocalVars()

【練習 21】 LocalVars localVars = new LocalVars()