

教科書 対応

全商情報処理検定 問題集

**J**ava

**2**級

©商業科の Java 教育を考える会

## 1 「教科書対応 全商情報処理検定問題集 Java 2級」の活用方法

- ・教科書で J a v a の基本的なプログラムを学び、その都度該当する問題を問題集で学習する。
- ・教科書には載っていない内容（応用問題）を解くことにより、一歩進んだ Java を理解する。
- ・検定範囲の筆記問題（データの集計と平均や線形探索など）を解くことにより、検定試験に合格できる力を身につける。

## 2 問題について

第1章 プログラミングの基礎

第1章 プログラミングの基礎

問題 1-1

教科書 P32

教科書の該当ページ数が書いています。

処理内容  
コンソールに「Hello Java!」と表示する。

実行結果  
Hello Java!

プログラム

```
1 class Hello {  
2     public static void main(String[] args) {  
3         [ ] ("Hello Java!");  
4     }  
5 }
```

[ ]

J a v a では大文字と小文字は区別されます。気をつけて記入しましょう。

## 3 連絡先

サポート情報や訂正などの情報は下記 W e b サイトをご覧ください。

「教科商業の工具箱」 <http://syougyou.ninpou.jp/>

誤植や問題の間違いなど何かありましたら下記アドレスまでご連絡下さい。

「商業科の J a v a 教育を考える会」事務局 [syougyoukyouiku@yahoo.co.jp](mailto:syougyoukyouiku@yahoo.co.jp)

# 教科書対応 全商情報処理検定問題集 J a v a 2 級

本書はプログラミングの J a v a 教科書および全商情報処理検定プログラミング部門 2 級 J a v a に対応した問題集です。全商検定第 2 級検定基準も完全に網羅しています。

<b>第 1 章</b>	<b>プログラミングの基礎</b>	<hr/>	<b>3</b>
	・データの出力	.....	3
	・変数	.....	4
	・データの入力	.....	7
<b>第 2 章</b>	<b>分岐と繰り返し処理</b>	<hr/>	<b>9</b>
	・二分岐	.....	9
	・多分岐	.....	14
	・繰り返し処理	.....	15
<b>第 3 章</b>	<b>クラスの基礎</b>	<hr/>	<b>20</b>
	・クラスの定義とメソッド	.....	20
	・オブジェクトの生成	.....	20
	・アクセス修飾子	.....	25
<b>第 4 章</b>	<b>配列の利用</b>	<hr/>	<b>28</b>
	・一次元配列	.....	28
	・参照型の配列	.....	32
<b>第 5 章</b>	<b>データの集計と平均</b>	<hr/>	<b>34</b>
<b>第 6 章</b>	<b>最大値と最小値の算出</b>	<hr/>	<b>42</b>
<b>第 7 章</b>	<b>配列を用いた集計</b>	<hr/>	<b>48</b>
<b>第 8 章</b>	<b>線形探索</b>	<hr/>	<b>54</b>
<b>第 9 章</b>	<b>ファイル入出力</b>	<hr/>	<b>60</b>
	・ファイル入力	.....	60
	・ファイル出力	.....	63

# 第1章 プログラミングの基礎

問題 1 - 1

教科書 P32

処理内容

コンソールに「Hello Java!」と表示する。

実行結果

Hello Java!

プログラム

```

1 class Hello {
2     public static void main(String[] args) {
3         _____("Hello Java!");
4     }
5 }

```

\_\_\_\_\_

問題 1 - 2

教科書 P33

処理内容

コンソールに「Hello」と表示し、次の行に「Java World!」と表示する。

実行結果

Hello  
Java World!

プログラム

```

1 class Hello {
2     public static void main(String[] args) {
3         _____("Hello");
4         _____("Java World!");
5     }
6 }

```

※2つの空欄部分は同一の内容である

\_\_\_\_\_

## 問題 1 - 3

教科書 P32

## 処理内容

コンソールに「Hello」と表示し、次の行に「Java World!」と表示する。

## 実行結果

```

Hello
Java World!

```

## プログラム

```

1  class Hello {
2      public static void main(String[] args) {
3          System.out.print("[        ]");
4      }
5  }

```

---

## 問題 1 - 4

教科書 P38

## 処理内容

数値を代入した変数を表示する。

## 実行結果

```

得点は 100 点です

```

## プログラム

```

1  class TokutenSample {
2      public static void main(String[] args) {
3          int tokuten = 100;
4          System.out.print("[                ]");
5      }
    }

```

---

## 問題 1 - 5

教科書 P38

## 処理内容

数値を代入した変数 a と b を合計して表示する。

## 実行結果

## プログラム

```
1 class Goukeiten {
2     public static void main(String[] args) {
3         int a = 100;
4         int b = 80;
5         System.out.print("合計点は" +  + "点です");
6     }
7 }
```

## 問題 1 - 6

2 級範囲

## 処理内容

変数 avg1 を avg2 にキャストして表示する。

## 実行結果

## プログラム

```
1 class Avg {
2     public static void main(String[] args) {
3         double avg1 = 73.45;
4         int avg2 = ;
5         System.out.print(avg2);
6     }
7 }
```

問題 1 - 7

2 級範囲

処理内容

文字列型や定数を表示する。変数 CPI は定数として定義する。

実行結果

山田さんの身長は 66.92913385826772 インチです

プログラム

```

1 class SintyouKeisan {
2     public static void main(String[] args) {
3         (1)   namae = "山田";
4         int cmSintyou = 170;
5         (2)   double CPI = 2.54;
6         double inchSintyou = cmSintyou / CPI;
7         System.out.print(namae + "さんの身長は");
8         System.out.print(inchSintyou + "インチです");
9     }
10  }
    
```

(1)	_____
(2)	_____

問題 1 - 8

教科書 P94

処理内容

指定した書式で表示する。

実行結果

単価は¥580  
売上は 3,200,000

プログラム

```

1 class UriageOutput {
2     public static void main(String[] args) {
3         int tanka = 580;
4         int uriage = 3200000;
5         System.out.println("単価は¥¥" + (1) );
6         System.out.(2) ("売上は%,d", uriage);
7     }
8 }
    
```

(1)	_____
(2)	_____

問題 1 - 9

教科書 P40

処理内容

キーボードからデータを入力し、表示する。

実行結果

```
名前を入力してください：
yamada
体重を入力してください：
65
あなたの名前は yamada
あなたの体重は 65
```

プログラム

```

1 import java.util.Scanner;
2 class ScannerSample {
3     public static void main(String[] args) {
4         (1) sc = new (1)(System.in);
5         System.out.println("名前を入力してください：");
6         String namae = (2);
7         System.out.println("体重を入力してください：");
8         int weight = (3);
9         System.out.println("あなたの名前は" + namae);
10        System.out.println("あなたの体重は" + weight);
11        sc.close();
12    }
13 }
```

(1)	
(2)	
(3)	



問題 1 - 10

教科書 P40

処理内容

キーボードからデータを入力し、定数との計算結果を表示する。

実行結果

直径を入力してください  
20  
円周は 62.80 です

プログラム

```

1  import java.util.Scanner;
2  class Circle {
3      public static void main(String[] args) {
4          final (1) = 3.14;
5          (2) = new Scanner(System.in);
6          System.out.println("直径を入力してください");
7          int tyokkei = sc.nextInt();
8          double ensyu = tyokkei * PI;
9          System.out.printf("円周は%.2f です", (3));
10         sc.close();
11     }
12 }
```

(1)	
(2)	
(3)	

## 第2章 分岐と繰り返し処理

### 問題 2 - 1

教科書 P52

#### 処理内容

- ・得点を基準点（70点）と比較して、合格か不合格か表示する。
- ・基準点より低ければ不合格、基準点以上であれば合格である。

#### 実行結果

不合格です。

#### プログラム

```

1  class IfSample {
2      public static void main(String[] args) {
3          int kijyun = 70, tokuten = 60;
4          if( ) {
5              System.out.print("不合格です");
6          } else {
7              System.out.print("合格です");
8          }
9      }
10 }

```

### 問題 2 - 2

教科書 P52

#### 処理内容

- ・入力された得点を最高点（100点）と比較して、同じ点数であれば saikouKaisu の値を 1 プラスする。

#### プログラム

```

1  import java.util.Scanner;
2  class IfSample {
3      public static void main(String[] args) {
4          int saikou = 100, saikouKaisu = 0;
5          Scanner sc = new Scanner(System.in);
6          System.out.println("得点を入力して下さい");
7          int tokuten = sc.nextInt();
8          if( ) {
9              saikouKaisu++;
10         }
11         sc.close();
12     }
13 }

```







## 問題 2 - 6

教科書 P57

## 処理内容

変数  $x$  と  $y$  と  $z$  の 3 つの値が同じかどうか判定する。

## 実行結果

(この問題では 3 つが違う値なので何も表示されない)

## プログラム

```

1 class LogicalSample {
2     public static void main(String[] args) {
3         int x = 5, y = 5, z = 4;
4         if(  ){
5             System.out.print("3 つが同じ値です");
6         }
7     }
8 }

```

## 問題 2 - 7

教科書 P57

## 処理内容

国語か英語の点のいずれかが 80 点以上なら、合格と表示する。

## 実行結果

## プログラム

```

1 class LogicalSample {
2     public static void main(String[] args) {
3         int japanese = 85, english = 75;
4         if(  ){
5             System.out.print("合格");
6         }
7     }
8 }

```

## 問題 2 - 8

教科書 P58

## 処理内容

キーボードから入力した区分に応じて在庫数を増減する。

区分 1	在庫数を 2 プラスする
区分 2	在庫数を 1 プラスする
区分 3 か 4	在庫数を 1 マイナスする
上記以外	区分エラーと表示する

## プログラム

```

1  import java.util.Scanner;
2  class CategorySample {
3      public static void main(String[] args) {
4          int zaiko = 10, kubun;
5          Scanner sc = new Scanner(System.in);
6          System.out.println("区分を入力して下さい");
7          kubun = sc.nextInt();
8          if( (1) ){
9              System.out.println("区分エラーです");
10             }else{
11                 (2) {
12                     case 1:
13                         zaiko = zaiko + 2;
14                         (3) ;
15                     case 2:
16                         zaiko = zaiko + 1;
17                         (3) ;
18                     (4) :
19                         zaiko = zaiko - 1;
20                 }
21             }
22             sc.close();
23         }
24     }

```

(1)	
(2)	
(3)	
(4)	





## 問題 2 - 11

処理内容

掛け算の九九を表示する。

実行結果

1	2	3	4	5	6	7	8	9
2	4	6	8	10	12	14	16	18
3	6	9	12	15	18	21	24	27
4	8	12	16	20	24	28	32	36
5	10	15	20	25	30	35	40	45
6	12	18	24	30	36	42	48	54
7	14	21	28	35	42	49	56	63
8	16	24	32	40	48	56	64	72
9	18	27	36	45	54	63	72	81

プログラム

```
1  class Multiplication {
2      public static void main(String[] args) {
3          for(int i = 1; i < 10; i++){
4              _____ {
5                  System.out.print((i * j) + "%t");
6              }
7              System.out.println();
8          }
9      }
10 }
```





問題 2 - 13

教科書 P65

処理内容

あらかじめ設定されている正解の数字（今回の場合は53）を当てるゲーム。正解の数字より大きい小さいかを表示し、一致すればプログラムを終了する。

実行結果

```

数字を入力して下さい：
70
大きいです！数字を入力して下さい：
40
小さいです！数字を入力して下さい：
55
大きいです！数字を入力して下さい：
53
正解です
    
```

プログラム

```

1  import java.util.Scanner;
2  public class Guess {
3      public static void main(String[] args) {
4          Scanner sc = new Scanner(System.in);
5          int number = 53;
6          int answer;
7          System.out.println("数字を入力して下さい：");
8          while((answer = sc.nextInt()) != (1) ){
9              if((2) ){
10                 System.out.print("大きいです！");
11             }else if((3) ){
12                 System.out.print("小さいです！");
13             }
14             System.out.println("数字を入力して下さい：");
15         }
16         System.out.println("正解です");
17         sc.close();
18     }
19 }
    
```

(1)	
(2)	
(3)	

## 問題 2 - 14

教科書 P71

処理内容

0~9を表示する。ただし、5は表示しない。

実行結果

0 1 2 3 4 6 7 8 9

プログラム

```

1 class Escape {
2     public static void main(String[] args) {
3         for(int i = 0; i < 10; i++){
4             if(i == 5){
5                 ;
6             }
7             System.out.print(i + " ");
8         }
9     }
10 }
```

## 問題 2 - 15

教科書 P71

処理内容

0~9を表示する。ただし、変数iが5になると、表示をせずに繰り返し処理を抜ける。(結果的に0~4までしか表示されないプログラムとなる)

実行結果

0 1 2 3 4

プログラム

```

1 class Escape {
2     public static void main(String[] args) {
3         for(int i = 0; i < 10; i++){
4             if(i == 5){
5                 ;
6             }
7             System.out.print(i + " ");
8         }
9     }
10 }
```





**問題 3 - 4**

教科書 P79

処理内容

コンストラクタを利用しフィールドに値を設定し、表示する。

実行結果

<pre>ペン 110</pre>
-------------------

プログラム

```

1  class Item {
2      String name;
3      int price;
4      Item( (1) ){
5          (2) .name = name;
6          (2) .price = price;
7      }
8  }

1  class ItemSample {
2      public static void main(String[] args) {
3          Item pen = new Item("ペン", 110);
4          System.out.println(pen.name);
5          System.out.println(pen.price);
6      }
7  }

```

(1)	
(2)	

問題 3 - 5

教科書 P94

処理内容

Item クラスをインスタンス化し、フィールドに値を代入し、メソッドを呼び出し表示する。また、インスタンスから売上高を受け取り、売上高を表示する。

実行結果

```
商品名：ペン
単価：110
売上数量：200
売上高は 22000 です
```

プログラム

```

1 class Item {
2     String namae;
3     int tanka;
4     int urisuu;
5     int uriage;
6     void showItem(){
7         System.out.println("商品名：" + namae);
8         System.out.println("単価：" + tanka);
9         System.out.println("売上数量：" + urisuu);
10    }
11    String showSales(){
12        uriage = tanka * urisuu;
13        (1) "売上高は" + uriage + "です";
14    }
15 }
```

```

1 class SalesSample {
2     public static void main(String[] args) {
3         String output;
4         Item pen = new Item();
5         pen.namae = "ペン";
6         pen.tanka = 110;
7         pen.urisuu = 200;
8         (2);
9         output = (3);
10        System.out.println(output);
11    }
12 }
```

(1)	
(2)	
(3)	



問題 3 - 6

教科書 P75

処理内容

インスタンスメソッドを利用する。

実行結果

売上高は 800

プログラム

```

1 class SalesTest1 {
2     int sales;
3     void showSales(){
4         System.out.println("売上高は" + sales);
5     }
6     public static void main(String[] args) {
7         SalesTest1 textbook = ;
8         textbook.sales = 800;
9         textbook.showSales();
10    }
11 }

```

Empty box for answer

問題 3 - 7

教科書 P92

処理内容

クラスメソッドを利用する。

実行結果

売上高は 800

プログラム

```

1 public class SalesTest2 {
2     static int sales;
3     static void showSales(){
4         System.out.println("売上高は" + sales);
5     }
6     public static void main(String[] args) {
7         .sales = 800;
8         .showSales();
9     }
10 }

```

※2つの空欄部分は同一の内容である

Empty box for answer



**問題 3 - 9**

教科書 P92

処理内容

Player クラスと Enemy クラスのインスタンスを生成し、レベルの高い方が勝ちと表示する。同じレベルの場合は引き分けと表示する。

実行結果

上田の勝ちです

プログラム

```

1 public class Player {
2     String name;
3     int level;
4     public Player(String name, int level){
5         this.name = name;
6         (1);
7     }
8 }

1 public class Enemy {
2     String name;
3     int level;
4     public Enemy(String name, int level){
5         this.name = name;
6         (1);
7     }
8 }

1 public class GameMain {
2     public static void main(String[] args) {
3         Player (2) = new Player("上田", 5);
4         Enemy enemy = new Enemy("スライム", 3);
5         if(player.level > enemy.level){
6             System.out.println((3) + "の勝ちです");
7         }else if(player.level < enemy.level){
8             System.out.println((4) + "の勝ちです");
9         }else{
10            System.out.println("引き分けです");
11        }
12    }
13 }

```

(1)	
(2)	
(3)	
(4)	



# 第4章 配列の利用

問題 4 - 1

教科書 P98

処理内容

- String 型の一次元配列を宣言し、その後配列オブジェクトを作成する。
- 作成した配列に、以下の文字列を代入する。

(0)	(1)	(2)
katou	kida	kuroda

- 配列の 2 つ目の要素を出力する。

実行結果

kida

プログラム

```

1 public class ArraySample {
2     public static void main(String[] args) {
3         String[] name;
4         name = (1);
5         (2) = "katou";
6         (3) = "kida";
7         (4) = "kuroda";
8         System.out.println(name[1]);
9     }
10 }

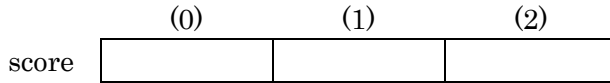
```

(1)	
(2)	
(3)	
(4)	

問題 4 - 2

処理内容

配列 score に 3 人分の点数を代入し、平均点を計算し表示する。



実行結果

```

3 人の得点を入力して下さい
50
60
90
平均点 : 66
    
```

プログラム

```

1  import java.util.Scanner;
2  public class ArraySample {
3      public static void main(String[] args) {
4          int[] score =  (1) ;
5          int sum, average;
6          Scanner sc = new Scanner(System.in);
7          System.out.println("3 人の得点を入力して下さい");
8          for(int i = 0;  (2) ; i++){
9               (3)  = sc.nextInt();
10         }
11         sum = score[0] + score[1] + score[2];
12          (4)  = sum / 3;
13         System.out.println("平均点 : " + average);
14         sc.close();
15     }
16 }
    
```

(1)	<input type="text"/>
(2)	<input type="text"/>
(3)	<input type="text"/>
(4)	<input type="text"/>

## 問題 4 - 3

教科書 P105

## 処理内容

配列 code と配列 price に以下のデータを代入し、それらを表示する。

コード	101	102	103
価格	380	240	520

## 実行結果

```
101:380 円
102:240 円
103:520 円
```

## プログラム

```

1 public class ArraySample {
2     public static void main(String[] args) {
3         int[] code = {101, 102, 103};
4         int[] price = (1);
5         for(int i = 0; i < code.length; i++){
6             System.out.println((2) + ":" + (3) + "円");
7         }
8     }
9 }

```

(1)	
(2)	
(3)	

## 問題 4 - 4

処理内容

配列 `data` の初期値として以下の値を設定し、合計を計算し表示する。

	(0)	(1)	(2)	(3)	(4)	(5)
data	50	30	40	25	60	80

実行結果

sum:285

プログラム

```

1 public class SumSample {
2     public static void main(String[] args) {
3         int[] data = {50,30,40,25,60,80};
4         int (1);
5         for(int i = 0; i < data.(2); i++){
6             sum += (3);
7         }
8         System.out.println("sum:" + sum);
9     }
10 }

```

(1)	
(2)	
(3)	

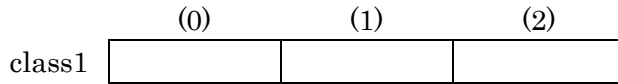


問題 4 - 5

応用問題

処理内容

Student インスタンスを 3 つ生成し、参照型の変数 class1 にそれらインスタンスの参照を代入する。



※それぞれの要素に Student インスタンスの参照を代入

実行結果

佐藤 : 18 才  
 清水 : 22 才  
 鈴木 : 16 才

プログラム

```

1 public class Student {
2     private String name;
3     private int age;
4     public Student(String name, int age){
5         this.name = name;
6         this.age = age;
7     }
8     public void show(){
9         System.out.println(name + " : " + age + "才");
10    }
11 }

1 public class ReferenceArray {
2     public static void main(String[] args) {
3         (1) class1 = new Student[3];
4         class1[0] = new Student("佐藤", 18);
5         class1[1] = new Student("清水", 22);
6         class1[2] = new Student("鈴木", 16);
7         for(int i = 0; i < class1.length; i++){
8             (2).show();
9         }
10    }
11 }
    
```

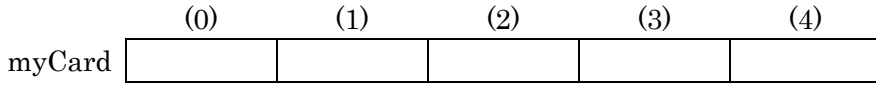
(1)	
(2)	

問題 4 - 6

応用問題

処理内容

Player クラスのインスタンスが Card クラスのインスタンスを配列で保持する。



※それぞれの要素に Card インスタンスの参照を代入

実行結果

```

カード名：シマウマ 強さ：230
カード名：ライオン 強さ：400
カード名：カバ 強さ：360
    
```

プログラム

```

1 public class Card {
2     private String name;
3     private int power;
4     public Card(String name, int power){
5         this.name = name;
6         this.power = power;
7     }
8     public void show(){
9         System.out.println("カード名：" + name + " 強さ：" + power);
10    }
11 }

1 public class Player {
2     private String name;
3     private int cardNumber = 0;
4     private Card[] myCard = new Card[5];
5     public Player(String name){
6         this.name = name;
7     }
8     public void getCard(String name, int power){
9         myCard[cardNumber] = new Card(name, power);
10        cardNumber++;
11    }
12    public void myCardShow(){
13        for(int i = 0; i < cardNumber; i++){
14            myCard[i].show();
15        }
16    }
17 }

1 public class GameMain {
2     public static void main(String[] args) {
3         Player player = new Player("山本");
4         player.getCard("シマウマ", 230);
5         player.getCard("ライオン", 400);
6         player.getCard("カバ", 360);
7         player.myCardShow();
8     }
9 }
    
```

(1)	
(2)	
(3)	

# 第5章 データの集計と平均

問題 5 - 1

2 級範囲

## 処理内容

売上高を入力し合計を計算する。売上高に 999999 が入力されると、合計を出力する。

## 実行結果

```

売上高の入力：300
売上高の入力：500
売上高の入力：850
売上高の入力：999999
合計：1650
    
```

## プログラム

```

1 import java.util.Scanner;
2 public class Syuukei {
3     public static void main(String[] args) {
4         int uriage;
5          (1);
6         Scanner sc = new Scanner(System.in);
7         System.out.print("売上高の入力：");
8         uriage = sc.nextInt();
9         while(uriage != 999999){
10             goukei =  (2);
11             System.out.print("売上高の入力：");
12             uriage = sc.nextInt();
13         }
14         System.out.printf("合計：%d", goukei);
15         sc.close();
16     }
17 }
    
```

(1)	
(2)	

問題 5 - 2

2 級範囲

処理内容

売上高を入力し合計を計算する。売上高に 999999 が入力されると、合計と平均を出力する。

実行結果

```

売上高の入力：300
売上高の入力：500
売上高の入力：850
売上高の入力：999999
合計：1650
平均：550
    
```

プログラム

```

1  import java.util.Scanner;
2  public class Syuukei {
3      public static void main(String[] args) {
4          int uriage;
5          int goukei = 0;
6          int kensu = 0;
7          Scanner sc = new Scanner(System.in);
8          System.out.print("売上高の入力：");
9          uriage = sc.nextInt();
10         while(uriage != 999999){
11              (1);
12              (2);
13             System.out.print("売上高の入力：");
14             uriage = sc.nextInt();
15         }
16         System.out.printf("合計：%d¥r", goukei);
17         int heikin = goukei / kensu;
18         System.out.printf("平均：%d", heikin);
19         sc.close();
20     }
21 }
    
```

(1)	
(2)	

## 問題 5 - 3

## 2 級範囲

処理内容

売上高を入力し合計を計算する。売上高に 999999 が入力されると、合計と平均を出力する。

実行結果

```
売上高の入力：300
売上高の入力：500
売上高の入力：850
売上高の入力：999999
合計：1650
平均：550
```

プログラム

```
1 import java.util.Scanner;
2 public class Syuukei {
3     public static void main(String[] args) {
4         int uriage;
5         int goukei = 0;
6         int kensu = 0;
7         Scanner sc = new Scanner(System.in);
8         do{
9             System.out.print("売上高の入力：");
10            uriage = sc.nextInt();
11            if( (1) )
12                break;
13            goukei = goukei + uriage;
14            kensu = kensu + 1;
15        }while( (2) );
16        System.out.printf("合計：%d¥r", goukei);
17        int heikin = goukei / kensu;
18        System.out.printf("平均：%d", heikin);
19        sc.close();
20    }
21 }
```

(1)	
(2)	

問題 5 - 4

2 級範囲

処理内容

売上単価と売上数量を入力し、売上金額を計算し表示する。売上数量が 10 個以上の場合は売上金額を 1 割引きにする。また、売上単価に 0 が入力されると売上金額合計を表示しプログラムを終了する。

実行結果

```

売上単価を入力：100
売上数量を入力：8
売上金額は 800
売上単価を入力：200
売上数量を入力：10
売上金額は 1,800
売上単価を入力：0
売上金額合計は 2,600
    
```

プログラム

```

1  import java.util.Scanner;
2  public class SumSample {
3      public static void main(String[] args) {
4          int tanka, suu, kingaku, goukei = 0;
5          Scanner sc = new Scanner(System.in);
6          System.out.print("売上単価を入力：");
7          tanka = sc.nextInt();
8          while(tanka != 0){
9              System.out.print("売上数量を入力：");
10             suu = sc.nextInt();
11             if(suu >= 10)
12                 kingaku = (int) (  );
13             else
14                 kingaku = tanka * suu;
15                  ;
16             System.out.printf("売上金額は %,d¥r", kingaku);
17             System.out.print("売上単価を入力：");
18             tanka = sc.nextInt();
19         }
20         System.out.printf("売上金額合計は %,d", goukei);
21         sc.close();
22     }
23 }
    
```

(1)	
(2)	

問題 5 - 5

2 級範囲

処理内容

世帯当たりの人数と 1 か月の食費を入力し、一人当たりの 1 か月の食費を計算し求める。なお、家族の人数に 99 が入力されると結果を出力し終了する。

実行結果

```

家族の人数  3
1 か月の食費 15000
家族の人数  4
1 か月の食費 14000
家族の人数  99
一人当たりの 1 か月の食費  4142
    
```

プログラム

```

1  import java.util.Scanner;
2  public class CountSample {
3      public static void main(String[] args) {
4          int nin, ninkei = 0, syokuhi, syokuhikei = 0;
5          int syokuhiAvg;
6          boolean flg = (1);
7          Scanner sc = new Scanner(System.in);
8          while(flg){
9              System.out.print("家族の人数 ");
10             nin = sc.nextInt();
11             if(nin == 99){
12                 flg = false;
13             }else{
14                 (2);
15                 System.out.print("1 か月の食費 ");
16                 syokuhi = sc.nextInt();
17                 syokuhikei = (3);
18             }
19         }
20         syokuhiAvg = syokuhikei / ninkei;
21         System.out.print("一人当たりの 1 か月の食費 " + syokuhiAvg);
22         sc.close();
23     }
24 }
    
```

(1)	
(2)	
(3)	

問題 5 - 6

2 級範囲

処理内容

利用場所コード 1 番の利用時間合計を求める。利用時間と分は 4 桁で入力する。

例) 3 時間 45 分利用 → 0345 と入力

実行結果

```

利用場所コードを入力して下さい：1
利用時間と分を 4 桁で入力して下さい：0345
利用場所コードを入力して下さい：2
利用時間と分を 4 桁で入力して下さい：1015
利用場所コードを入力して下さい：0
利用場所 1 番の利用時間合計 225 分
    
```

プログラム

```

1 import java.util.Scanner;
2 public class Jikan {
3     public static void main(String[] args) {
4         int code, jikan, ji, hun, hunTotal = 0;
5         Scanner sc = new Scanner(System.in);
6         System.out.print("利用場所コードを入力して下さい：");
7         code = sc.nextInt();
8         while(code != 0){
9             System.out.print("利用時間と分を 4 桁で入力して下さい：");
10            jikan = sc.nextInt();
11            if( (1) ){
12                ji = jikan / 100;
13                hun = jikan - (2);
14                hunTotal = hunTotal + (3) + hun;
15            }
16            System.out.print("利用場所コードを入力して下さい：");
17            code = sc.nextInt();
18        }
19        System.out.printf("利用場所 1 番の利用時間合計 %d 分", hunTotal);
20        sc.close();
21    }
22 }
    
```

(1)	
(2)	
(3)	



## 問題 5 - 7

## 2 級範囲

## 処理内容

点数（0 点以上 100 点以下）を入力し、70 点以上なら「合格です」と表示し、そうでないなら「不合格です」と表示する。点数に 999 を入力すると処理を終了し、合格率を表示する。

## 実行結果

```
点数を入力して下さい 80
合格です
点数を入力して下さい 60
不合格です
点数を入力して下さい 95
合格です
点数を入力して下さい 999
合格率 66%
```

## プログラム

```
1 import java.util.Scanner;
2 public class Gouhi {
3     public static void main(String[] args) {
4         int score, ninzu = 0, goukaku = 0, goukakuritu;
5         Scanner sc = new Scanner(System.in);
6         System.out.print("点数を入力して下さい ");
7         score = sc.nextInt();
8         while(score != 999){
9             if(score >=0 && score <=100){
10                (1);
11                if(score >= 70){
12                    System.out.println("合格です");
13                    (2);
14                }else{
15                    System.out.println("不合格です");
16                }
17            }else{
18                System.out.println("得点が正しくありません");
19            }
20            System.out.print("点数を入力して下さい ");
21            score = sc.nextInt();
22        }
23        (3) = goukaku * 100 / ninzu;
24        System.out.print("合格率 " + goukakuritu + "%");
25        sc.close();
26    }
27 }
```

(1)	_____
(2)	_____
(3)	_____







問題 6 - 3

2 級範囲

処理内容

名前と得点を入力し、名前に「end」が入力されると入力処理を終了し、最高点を取得した人の名前を表示する。なお、同点の場合は先に最高点を取得した人の名前を表示する。

実行結果

名前を入力して下さい 橋本
得点を入力して下さい 80
名前を入力して下さい 平岡
得点を入力して下さい 70
名前を入力して下さい 藤井
得点を入力して下さい 80
名前を入力して下さい end
最高点は 橋本 さんです

プログラム

```
1 import java.util.Scanner;
2 public class SaikouNamae {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         int score, max = 0;
6         String namae, maxNamae = null;
7         System.out.print("名前を入力して下さい ");
8         namae = sc.next();
9         while(!(namae.equals("end"))){
10            System.out.print("得点を入力して下さい ");
11            score = sc.nextInt();
12            if( (1) ){
13                max = score;
14                (2) ;
15            }
16            System.out.print("名前を入力して下さい ");
17            namae = sc.next();
18        }
19        System.out.print("最高点は " + maxNamae + " さんです");
20        sc.close();
21    }
22 }
```

Table with 2 rows and 1 column for answers (1) and (2).

問題 6 - 4

2 級範囲

処理内容

名前と得点を入力し、名前に「end」が入力されると入力処理を終了し、最高点を取得した人の名前を表示する。なお、同点の場合は後に最高点を取得した人の名前を表示する。

実行結果

名前を入力して下さい 橋本  
 得点を入力して下さい 80  
 名前を入力して下さい 平岡  
 得点を入力して下さい 70  
 名前を入力して下さい 藤井  
 得点を入力して下さい 80  
 名前を入力して下さい end  
 最高点は 藤井 さんです

プログラム

```

1 import java.util.Scanner;
2 public class SaikouNamae {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         int score, max = 0;
6         String namae, maxNamae = null;
7         System.out.print("名前を入力して下さい ");
8         namae = sc.next();
9         while(!(namae.equals("end"))){
10            System.out.print("得点を入力して下さい ");
11            score = sc.nextInt();
12            if( (1) ){
13                (2);
14                (3);
15            }
16            System.out.print("名前を入力して下さい ");
17            namae = sc.next();
18        }
19        System.out.print("最高点は " + maxNamae + " さんです");
20        sc.close();
21    }
22 }
```

(1)	
(2)	
(3)	



問題 6 - 6

2 級範囲

処理内容

売上数を入力し、最高売上数と最低売上数を算出する。売上数に 999999 が入力されると処理を終了し、最高売上数と最低売上数を表示する。

実行結果

```

売上数を入力：160
売上数を入力：140
売上数を入力：60
売上数を入力：100
売上数を入力：155
売上数を入力：999999
最高売上数 160
最低売上数 60
    
```

プログラム

```

1  import java.util.Scanner;
2  public class MaxMinMethod {
3      int urisuu, uriMax = 0, uriMin = 999999;
4      Scanner sc = new Scanner(System.in);
5      public MaxMinMethod(){
6          input();
7      }
8      public void input(){
9          System.out.print("売上数を入力：");
10         urisuu = sc.nextInt();
11     }
12     public void hantei(){
13         if(urisuu != 999999)
14             if(urisuu > uriMax)
15                 (1);
16             if(urisuu < uriMin)
17                 (2);
18     }
19     public void output(){
20         System.out.println("最高売上数 " + uriMax);
21         System.out.println("最低売上数 " + uriMin);
22     }
23     public static void main(String[] args) {
24         MaxMinMethod maxMin = new MaxMinMethod();
25         while(maxMin.urisuu != 999999){
26             (3);
27             maxMin.input();
28         }
29         (4);
30         maxMin.sc.close();
31     }
32 }
    
```

(1)	
(2)	
(3)	
(4)	











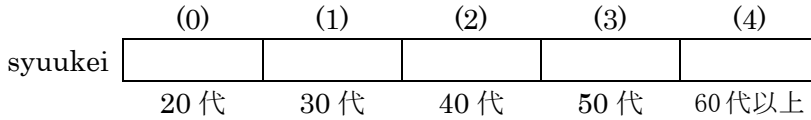


問題 7 - 6

2 級範囲

処理内容

参加者の年齢を入力し、下記の配列に集計する。なお、参加者は 20 歳以上である。年齢に 999 を入力すると集計結果を出力する。



実行結果

```

参加者の年齢を入力：25
参加者の年齢を入力：52
参加者の年齢を入力：34
参加者の年齢を入力：70
参加者の年齢を入力：63
参加者の年齢を入力：44
参加者の年齢を入力：999
20代   30代   40代   50代   60代以上
1人   1人   1人   1人   2人
    
```

プログラム

```

1  import java.util.Scanner;
2  public class NenreiSyuukei {
3      public static void main(String[] args) {
4          int sankasya, nendai;
5          int[] syuukei = {0, 0, 0, 0, 0};
6          Scanner sc = new Scanner(System.in);
7          System.out.print("参加者の年齢を入力：");
8          sankasya = sc.nextInt();
9          while(sankasya != 999){
10             nendai = sankasya / 10;
11             if(nendai >= 6)
12                 (1);
13             else
14                 (2);
15             syuukei[nendai]++;
16             System.out.print("参加者の年齢を入力：");
17             sankasya = sc.nextInt();
18         }
19         System.out.println("20代¥t30代¥t40代¥t50代¥t60代以上");
20         for(int i = 0; i < syuukei.length; i++){
21             System.out.print((3) + "人¥t");
22         }
23         sc.close();
24     }
25 }
    
```

(1)	
(2)	
(3)	







問題 8 - 3

2 級範囲

処理内容

荷物の重さをグラムで入力し、該当する価格を表示する。グラム数と価格は以下の通りである。なお、9999999 グラムを超える荷物は取り扱わない。

重量	1~100g	101~300g	301~500g	501~800g	801g 以上
価格	320 円	450 円	680 円	820 円	1000 円

	(0)	(1)	(2)	(3)	(4)
weight	100	300	500	800	9999999

	(0)	(1)	(2)	(3)	(4)
price	320	450	680	820	1000

実行結果

```

重量を入力して下さい 300
価格は 450 です
重量を入力して下さい 1500
価格は 1000 です
重量を入力して下さい 0
(プログラム終了)
    
```

プログラム

```

1  import java.util.Scanner;
2  public class HaniSearch {
3      public static void main(String[] args) {
4          (1);
5          int[] weight = {100, 300, 500, 800, 9999999};
6          int[] price = {320, 450, 680, 820, 1000};
7          Scanner sc = new Scanner(System.in);
8          System.out.print("重量を入力して下さい ");
9          inputWeight = sc.nextInt();
10         while(inputWeight != 0){
11             int i = 0;
12             while((2)){
13                 i++;
14             }
15             System.out.println("価格は " + price[i] + " です");
16             System.out.print("重量を入力して下さい ");
17             inputWeight = sc.nextInt();
18         }
19         sc.close();
20     }
21 }
    
```

(1)	
(2)	



## 問題 8 - 5

## 処理内容

売上げた商品のコードを入力し売上数を集計する。コードに 999 が入力されると集計結果を出力する。なお、存在しないコードは入力されない。また、配列 code と配列 uriagesuu は添字で対応している。

## 実行結果

```

コードを入力して下さい 102
コードを入力して下さい 103
コードを入力して下さい 102
コードを入力して下さい 101
コードを入力して下さい 999
コード 101 売上数 1
コード 102 売上数 2
コード 103 売上数 1
コード 104 売上数 0
コード 105 売上数 0

```

## プログラム

```

1 import java.util.Scanner;
2 public class SyuukeiMethod {
3     int inputCode, i;
4     int[] code = {101, 102, 103, 104, 105};
5     int[] uriagesuu = {0, 0, 0, 0, 0};
6     Scanner sc = new Scanner(System.in);
7     public void input(){
8         System.out.print("コードを入力して下さい ");
9         inputCode = sc.nextInt();
10    }
11    public void tansaku(){
12        i = 0;
13        while(inputCode != code[i]){
14            i++;
15        }
16    }
17    public void output(){
18        for(int j = 0; j < code.length; j++){
19            System.out.print("コード " + code[j]);
20            System.out.println(" 売上数 " + uriagesuu[j]);
21        }
22    }
23    public static void main(String[] args) {
24        SyuukeiMethod sm = new SyuukeiMethod();
25        sm.input();
26        while( (1) ){
27            sm.tansaku();
28            (2);
29            sm.input();
30        }
31        sm.output();
32        sm.sc.close();
33    }
34 }

```

(1)

(2)

問題 8 - 6

処理内容

売り上げた商品名と売上数を入力し集計する。集計後、売上が一番多い商品名と売上数を表示する。なお、存在しない商品名は入力されない。また、売上数は同一にならないものとする。配列 `hinmei` と配列 `uriage` は添字で対応している。

実行結果

```

商品名を入力して下さい うどん
売上数を入力して下さい 15
商品名を入力して下さい たこ焼き
売上数を入力して下さい 20
商品名を入力して下さい うどん
売上数を入力して下さい 30
商品名を入力して下さい 唐揚げ
売上数を入力して下さい 10
商品名を入力して下さい end
売上が一番多い商品は うどん です
売上数は 45 です
    
```

プログラム

```

1 import java.util.Scanner;
2 public class Syokudou {
3     public static void main(String[] args) {
4         String inputMei;
5         int inputUriage;
6         String[] hinmei = {"焼きそば", "うどん", "たこ焼き", "唐揚げ"};
7         int[] uriage = {0, 0, 0, 0};
8         Scanner sc = new Scanner(System.in);
9         System.out.print("商品名を入力して下さい ");
10        inputMei = sc.next();
11        while(!inputMei.equals("end")){
12            System.out.print("売上数を入力して下さい ");
13            inputUriage = sc.nextInt();
14            int i = 0;
15            while(!inputMei.equals(hinmei[i])){
16                (1);
17            }
18            (2);
19            System.out.print("商品名を入力して下さい ");
20            inputMei = sc.next();
21        }
22        int max = 0;
23        String maxMei = null;
24        for(int i = 0; i < hinmei.length; i++){
25            if((3)){
26                max = uriage[i];
27                (4);
28            }
29        }
30        System.out.println("売上が一番多い商品は " + maxMei + " です");
31        System.out.println("売上数は " + max + " です");
32        sc.close();
33    }
34 }
    
```

(1)	
(2)	
(3)	
(4)	

## 第9章 ファイル入出力

### 問題 9 - 1

教科書 P139

#### 処理内容

FileReader クラスを利用し、ファイル「data.txt」を読み込み表示する。

#### 入力ファイル data.txt

新井 優志
62
吉岡 亮太
74
吉野 ひより
97
.....

#### 実行結果

新井 優志
62
吉岡 亮太
74
吉野 ひより
97
.....

#### プログラム

```
1 import java.io.FileReader;
2 import java.io.IOException;
3 public class FileReaderSample {
4     public static void main(String[] args) throws IOException {
5         FileReader fr = new FileReader("c:¥¥data.txt");
6         int  = fr.read();
7         while(moji != -1){
8             System.out.print((char)moji);
9             moji = fr.read();
10        }
11        fr.close();
12    }
13 }
```





## 問題 9 - 3

教科書 P139

## 処理内容

CSV ファイル「data.csv」を読み込み表示する。

入力ファイル data.csv

新井 優志,62
吉岡 亮太,74
吉野 ひより,97
.....

実行結果

新井 優志	62
吉岡 亮太	74
吉野 ひより	97
.....	

## プログラム

```

1 import java.io.BufferedReader;
2 import java.io.FileReader;
3 import java.io.IOException;
4 public class CsvTest {
5     public static void main(String[] args){
6         BufferedReader (1) = null;
7         try {
8             br = new BufferedReader(new FileReader("c:¥¥data.csv"));
9             String line;
10            while(( (2) = br.readLine()) != null){
11                String[] str = line.split(",");
12                String (3) = str[0];
13                int (4) = Integer.parseInt(str[1]);
14                System.out.println(name + "¥t" + score);
15            }
16            br.close();
17        } catch (IOException e) {
18            System.out.println("入出力例外が発生しました");
19        }
20    }
21 }

```

(1)	
(2)	
(3)	
(4)	

## 問題 9 - 4

教科書 P139

処理内容

FileWriter クラスを利用し、A~Z までの文字をファイル「moji.txt」に書き込む。  
なお、A の文字コードは 65 であり、Z の文字コードは 90 である。

出力ファイル moji.txt

ABCDEFGHIJKLMNPOQRSTUVWXYZ

プログラム

```
1 import java.io.FileWriter;
2 import java.io.IOException;
3 public class FileWriterSample {
4     public static void main(String[] args) throws IOException {
5         FileWriter fw = new FileWriter("c:¥¥moji.txt");
6         for(int i = 65; i < 91; i++){
7             .write(i);
8         }
9         fw.close();
10    }
11 }
```





**編 集**

商業科の Java 教育を考える会

**著 作 者**

商業科の Java 教育を考える会

**連 絡 先**

商業科の Java 教育を考える会 事務局  
syougyoukyouiku@yahoo.co.jp  
<http://syougyou.ninpou.jp/>

第 1 版 (V1.4) 2015.04.19

年 組 番 氏名