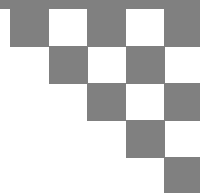


ゼロから始めるJava基礎研修 問題集



演習問題1. 文字列の出力

【問題】

自分の名前と年齢を画面に出力するプログラムを作成してください。

※名前と年齢は変数に格納して、出力の際も変数を使うようにしてください。

【実行結果】

私の名前は鈴木一郎です。年齢は47歳です。

演習問題2. 繰返し文による配列要素の出力と入れ替え

【問題】

下記の【配列】に格納された配列要素を全て画面に出力させてください。
また、配列要素の順番を逆順に入れ替えて、
入れ替え後の配列要素を全て画面に出力するプログラムを作成してください。

【配列】

24, 10, 55, 69, 2, 91, 73, 12, 32, 46

【実行結果】

入れ替え前の配列

24, 10, 55, 69, 2, 91, 73, 12, 32, 46

入れ替え後の配列

46, 32, 12, 73, 91, 2, 69, 55, 10, 24

演習問題3. コマンドラインからの値の入力

【問題】

コマンドラインから1つ以上の数値データを入力し、

それらの合計(int 型で計算)と平均(float 型で計算)を算出してください。

入力された数値データの個数と算出結果(合計と平均)を画面に出力するプログラムを作成してください。

※コマンドライン引数の入力値が1つもない場合は、入力が必要な旨を表示し、プログラムを強制終了します。

【準備済みのコード】

```
class CommandLine {  
    public static void main(String[] args) {  
        int total = 0;  
        float ave = 0.0f;  
        //以下に必要なコードを記述してください。  
    }  
}
```

【実行結果】

//引数を1つ以上入力した場合(12 59 41を入力)

入力された数値 : 3 個

合計 : 112

平均 : 37.0

//引数の入力値が1つもない場合

1 以上の数値を入力してください

演習問題4. 文字列の比較

【問題】

10人にごはん派かパン派かどちらであるかのアンケートを実施しました。
アンケート結果を配列にすると、下記の【アンケート結果】の様な配列になりました。
この配列を利用して、ごはん派の人数およびパン派の人数を求め、
画面に出力するプログラムを作成してください。
※ごはん派とパン派のどちらでもない人およびどちらでもある人はいないものとします。

【アンケート結果】

```
String style[] = { "パン", "ごはん", "パン", "ごはん", new String("パン"),  
                  "ごはん", "ごはん", "ごはん", "パン", new String("ごはん") }
```

【準備済みのコード】

```
class CompareString {  
    public static void main(String[] args) {  
        String style[] = { "パン", "ごはん", "パン", "ごはん", new String("パン"),  
                           "ごはん", "ごはん", "ごはん", "パン", new String("ごはん") }  
  
        //以下に必要なコードを記述してください。  
    }  
}
```

【実行結果】

ごはん派の人は 6 人です。
パン派の人は 4 人です。

演習問題5. インスタンスの生成と実行

【問題】

Reservation(予約)クラスは、レンタカーの予約状況を管理するクラスです。

このクラスのインスタンスを生成し、以下の【操作】を実行するMainクラスのmainメソッドを作成してください。

【操作】

- ① レンタカーの予約状況を確認
- ② 4号車を予約
- ③ 2号車を予約
- ④ レンタカーの予約状況を確認

【実行結果】

現在、1号車、4号車が空いています。

4号車を予約しました。

2号車の予約を取り消しました。

現在、1号車、2号車が空いています。

【Mainクラスの処理手順】

- ① Reservationクラスのインスタンスを生成する。
- ② ①で生成したReservationインスタンスのcheck()メソッドを呼び出す。
- ③ Reservationインスタンスのreserv()メソッドを呼び出す。
- ④ Reservationインスタンスのcancel()メソッドを呼び出す。
- ⑤ ①で生成したReservationインスタンスのcheck()メソッドを呼び出す。

演習問題6. 利用されるクラスの作成とメソッドの定義

【問題】

MainクラスはFruits(果物)クラスのインスタンスを生成、注文、表示処理を実行します。

【実行結果】およびMainクラスを参考に、【クラスの構成】に記載されているメンバを持つFruits(果物)クラスを作成してください。

【クラスの構成】

- | | |
|---|--|
| <ul style="list-style-type: none">・メンバ変数(括弧内は変数名)消費税(TAX) : クラス変数, 値は0.08品名(name)単位(unit)単価(price)注文数(stock) | <ul style="list-style-type: none">・メンバメソッド(括弧内はメソッド名)初期化处理(initialize)
→Fruitsインスタンスのname, unit, priceを設定する。注文処理(order)
→Fruitsインスタンスのstockを設定する。合計金額の取得(getTotal)
→単価(price) × 注文数(quantity)の結果を返す |
|---|--|

【実行結果】

品名 : バナナ
単位 : 房
単価 : 200
注文数 : 10

売上金額 : 1000
消費税 : 160
合計 : 2160

演習問題7. コンストラクタの利用

【問題】

MainクラスはFruits(果物)クラスのインスタンスを生成、注文、表示処理を実行します。

【実行結果】 およびMainクラスを参考に、Fruits(果物)クラスのコンストラクタを作成してください。

【実行結果】

品名 : バナナ
単位 : 房
単価 : 200
注文数 : 10

売上金額 : 2000

演習問題8. パッケージの利用とJARファイル化

【問題】

Mainクラスは、Store(お店)クラスのインスタンスを生成し、果物を購入するJavaプログラムです。

【実行結果】、「q8」プロジェクトのStoreクラスおよびFruitsクラスを参考にMainクラスを完成させてください。

※Mainクラスでは、Fruitsクラスのインスタンス生成を行いません。

※「q8」プロジェクトをJARファイルとしてエクスポートして、Mainクラスにインポートさせてください。

(詳細は【作業手順】を参照してください。)

【実行結果】

レーズンを4袋購入

>>ありがとうございます。代金は1600円です。

バナナを15房購入

>>申し訳ありません。バナナは在庫が足りません。

レーズンを3袋購入

>>申し訳ありません。レーズンは売り切れ中です。

なしを10個購入

>>申し訳ありません。なしは扱っておりません。

【作業手順】

- (1) Mainクラスは【実行結果】、「q8」プロジェクトにあるShopクラスとFruitsクラスをもとに1から作成してください。
- (2) 「q8」プロジェクトを右クリック>エクスポート>JARファイルを選択>「次へ」をクリック
- (3) JARファイル仕様でエクスポート先を「JavaWork/lib/q8.jar」に指定する
- (4) 「完了」をクリック
- (5) 「JavaWork」プロジェクトを右クリック>ビルドパス>ビルドパスの構成をクリック
- (6) 「ライブラリー」タブの「モジュールパス」をクリック
- (7) 「外部JARの追加」をクリックし、「q8.jar」を選択
- (8) 「適用して閉じる」をクリック
- (9) Mainクラスに「import q8.Store;」と入力、保存、実行する。

演習問題9. 応用問題

【問題】

Fortune(占い)クラスはコマンドライン引数に入力された2つの生まれ年をもとに相性を診断するクラスです。

Mainクラスはインスタンスを生成し、2つの生まれ年をもとに干支と診断結果を表示するメソッドを呼び出します。

【実行結果】、【Fortuneクラスについて】およびMainクラスをもとにFortuneクラスのメソッドを記述してください。

【生まれ年と干支の関係】

●生まれ年を12で割った余りの数で干支が決定する。

余り	0	1	2	3	4	5	6	7	8	9	10	11
干支	申	酉	戌	亥	子	丑	寅	卯	辰	巳	午	未

例) 1997年生まれの場合 : $1997 \% 12 = 5 \cdots$ 丑年 1999年生まれの場合 : $1999 \% 12 = 7 \cdots$ 卯年

【Fortuneクラスについて】

●メソッド

- ・コンストラクタ(引数に入力された2つの生まれ年を各フィールドに設定する)
- ・干支を算出するメソッド(引数は生まれ年、戻り値は干支の文字列)
- ・相性を診断するメソッド(引数は自分と相手の生まれ年、戻り値は相性の文字列)
- ・結果を表示させるメソッド(干支の算出(自分と相手分)と相性診断メソッドを呼び出し、結果を表示)

●相性診断の方法

- ① 入力された2つの生まれ年が、それぞれどの干支になるかを調べる
- ② 2つの干支に対応する番号の差がどれだけ離れているか(絶対値)を調べ、相性診断を行なう。

●干支と対応する数の関係

余り	9	10	11	12	1	2	3	4	5	6	7	8
干支	申	酉	戌	亥	子	丑	寅	卯	辰	巳	午	未

●数の差と相性診断の結果

- ・2つの干支に対応する数の差の絶対値が1または11の場合 : 相性は「とてもよい」
- ・2つの干支に対応する数の差の絶対値が2または10の場合 : 相性は「よい」
- ・2つの干支に対応する数の差の絶対値が6の場合 : 相性は「悪い」
- ・2つの干支に対応する数の差の絶対値が4または8の場合 : 相性は「やや悪い」
- ・2つの干支に対応する数の差の絶対値が上記以外の場合 : 相性は「普通」

【絶対値の調べ方】

- ・絶対値は「Math.abs」メソッドを利用することで求められます。

例) 「8 - 12」の計算結果の絶対値を求める場合、Math.abs(8 - 12) ... 出力結果「4」

【実行結果】

// 「1992」と「1994」を指定した場合

あなたの生まれ年 : 1992

あなたの干支 : 申年

パートナーの生まれ年 : 1994

パートナーの生まれ年 : 戌年

二人の相性は「よい」です。

// 「1997」と「2006」を指定した場合

あなたの生まれ年 : 1997

あなたの干支 : 丑年

パートナーの生まれ年 : 2006

パートナーの干支 : 戌年

二人の相性は「普通」です。

// 2つ以外の数を指定した場合

自分の生まれ年と相手の生まれ年を西暦'yyyy'の形で入力してください。

演習問題10. 継承を利用したサブクラスの定義

【問題】

Worker (労働者) クラスとMainクラスがあります。

【実行結果】とこれらのクラスの内容をもとに、
Workerクラスを継承したMemberクラス(メンバ)とManager(マネージャ)クラスを作成してください。
ただし、サブクラス内でスーパークラスのメソッドと同じ処理が必要な場合、
スーパークラスのメソッドを使用します。

【実行結果】

名前は武井江洲太郎、年齢は26歳です。
入社3年目です。

名前は茂出須一郎、年齢は45歳です。
私の役職は部長です。

演習問題11. 抽象クラスとポリモーフィズム

【問題】

商品情報を持つMenu(メニュー)クラスを継承したAppOrder(アプリ注文)クラスとCallOrder(電話注文)クラスがあり、それらを利用するMainクラスがあります。

【実行結果】とAppOrderクラスおよびCallOrderクラスの内容を元に

Menuクラスを作成し、Mainクラスの処理を追加してください。

ただし、Menuクラスは抽象クラスとし、Menuクラスが持つ「calcDiscount」メソッドは、抽象メソッドとします。

【実行結果】

アプリ注文時の価格

商品 : みそラーメン

単価 : 680

個数 : 5

金額 : 3400

割引額 : 510

請求額 : 2890

アプリ注文時の価格

商品 : マーボー豆腐

単価 : 650

個数 : 2

金額 : 1300

割引額 : 195

請求額 : 1105

アプリ注文時の価格

商品 : チャーハン

単価 : 630

個数 : 4

金額 : 2520

割引額 : 378

請求額 : 2142

電話注文時の価格

商品 : 餃子

単価 : 350

個数 : 6

金額 : 2100

割引額 : 420

送料 : 500

請求額 : 2180

電話注文時の価格

商品 : からあげ

単価 : 600

個数 : 3

金額 : 1800

割引額 : 360

送料 : 500

請求額 : 1940

演習問題12. インタフェースの定義とクラスの実装

【問題】

Product (製品) Interface インタフェースを実装したDispenser (ディスペンサ) クラスとBeverage (飲み物) クラスがあり、さらにBeverage クラスを継承したAlcohol (アルコール) クラスがあります。

【実行結果】とこれらのクラスを利用するMainクラス、Beverageクラスのソースコードの内容をもとに、Product インタフェースとDispenser クラスおよびAlcohol クラスを作成してください。

※Product インタフェースではorder (注文) メソッドとdisplay (表示) メソッドのみを定義してください。

【実行結果】

注文情報

品名 : ビール
単位 : 10リットル
単価 : 5800
注文数 : 3
合計金額 : 17400
度数 : 7%

品名 : ビールディスペンサ
ノズル : 二口
単価 : 83780
注文数 : 1
合計金額 : 83780

【シーケンス図】

別途記載

【クラス図】

別途記載

演習問題13. インタフェースの継承

【問題】

MemberInterfaceインタフェースを実装したMemberクラスと

ManagerInterfaceインタフェースを実装したManagerクラスがあります。

また、PlayingManager(プレイイングマネージャ)InterfaceインタフェースはMemberInterfaceインタフェースとManagerInterfaceインタフェースを継承しています。

PlayingManagerクラスはPlayingManagerInterfaceインターフェースを実装しており、

MemberクラスとManagerクラスの特徴を併せ持ちます。

【実行結果】、各種インタフェース、クラスおよびMainクラスの内容をもとに、

PlayingManagerInterfaceとPlayingManagerクラスを作成してください。

PlayingManagerクラスは、MemberクラスとManagerクラスのインスタンスをメンバ変数で保持し、各インスタンスのメソッドを利用して、処理を行なうこととします。

【実行結果】

武井江洲太郎です

社会人3年目です

茂出須一郎です

社会人 22 年目です

私の役職は部長です

マネジメントが得意です

演習問題14. コレクションの利用

【問題】

Staff(スタッフ)クラスは、個人情報(名前、出身地、月収のデータ)を管理します。

MainクラスはStaffインスタンスを生成し、個人情報を登録します。

その後、個人情報としてStaffインスタンスをArrayListへ格納し、画面へ表示します。

【実行結果】とStaffクラスの内容をもとに、Mainクラスを作成してください。

【実行結果】

スタッフ情報

```
-----  
武威江洲太郎さん > 出身地: 神奈川県川崎市 > 月収: 29万円  
茂出須一郎さん > 出身地: 大阪府大阪市 > 月収: 65万円  
天王洲愛子さん > 出身地: 兵庫県神戸市 > 月収: 80万円  
中塔二三男さん > 出身地: 埼玉県所沢市 > 月収: 45万円  
-----
```

平均月収 : 54.75万円

演習問題15. 発生箇所での例外処理

【問題】

Mainクラスは、コマンドライン引数に指定された2つの数値をもとに、Union(結合)クラスのクラスメソッドを利用して、Unionクラスのフィールドであるlangage配列とlevel配列に格納されている文字列を結合します。コマンドライン引数に数値以外を指定した場合、NumberFormatExceptionの例外が発生します。また、各配列に存在しない添え字を指定した場合、ArrayIndexOutOfBoundsExceptionの例外が発生します。例外が発生した時に、正しい入力を促すように文字を表示させるようにしてください。
※例外処理は例外発生個所で行なってください。

【発生する例外】

NumberFormatException: 数値化できない文字列を指定した場合の例外
ArrayIndexOutOfBoundsException: 配列に存在しない添え字を指定した場合の例外

【配列】

- ・ language = {"Java", "Python", "Ruby", "PHP", "C", "C++"};
- ・ level = {"未経験者", "入門者", "初心者", "中級者", "上級者"};

【実行結果】

//引数を「0 1」とした場合
Java入門者

//引数を「0 -」とした場合
数値1、数値2には整数を入力してください

//引数を「5 5」とした場合
引数には配列内の添え字を指定してください

演習問題16. throwsを利用した例外処理

【問題】

Mainクラスは、コマンドライン引数に指定された2つの数値をもとに、Union(結合)クラスのクラスメソッドを利用して、Unionクラスのフィールドであるlangage配列とlevel配列に格納されている文字列を結合します。コマンドライン引数に数値以外を指定した場合、NumberFormatExceptionの例外が発生します。また、各配列に存在しない添え字を指定した場合、ArrayIndexOutOfBoundsExceptionの例外が発生します。例外が発生した時に、正しい入力を促すように文字を表示させるようにしてください。
※例外処理はすべてMainクラスで行なってください。(UnionクラスからMainクラスへ例外を伝播させる)

【発生する例外】

NumberFormatException:数値化できない文字列を指定した場合の例外
ArrayIndexOutOfBoundsException:配列に存在しない添え字を指定した場合の例外

【配列】

- ・ language = {"Java", "Python", "Ruby", "PHP", "C", "C++"};
- ・ level = {"未経験者", "入門者", "初心者", "中級者", "上級者"};

【実行結果】

//引数を「0 1」とした場合
Java入門者

//引数を「0 ー」とした場合
数値1、数値2には整数を入力してください

//引数を「5 5」とした場合
引数には配列内の添え字を指定してください

※ソースコードは演習問題17.と同様です。

演習問題17. ユーザ定義例外

【問題】

Mainクラスは、商品一覧を持つFruitsList(フルーツリスト)クラスのインスタンスを生成し、コマンドライン引数に指定された商品コードを持つ商品情報をFruitsListインスタンスから検索します。その後、検索結果を表示します。

商品コードを複数入力した場合、入力された商品コードに該当する商品情報を全て画面に表示します。商品コードに対応する商品がなかった場合、ユーザ定義の例外 (UserException) を発生します。

【実行結果】、Mainクラス、FruitsクラスおよびFruitsListクラスを参考に、UserExceptionクラスを作成し、FruitsListクラスを完成させてください。

※例外処理はすべてMainクラスで行なってください。

【実行結果】

```
//引数を「F001」とした場合
コード 品名 単位 単価
```

```
-----
F001 リンゴ 個 150
```

```
//引数を「F001 F005」とした場合
コード 品名 単位 単価
```

```
-----
F001 リンゴ 個 150
F005 レーズン 袋 300
```

```
//引数を「F006」とした場合
コード 品名 単位 単価
```

```
-----
商品コード「F006」に対応する商品はありません
```

```
//引数を「F006 F001」とした場合
コード 品名 単位 単価
```

```
-----
商品コード「F006」に対応する商品はありません
```




株式会社VSN

Copyright© VSN, Inc All right reserved

