

【JAVA言語スキル講座 Step1】

■ 課題7 ■

Eclipse上で『J1kadai7 xxxx』(xxxxは社員番号)というプロジェクトを作成し、以下のプログラムを作成すること

問題① これまでの課題のまとめ

以下の要件を満たすプログラムを作成すること。

・クラス構成は自由とする

・開始すると、キーボードから以下の3つの情報を受け取ること

- 1: タイプ 0/1
- 2: 配列の要素数(1-100)
- 3: 処理モード [0/1]

文字入力を受け付けた後のバリデーションでエラーが発生した場合は、独自例外クラス(Exceptionを継承したクラス)にエラーIDを与え、スローするようにすること。独自例外クラス「Kadai7Exception」を作成し、Int変換に失敗した場合のスロー方法の例を以下に示す。

```
try {
    num = Integer.parseInt(line);
} catch (NumberFormatException e) {
    Kadai7Exception exception = new Kadai7Exception(Kadai7Exception.ERR_CODE_NOT_NUM);
    throw exception;
}
```

最低限想定しているエラーは以下である。必要に応じて、エラーIDと対応メッセージを追加すること。

- 数値でない
- 下限を下回る
- 上限を上回る

・入力を受け付け時に[q|Q]が入力された場合は、その時点でプログラムを終了すること

・タイプ:モードごとの処理概要を以下に示す

タイプ	0:INT	1:String
処理モード	0: 数字の出現回数のカウント 1: int → アルファベット変換	0: ソート 1: 提供なし

・タイプ:INT、String用に別々にクラスを作成すること

各処理の流れは以下のように実装すること

	処理の流れ	概要		
		INT		String
		モード0	モード1	モード0
1	配列の初期化	指定された要素数の配列を作成し、それぞれの要素にランダム値[10~90]を格納する		指定された要素数の配列を作成し、それぞれの要素に1~9文字のランダムな文字列([a-z][1,6])を生成すること
2	配列の操作	10~90の出現回数をカウントする	10~90それぞれの数字をアルファベットに変換する ※1	第1ソート:文字数 第2ソート:文字の昇順ソート ※2
3	結果の出力	検索結果を出力する 1行に20項目出力すること※3		検索結果を出力する1行1件で良い

※1 a-zをそれぞれ0-25の数字に割り振る。配列値%26の値を変換値とすること

※2 Collectionフレームワークの機能(Arrays.sortやCollection.sort等)は使用せず、ベタに実装すること

※3 10~90の出現回数を表示する際、出現回数が0回のは出力しない

【JAVA言語スキル講座 Step1】

■ 課題7 ■

- ・[q][Q]が入力されるまでは、何度でもタイプ/配列数/モードの入力を促し、処理結果を出力すること
(値が不正な場合は、再度入力を促すこと)
- ・継承、実装、カプセル化の概念を用い、できる限り冗長な実装を防ぎ、Object指向を意識したプログラムを作成すること
- ・実行結果例を以下に示します

タイプ0:モード0

```
処理タイプ[0:int]/[1:String]を入力してください。[0-1][終了:Q] ↵
>0 ↵
作成する配列の数を入力してください。[1-100][終了:Q] ↵
>1000 ↵
エラーID:4 [上限を超えています] ↵
作成する配列の数を入力してください。[1-100][終了:Q] ↵
>100 ↵
処理モードを入力してください。[0-1][終了:Q] ↵
>0 ↵
作業用配列を初期化します ↵
要素の出現数をカウントします ↵
変更前の配列は以下です ↵
50 10 19 64 52 45 23 74 18 55 15 64 56 39 49 48 52 53 50 52 ↵
68 27 48 51 19 57 66 19 68 70 45 65 67 60 61 53 26 53 19 73 ↵
71 78 52 70 51 59 29 74 67 68 61 20 61 54 44 67 60 26 61 12 ↵
42 49 15 38 19 33 69 71 64 56 17 72 73 45 24 16 44 28 75 78 ↵
38 19 67 67 15 36 49 39 75 68 67 15 30 20 76 21 66 52 11 51 ↵
↵
文字の出現頻度は以下です。 ↵
10:1 11:1 12:1 15:4 16:1 17:1 18:1 19:6 20:2 21:1 23:1 24:1 26:2 27:1 28:1 29:1 30:1 33:1 36:1 38:2 ↵
39:2 42:1 44:2 45:3 48:2 49:3 50:2 51:3 52:5 53:3 54:1 55:1 56:2 57:1 59:1 60:2 61:4 64:3 65:1 66:2 ↵
67:6 68:4 69:1 70:2 71:2 72:1 73:2 74:2 75:2 76:1 78:2 ↵
処理タイプ[0:int]/[1:String]を入力してください。[0-1][終了:Q] ↵
>
```

タイプ0:モード1

```
処理タイプ[0:int]/[1:String]を入力してください。[0-1][終了:Q] ↵
>a ↵
エラーID:1 [数字を入力してください] ↵
処理タイプ[0:int]/[1:String]を入力してください。[0-1][終了:Q] ↵
>0 ↵
作成する配列の数を入力してください。[1-100][終了:Q] ↵
>100 ↵
処理モードを入力してください。[0-1][終了:Q] ↵
>1 ↵
作業用配列を初期化します ↵
アルファベットに変換します ↵
変更前の配列は以下です ↵
67 64 64 16 29 15 20 58 30 41 38 19 31 23 46 44 30 64 57 52 ↵
47 44 31 50 67 41 51 15 51 58 25 56 56 21 22 12 68 55 20 44 ↵
12 76 79 76 11 23 65 41 79 36 77 55 45 76 54 56 32 36 74 26 ↵
25 70 31 41 54 10 10 18 48 40 39 20 14 63 56 30 24 56 28 76 ↵
38 41 30 12 45 15 37 23 73 32 72 59 56 66 12 35 57 76 77 69 ↵
↵
数値からアルファベットに変更しました。 ↵
67→p 64→n 64→n 16→q 29→d 15→p 20→u 58→g 30→e 41→p 38→n 19→t 31→f 23→x 46→u 44→s 30→e 64→n 57→f 52→a ↵
47→v 44→s 31→f 50→y 67→p 41→p 51→z 15→p 51→z 58→g 25→z 56→e 56→e 21→v 22→w 12→n 68→q 55→d 20→u 44→s ↵
12→n 76→y 79→b 76→y 11→l 23→x 65→m 41→p 79→b 36→k 77→z 55→d 45→t 76→y 54→c 56→e 32→g 36→k 74→w 26→a ↵
25→z 70→s 31→f 41→p 54→c 10→k 10→k 18→s 48→w 40→o 39→m 20→u 14→o 63→l 56→e 30→e 24→y 56→e 28→c 76→y ↵
38→n 41→p 30→e 12→n 45→t 15→p 37→l 23→x 73→v 32→g 72→u 59→h 56→e 66→o 12→n 35→j 57→f 76→y 77→z 69→r ↵
```

タイプ1:モード0

```
処理タイプ[0:int]/[1:String]を入力してください。[0-1][終了:Q]
>1
作成する配列の数を入力してください。[1-100][終了:Q]
>10
処理モードを入力してください。[0-1][終了:Q]
>0
ソートします
ソート前の順番は以下です。
0:ahoxbin
1:ueplnu
2:swvyg
3:ojbdpe
4:zc
5:zwbrimt
6:wnwvbq
7:exjn
8:adcdooch
9:kjkfcp
ソート後の順番は以下です。
0:zc
1:exjn
2:swvyg
3:kjkfcp
4:ojbdpe
5:ueplnu
6:ahoxbin
7:wnwvbq
8:zwbrimt
9:adcdooch
```

タイプ1:モード0

```
処理タイプ[0:int]/[1:String]を入力してください。[0-1][終了:Q]
>1
作成する配列の数を入力してください。[1-100][終了:Q]
>10
処理モードを入力してください。[0-1][終了:Q]
>1
文字列モードは:モード0しか提供していません
```

Qを入力

```
処理タイプ[0:int]/[1:String]を入力してください。[0-1][終了:Q]
>q
終了コードが入力されたため、終了します
```