

# プログラミング演習 I (平成 21 年度)

## – 第 2 回 レポート課題 –

担当: 時井, 松村

提出日 12 月 1 日 (火) 17 時

### 1 課題 (設問 1)

Web ページに掲載した入力ファイル (uriage.txt) を読み込むと, 各顧客毎の年購入額, 月毎の各項目販売金額を表示するプログラムを作成する.

#### [入力ファイルに関して]

- ファイルの中身は, 顧客名, 日付 (月. 日), 商品名, 金額 (円) の順である.
- ファイルの最後は . (ピリオド) である.

#### [プログラム作成上の条件]

- 入力ファイル (uriage.txt) を読み込む.
- 顧客名, 食品名, 衣料品名, 雑貨名を保存する配列を用意する. 各配列には, 以下の要素を保存する.

配列名	要素
kokyaku	"A さん" "B さん" "C さん" "D さん"
syoku	"魚肉" "飲料" "米" "野菜"
irui	"服" "靴下"
zakka	"台所用品" "文具" "バス用品"

- 下記の配列を準備し, 入力ファイルから読み込んだ値を保存していく.

配列名	要素数	保存する値
kokyaku_kaikei	4	各顧客毎の年購入額
syoku_kaikei	12	月毎の食料品の販売金額
irui_kaikei	12	月毎の衣料品の販売金額
zakka_kaikei	12	月毎の雑貨の販売金額
sonota_kaikei	12	月毎のその他の販売金額

- 「各顧客毎の年購入額」と「月毎の各項目の販売金額」を表示する.
- 各項目毎の年間販売金額を返すメソッド (goukei) を作成し, 各項目毎に年間販売金額を表示する. なお, メソッド goukei は引数を配列, 戻り値を年間販売金額とする. 使い方は例えば, 以下の通りである.

```
print("食料費 年合計", goukei(syoku_kaikei), "円\n")
print("雑貨費 年合計", goukei(zakka_kaikei), "円\n")
```

## 表示例

---

### 顧客別購入額

A さん:232200 円  
B さん:282200 円  
C さん:269900 円  
D さん:202000 円

---

---

### 1月の販売金額

食料費:25200 円  
衣料費:55800 円  
雑貨費:6100 円  
その他:10200 円

---

---

### 2月の販売金額

食料費:12100 円  
衣料費:11200 円  
雑貨費:7600 円  
その他:9400 円

---

---

### 3月の販売金額

食料費:16400 円  
衣料費:44000 円  
雑貨費:4500 円  
その他:12700 円

---

---

### 4月の販売金額

食料費:5500 円  
衣料費:46800 円  
雑貨費:7900 円  
その他:23200 円

---

---

### 5月の販売金額

食料費:10000 円  
衣料費:33200 円  
雑貨費:7500 円  
その他:18600 円

---

---

### 6月の販売金額

食料費:8600 円  
衣料費:54400 円  
雑貨費:5100 円  
その他:25500 円

---

---

### 7月の販売金額

食料費:12200 円  
衣料費:67400 円  
雑貨費:5600 円  
その他:10500 円

---

---

### 8月の販売金額

食料費:20000 円  
衣料費:51800 円  
雑貨費:7000 円  
その他:14900 円

---

---

### 9月の販売金額

食料費:12000 円  
衣料費:52400 円  
雑貨費:7300 円  
その他:20400 円

---

---

### 10月の販売金額

食料費:17000 円  
衣料費:26200 円  
雑貨費:6200 円  
その他:12900 円

---

---

### 11月の販売金額

食料費:8600 円  
衣料費:59800 円  
雑貨費:5600 円  
その他:17800 円

---

---

### 12月の販売金額

食料費:14600 円  
衣料費:46600 円  
雑貨費:2400 円  
その他:25600 円

---

++++  
食料費 年合計 162200 円

衣料費 年合計 549600 円

雑貨費 年合計 72800 円

その他 年合計 201700 円

++++

## 2 課題 (設問 2)

単語のリストをファイルから読み込み、最大頻度の単語とその頻度、頻度 2 以上の単語を頻度に応じたサイズと色 (どちらも 7 段階) で表示する HTML ファイルを作成する。それぞれの単語には Twitter への検索リンクを張る。

### [HTML ファイルの全体構造]

おおよそ以下の通りである。title タグと h1 タグは指定通りにすること。

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
  <title>第 2 回レポート課題 設問 2 みんなの関心</title>
</head>
<body>
<h1>みんなの関心</h1>
</body>
</html>
```

<=== ここまでは指定通り  
<=== 最大頻度の単語とその頻度  
<=== 単語リスト

### [各単語の記述部分]

「ポニヨ」という単語の場合の例を示す。

```
<a href="http://twitter.com/search?q=%E3%83%9D%E3%83%8B%E3%83%A7"><font size=? color=?>
ポニヨ</font></a>
```

- a タグで Twitter への検索リンクを作成する。
- font タグでサイズと色を指定する。なお、?は見やすくなるよう 7 段階で決める。
- %E3%83%9D%E3%83%8B%E3%83%A7 は文字列ポニヨを UTF-8 で URL エンコードした文字列である。UTF-8 での URL エンコードの仕方は「(補足) URL エンコード」を参照のこと。

### [プログラムの条件]

- 最大値を求めるメソッド (count\_max)、サイズを求めるメソッド (font\_size)、色を求めるメソッド (font\_color) の 3 つを定義して使う。
- メソッド font\_color は、メソッド font\_size を呼び出し、サイズに応じて色を決めるようにする。
- 入力ファイルとなる単語のリスト (kanshinW2009.txt) は共通で、演習のページからダウンロードする。なお、ファイルの末尾はデータの終わりを意味する。(ピリオド) となっている。

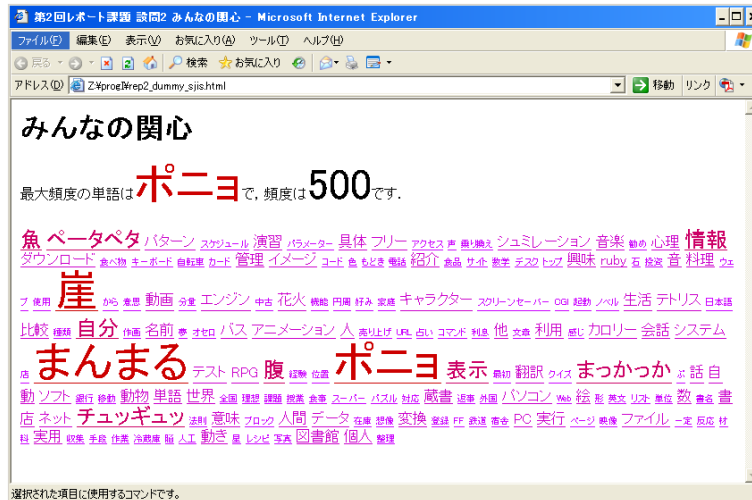
### [実行結果]

次の 2 種類を提出する。

- 通常の実行結果 (作成した HTML ファイルのソースが画面に表示されるもの)
- Internet Explorer で表示した結果を「カラー印刷」したもの。

## [作成例 (データは課題とは別のもの)]

見やすいかどうか各自評価してみよう。



## ( 補足 ) URL エンコード

URL に ASCII 文字以外を用いる場合に使われる表記法で、%xx の形で表記される。Twitter への検索 URL を作成するには UTF-8 で URL エンコードをしなければならない。手順は、以下の通りである。

### 1. UTF-8 に変換

- プログラム先頭の方に以下を記述しておく

```
require 'iconv'
```

- 変換したい文字列を "ポニョ" とすると、以下のようにすると UTF-8 に変換された文字列を keyword に代入できる。

```
keyword = Iconv.conv("UTF-8", "shift_jis", "ポニョ")
```

### 2. URL エンコード

- プログラム先頭の方に以下を記述しておく

```
require 'uri'
```

- UTF-8 に変換した keyword を URL エンコードする場合は、以下のようにする。

```
str = URI.encode(keyword)
```

まとめると、課題のような URL を作成する場合には、以下のような手順で行なうことになる。

```
keyword = "ポニョ"  
word = Iconv.conv("UTF-8", "shift_jis", keyword)      # UTF-8 に変換  
tmp = "http://twitter.com/search?q=" + word  
url = URI.encode(tmp)                                  # URL エンコード
```

この文字列 url は、先に示した `http://twitter.com/search?q=%E3%83%9D%E3%83%8B%E3%83%A7` のような文字列になる。

### 3 提出における注意

- 締切：締切を厳守すること。締切に遅れたレポートは受理しない。
- 提出場所：学務課レポート提出用ポスト  
自分の受講クラスのポストに入れること。他のクラスのポストに入れた場合は受理しない。

以下を遵守しないレポートは減点の対象となる。

- 設問毎に A4 用紙にまとめ左上をステーブラ（ホチキス）でとめる。（今回は設問が 2 個あるので、合計 2 部提出）
- 設問毎に表紙，本文の順でとめる。
  - － 表紙：科目名，曜日クラス，第 x 回レポート 設問番号，提出日，学籍番号，氏名を記入する。  
見本を Web ページに掲載するので，その形式にしたがうこと。
  - － 本文：行番号付きプログラムリスト，実行結果とプログラムの説明を この順で載せる。  
プログラムリストと実行結果は続けて記述しても良いが，プログラムの説明は別ページとすること。  
いずれも機械出力とする。その際，プログラムリストと実行結果は，等幅フォントとする。  
出力したプログラムや実行結果に手書きしないこと。  
なお，付録に形式の一例を載せたので参考にしてほしい。
- 実行結果に関する注意事項
  - － 実行は，全て教育用計算機システム (icho) の Windows 上で行うこと。
  - － 実行 (例 ruby `***.rb`)，出力される実行結果の順となるように，一連の操作の出力を記述する。  
(注意：余分な操作を途中で入れないこと)
  - － 実行結果は必要に応じて複数示すこと。
- 可読性が良くなるよう努めること。「プログラム」は適切な字下げと空行の挿入をし、「説明」は適切な見出しを付けるなど，構造がわかるように注意すること。
- 片面印刷にすること。

### 4 その他の注意

- 同一/類似レポートは「両成敗」。すなわち，見た方も見せた方も不正行為とみなし，単位を出さない。また，他の科目にも影響することもある。他人が不正にプログラムにアクセスできないよう，各自の責任において対処すること。
- レポートの差し替えや再提出は認めないので，内容をよく確認してから提出すること。
- レポートを書く上での一般常識として，参考文献がある場合には書誌情報を載せること。
- フォーマットが細かく設定されているのは決して意地悪からではない。100 名を越える受講者のレポートを効率的かつ公平に採点するために必要なことだからである。  
言い替えれば，フォーマットを無視したレポートは演習担当者の作業を著しく妨げることになる点を理解してほしい。

## (付録) 行番号付きプログラム等の形式の例

以下は一例であるので、各自で考えてより読みやすい形式に整えること。

設問 1: 球の体積計算プログラムを作成する。

-----  
プログラムリスト  
-----

```
1 print("球の半径を入力してください\n")
2 r = gets.chomp.to_f
3 v =
4
.....
```

-----  
実行結果  
-----

```
Z:\> ruby rep1.rb
.....
.....
```

```
Z:\> ruby rep1.rb
.....
.....
```

-----  
プログラムの説明  
-----

.....  
.....  
.....