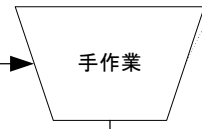


モジュール設計仕様書	システム	ほしい図書リストから一覧表を作成する	作成日	2018/10/16
補足	モジュール	ほしい図書リストから一覧表を作成する	作成者	

DFD:ほしい図書(全体像)



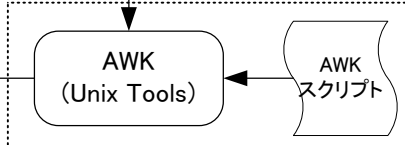
ブラウザの表示をコピーし  
テキストファイルにペースト  
する



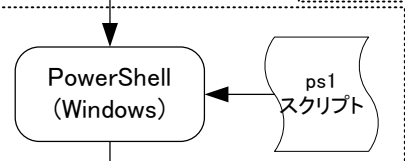
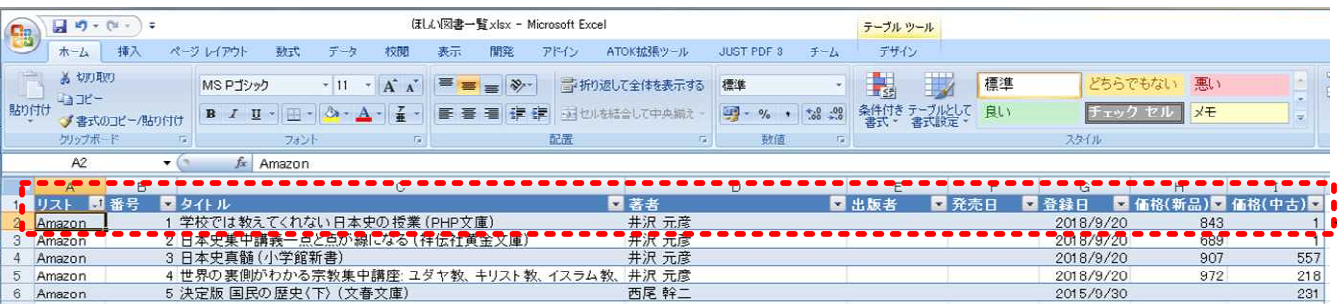
-----1件につき最大12行  
ベストセラー  
学校では教えてくれない日本史の授業 (PHP文庫) ※←オプション(ある場合あり)  
学校では教えてくれない日本史の授業 (PHP文庫)  
井沢 元彦 (文庫) ※著者名に後続の“(以)降を削除する  
5つ星のうち 4.0(38) ※←オプション(ない場合あり)  
¥843 ※←オプション(ない場合あり)  
73点の新品/中古品 ¥1より  
コメント、数量、優先度を追加する2018年9月20日に追加された商品  
カートに入れる

移動  
削除  
-----1件につき「削除」までがコピーされる

リスト番号	タイトル	著者	出版社	発売日	登録日	価格(新品)	価格(中古)
1	学校では教えてくれない日本史の授業 (PHP文庫)	井沢 元彦			2018/9/20	843	1
2	日本史集中講義一点と点が線になる (祥伝社黄金文庫)	井沢 元彦			2018/9/20	689	1
3	日本史真髓 (小学館新書)	井沢 元彦			2018/9/20	907	557
4	世界の裏側がわかる宗教集中講座: ユダヤ教、キリスト教、イスラム教、	井沢 元彦			2018/9/20	972	218
5	決定版 国民の歴史 (下) (文春文庫)	西尾 幹二			2015/9/30	231	



awk(次葉参照)



PowerShell  
(次次葉参照)

モジュール設計仕様書	システム	ほしい図書リストから一覧表を作成する	作成日	2018/10/16
補足	モジュール	ほしい図書リストからCSVファイルを作成する	作成者	

DFD:ほしい図書(awk)

### ほしい図書.txt

ベストセラー ※←オプション(ある場合あり)  
 学校では教えてくれない日本史の授業 (PHP文庫)  
 学校では教えてくれない日本史の授業 (PHP文庫)  
 井沢 元彦 (文庫) ※著者名に後続の“(以降を削除する  
 5つ星のうち 4.0(38) ※←オプション(ない場合あり)  
 ¥843 ※←オプション(ない場合あり)  
 73点の新品/中古品 ¥1より  
 コメント、数量、優先度を追加する2018年9月20日に追加された商品  
 カートに入れる

移動  
 削除  
 ...

#### 文字コード、改行コード

UNIX環境では文字コードUTF8、改行コードLFが使用される。Windows環境でファイル保存する際に上書き保存を選択しこれらを設定することができる

AWK  
 (Unix Tools)

ほしい図書.awk

#### awk -f ほしい図書 ...

環境変数AWKPATHを設定することによりスクリプトファイルのパスと拡張子を省略できる

```
#UTF8・LF
#
# Usage:awk -f ほしい図書 -v tarfile=a.txt [-v append=1] [-v startno=1] ほしい図書.txt
# ほしい図書.txt:UTF8、LF、項目の並びはAmazonのほしいものリストに準じる
# ./a.txtに図書の一覧を出力する

function tyosya(str){
  gsub("[()]*$","",str);
  return str
}

function datestr(str){
  gsub("日","",str);
  gsub("年|月","",str);
  return str
}

function yenstr(str){
  gsub("¥","",str);
  return str
}

BEGIN{
  FS="[t\n]";
  RS="削除\n";
  OFS=" ";
  ORS="\n";
  if(append!=1){system("if test -e " tarfile "; then rm -i " tarfile ";fi");}
  if(append!=1){print "リスト","番号","タイトル","著者","出版社","発売日","登録日","価格(新品)","価格(中古)" >> tarfile;}
  if(startno < 1){startno=1;}
}

# "リスト","番号","タイトル","著者","出版社","発売日"を出力する
if($0 ~ /^ベストセラー/) {printf "%s\t%d\t%s\t%s\t%s\t%s\t","Amazon",NR+startno-1,$3,tyosya($4) >> tarfile;}
else {printf "%s\t%d\t%s\t%s\t%s\t%s\t","Amazon",NR+startno-1,$2,tyosya($3) >> tarfile;}

# "登録日"を出力する
split($0,items,"[[:digit:]]+年[[:digit:]]+月[[:digit:]]+日",seps);
printf "%t%s",datestr(seps[1]) >> tarfile;

# "価格(新品)","価格(中古)"を出力する
nseps = split($0,items,"¥[[:digit:]]+*",seps);
printf "%t%s\t%s",nseps==3?yenstr(seps[1]):"",¥
yenstr(nseps==3?seps[2]:seps[1]) >> tarfile;

# レコードセパレータ(改行)を出力する
print "" >> tarfile;
}
```

#### tyosya()

“(以降の文字を削除する

#### yenstr()

“¥”を削除する

#### datestr()

“日”を削除し、  
 “年”、“月”は“/”に置き換える

RS="削除\n";

入力データのレコードセパレータを  
 “削除\n”に変更する

正規表現でデータ切り出しする

フィールドセパレータを正規表現とし、パターンマッチした文字列をデータとする

### a.txt

10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	リスト	番号	タイトル	著者	出版社	発売日	登録日	価格(新品)	価格(中古)					
2	Amazon	1	学校では教えてくれない日本史の授業 (PHP文庫)	井沢 元彦		2018/9/20	843	1						
3	Amazon	2	日本史集中講義一点と点が線になる (祥伝社黄金文庫)	井沢 元彦		2018/9/20	689	1						
4	Amazon	3	日本史真髓 (小学館新書)	井沢 元彦		2018/9/20	907	557						
5	Amazon	4	世界の裏側がわかる宗教集中講座: ユダヤ教、キリスト教、イスラム教、仏教、神道、儒教 (徳間文庫)	井沢 元彦		2018/9/20	972	218						
6	Amazon	5	決定版 国民の歴史 (下) (文春文庫)	西尾 幹二		2015/9/30	231							

モジュール設計仕様書	システム	ほしい図書リストから一覧表を作成する	作成日	2018/10/16
補足	モジュール	ほしい図書CSVファイルからExcelテーブルを作成する	作成者	

DFD:ほしい図書(PowerShell)

リスト	番号	タイトル	著者	出版社	発売日	登録日	価格(新品)	価格(中古)
Amazon	1	学校では教えてくれない日本史の授業 (PHP文庫)	井沢 元彦		2018/9/20		843	1
Amazon	2	日本史集中講義 一点と点が線になる (祥伝社黄金文庫)	井沢 元彦		2018/9/20		889	1
Amazon	3	日本史真髓 (小学館新書)	井沢 元彦		2018/9/20		907	557
Amazon	4	世界の裏側がわかる宗教集中講座: ユダヤ教、キリスト教、イスラム教、仏教、神道、儒教 (徳間文庫)	井沢 元彦		2018/9/20		972	218
Amazon	5	決定版 国民の歴史(下) (文春文庫)	西尾 幹二		2015/9/30		231	

```
#SJIS,CRLF
#
# CSVファイルをExcelにインポートする
# Usage: ..../ps/CSV2XLSX.ps1 -incsv="a.txt" -encoding=UTF8 -delimiter="¥" -otxls="ほしい図書一覧.xlsx"
#

#引数を内部変数に取得する
$opts = @{"-encoding"="Default";"-delimiter"=","} <# 省略可能オプションの初期値を設定する #>
if ($Args.Length -gt 0) {
  $Args | ForEach-Object -process { $opts[$_.Split('=')[0]]=$_.Split('=')[1] }
}
$incsv = $opts["-incsv"]
$encoding = $opts["-encoding"]
$delimiter = $opts["-delimiter"]
$otxls = $PWD.Path + "¥" + $opts["-otxls"]

# Excelを起動する
$xls = New-Object -ComObject Excel.Application
$xls.Visible = $true
$wb = $xls.WorkBooks.Add() <# WorkBookを追加する #>
$ws = $wb.WorkSheets[1] <# シートを選択する #>
$ws.name = $incsv <# シート名にCSVファイル名を設定する #>

# CSVファイルを読み込む
$csvdata = Get-Content -Path $incsv -Encoding $encoding
```

```
# 1行ずつ処理する
for($rows=0; $rows -lt $csvdata.length; $rows++) {
  # 区切り文字で項目に分割する
  $fields = ($csvdata[$rows]) -split $delimiter

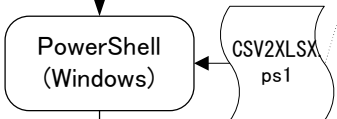
  # 1項目ずつ処理する
  for($cols=0; $cols -lt $fields.length; $cols++) {
    # Excelに書き込む
    $ws.Cells.Item($rows+1, $cols+1).Value = $fields[$cols]
  }
}

# Excelに書き込んだ部分をテーブルにする
$ws.ListObjects.Add(1, $ws.Range($ws.Cells.Item(1, 1), $ws.Cells.Item($rows, $cols)),0, 1).Name = $incsv
$ws.ListObjects($incsv).TableStyle = "TableStyleMedium2"

# 2つめ以降のシートを削除する
for($i=$wb.sheets.count;$i -gt 1;$i--){
  $ws = $wb.worksheets[$i]
  $ws.delete() > $null
}

$xls.DisplayAlerts = $false <# ファイルが既存の場合警告メッセージを表示しない #>
$wb.SaveAs($otxls) <# Excelファイルを保存する #>
$xls.DisplayAlerts = $true <# 警告メッセージの表示を元に戻す #>
$wb.Close() <# WorkBookを閉じる #>
$xls.Quit() <# Excelを終了する #>

# COM参照を解放する
[void][System.Runtime.InteropServices.Marshal]::ReleaseComObject($ws)
[void][System.Runtime.InteropServices.Marshal]::ReleaseComObject($wb)
[void][System.Runtime.InteropServices.Marshal]::ReleaseComObject($xls)
```



文字コード  
CSVファイル読み込み時にコード変換する。

