置換演算子 tr/// で日本語文字へ置換するor日本語文字を置換する

2016/03/12
がっつり忘れていたので備忘録代わりに作成。

概要

Perlの置換演算子 tr で半角英数字以外の文字を使う時の自分用お作法説明。

Perl環境とベースコード

```
$ perl -v

This is perl 5, version 20, subversion 3 (v5.20.3) built for amd64-freebsd-
thread-multi

Copyright 1987-2015, Larry Wall

Perl may be copied only under the terms of either the Artistic License or
the
GNU General Public License, which may be found in the Perl 5 source kit.

Complete documentation for Perl, including FAQ lists, should be found on
this system using "man perl" or "perldoc perl".  If you have access to the
Internet, point your browser at http://www.perl.org/, the Perl Home Page.

$ cat sample1.pl

my $str = "12345";

$str =~ tr/12345/abcde/;

print $str."\n";

$ perl sample1.pl
abcde
$
```

日本語文字が含まれる場合の対処

含まれる例

EUC-JPな環境でEUC-JPでスクリプトを書いていると、こんな感じでおかしくなる事がある。

変換先が日本語	変換元が日本語

sample2.pl
`my $str = "12345";`
 	
`$str =~ tr/12345/あいうえお/;`
 	
`print $str."\n";`
実行結果
`$ perl sample2.pl`
`あい?`
`$`
	sample3.pl
	`my $str = "あいうえお";`
	
	`$str =~ tr/あいうえお/12345/;`
	
	`print $str."\n";`
	実行結果
	`$ perl sample3.pl`
	`1211151515`
	`$`

UTF-8でスクリプトを作成する

スクリプトでutf-8を使う旨宣言し、スクリプト自体もutf-8エンコーディングで作成していればとりあえず動作はさせられる。
EUC-JP の環境で実行しているので nkf コマンドで utf-8 から euc-jp へ出力時の文字コードを変換している。

変換先が日本語	変換元が日本語

sample2utf8.pl
`use utf8;`
 	
`my $str = "12345";`
 	
`$str =~ tr/12345/あいうえお/;`
 	
`print $str."\n";`
実行結果
`$ perl sample2utf8.pl | nkf`
`Wide character in print at sample2utf8.pl line 7.`
`あいうえお`
`$`
	sample3utf8.pl
	`use utf8;`
	
	`my $str = "あいうえお";`
	
	`$str =~ tr/あいうえお/12345/;`
	
	`print $str."\n";`
	実行結果
	`$ perl sample3utf8.pl`
	`12345`
	`$`

“Wide character云々 ” は、Perl内部形式の文字列をそのまま標準出力へ流し込もうとしたから。 後述のencode()関数を使うとこんな感じ。

```
use utf8;
use Encode;

my $str = "12345";

$str =~ tr/12345/あいうえお/;

print encode('euc-jp',$str)."\n";
```

実行すると

```
$ perl sample2utf8.pl
あいうえお
```

```
$
```

Perl内部形式の文字列になっている $str をencode()関数で EUC-JP の文字列に変換している。

evalとEncode.pmで内部形式に変換

Perlはtr演算子の正規表現部分を実行前に確定させる。確定させるときにPerlがスクリプトの正規表現部分や文字列についてエンコーディングを知っていればそれを基にPerl内部形式への変換を行う。
先のUTF-8を使う例の use utf8; がそれだと思えばいい。同様に use eucjp; を使えればよいのだが残念ながらエラーになってしまう。

なので□evalを使って実行時に再評価させる。というかマニュアルにもeval使えと書いてあったりする。
評価時に処理する正規表現や文字列がPerl内部形式であればいい。

以下のサンプルスクリプトは EUC-JP で書いてある。

変換先が日本語	変換元が日本語

sample5.pl
`use Encode;`
 	
`my $str  = decode('euc-jp',"12345");`
`my $from = decode('euc-jp',"12345");`
`my $to   = decode('euc-jp',"あいうえお");`
 	
`# eval "\$str =~ tr/\$from/\$to/"; じゃ駄目だよ`
`eval "\$str =~ tr/$from/$to/";`
 	
`print encode('euc-jp',$str)."\n";`
実行結果
`$ perl sample5.pl`
`あいうえお`
`$`
	sample6.pl
	`use Encode;`
	
	`my $str  = decode('euc-jp',"あいうえお");`
	`my $from = decode('euc-jp',"あいうえお");`
	`my $to   = decode('euc-jp',"12345");`
	
	`# eval "\$str =~ tr/\$from/\$to/"; じゃ駄目だよ`
	`eval "\$str =~ tr/$from/$to/";`
	
	`print encode('euc-jp',$str)."\n";`
	実行結果
	`$ perl sample6.pl`
	`12345`
	`$`

Encodeモジュールは標準モジュール□Perl 5.18の時にはすでに標準。

decode()関数は指定のエンコーディングの文字列をPerl内部形式文字列に変換する。
encode()関数はPerl内部形式文字列を指定のエンコーディングの文字列に変換する。

euc-jpな環境でeuc-jpなスクリプトを書いたので□EUC-JP→内部形式／内部形式→EUC-JP の変換を decode/encode で行った。

技術資料, Perl, tr, 置換演算子