デバッグ

スクリプトの厳格化

曖昧な記述に対してエラーや警告を出させることで、バグの入り込む余地を減らします。これによりデバッガを使う以前に問題を発見できます。

strict

安全ではない記述をエラーにできます。

```
#!/usr/bin/perl

use strict;
```

strict - perldoc.perl.org strict - 安全ではない構文を制限する Perl プラグマ - perldoc.jp

warnings

```
use warnings;
```

warnings - perldoc.perl.org

起動オプションによる制御

```
Usage: C:¥Perl¥bin¥perl.exe [switches] [--] [programfile] [arguments]

  -0[octal]         specify record separator (¥0, if no argument)

  -a                autosplit mode with -n or -p (splits $_ into @F)

  -C[number/list]   enables the listed Unicode features

  -c                check syntax only (runs BEGIN and CHECK blocks)

  -d[:debugger]     run program under debugger

  -D[number/list]   set debugging flags (argument is a bit mask or
alphabets)

  -e program        one line of program (several -e's allowed, omit
programfile)

  -E program        like -e, but enables all optional features

  -f                don't do $sitelib/sitecustomize.pl at startup

  -F/pattern/       split() pattern for -a switch (//'s are optional)

  -i[extension]     edit <> files in place (makes backup if extension
supplied)

  -Idirectory       specify @INC/#include directory (several -I's
allowed)

  -l[octal]         enable line ending processing, specifies line
terminator

  -[mM][-]module    execute "use/no module..." before executing program

  -n                assume "while (<>) { ... }" loop around program
  -p                assume loop like -n but print line also, like sed
  -s                enable rudimentary parsing for switches after
programfile
  -S                look for programfile using PATH environment variable
  -t                enable tainting warnings
  -T                enable tainting checks
  -u                dump core after parsing program
  -U                allow unsafe operations
  -v                print version, patchlevel and license
  -V[:variable]      print configuration summary (or a single Config.pm
variable)
  -w                enable many useful warnings
  -W                enable all warnings
  -x[directory]      ignore text before #!perl line (optionally cd to
directory)
  -X                disable all warnings
```

perl --help

これらの起動オプションは、スクリプトで指定するならば、

```
#!/usr/bin/perl -W
```

のように記述します。

c オプション

実行時に c オプションを指定すると、構文チェックを行えます。

```
perl -c programfile
```

W オプション

すべての警告を有効にします。

d オプション

デバッガでプログラムを実行します。

List/search source lines

コマンド	説明
l [ln|sub]	List source code
- or .	List previous/current line
v [line]	View around line
f filename	View source in file
/pattern/ ?patt?	Search forw/backw
M	Show module versions

Debugger controls

コマンド	説明
o [...]	Set debugger options
<[<]|{[{]|>[>] [cmd]	Do pre/post-prompt
! [N|pat]	Redo a previous command
H [-num]	Display last num commands
= [a val]	Define/list an alias
h [db_cmd]	Get help on command
h h	Complete help page
|[|]db_cmd	Send output to pager
q or ^D	Quit

Control script execution

コマンド	説明
T	Stack trace
s [expr]	Single step [in expr]
n [expr]	Next, steps over subs

<CR/Enter>	Repeat last n or s
r	Return from subroutine
c [ln|sub]	Continue until position
L	List break/watch/actions
t [expr]	Toggle trace [trace expr]
b [ln|event|sub] [cnd]	Set breakpoint
B ln|*	Delete a/all breakpoints
a [ln] cmd	Do cmd before line
A ln|*	Delete a/all actions
w expr	Add a watch expression
W expr|*	Delete a/all watch exprs
![!]	syscmd Run cmd in a subprocess
R	Attempt a restart

Data Examination

コマンド	説明

expr	Execute perl code, also see: s,n,t expr
x|m expr	Evals expr in list context, dumps the result or lists methods.
p expr	Print expression (uses script's current package).
S [[!]pat]	List subroutine names [not] matching pattern
V [Pk [Vars]]	List Variables in Package. Vars can be ~pattern or !pattern.
X [Vars]	Same as "V current_package [Vars]". i class inheritance tree.
y [n [Vars]]	List lexicals in higher scope <n>. Vars same as V.
e	Display thread id
E	Display all thread ids.