Bash vs ZSH

Dennis Ruiz (darvein)

22/10/2021

Shell features

Shell feature:

- Jobs
- Input, Output and redirections
- Aliases
- History
- Completion
- Wildcard and expansiosn

config files

bash:

- /etc/profile
- /etc/bash.bashrc
- ~/.bashrc
- ~/.bash_profile
- ~/.bash_login
- ~/.profile
- ~/.bash_logout

zsh:

- /etc/zsh/zshenv
- ~/.zshenv
- /etc/zsh/zprofile
- ~/.zprofile
- /etc/zsh/zshrc
- ~/.zshrc
- /etc/zsh/zlogin
- ~/.zlogin
- ~/.zlogout
- /etc/zsh/zlogout

Global qualifiers

```
ls **/*.(dll|py)

# list the 3rd file
ls *(.[3])

# match files only, no dirs, no special files
du -h *(.)
```

```
# match executable files
du -h *(*.)

# files by size
file *(Lm+1)
ls *(.L0)

# latest two files
file git*(om[1,2])
```

```
# files by modified date
vim *(m0)
ls *(.m+2)
ls *(.^m0)

# files by permissions
ls *(.f644)

# files by permissions
ls *(.f644)
ls *(.u:apache:)
```

History in zsh

```
!!     # repeat last
!-1    # repeat previous of last
!!0    # take previous command
!^     # take first arg from last
!*     # take all params from last
```

Bash: `set -o vi`

Zsh: `bindkey -v`

ZSH builtin functions

Directories history:

- `cd -<TAB>`
- `pushd`
  - `cp file ~1`

ZMV:

`zmv '(*).jpeg' '$1.jpg'`

Long directory shortcut with `hash`:

`hash -d sh=$PWD`

Subprocess in bash and zsh

With new process:

```
$ (id)

$ cat <(date)

$ echo $date

$ ls 1> >(grep demo >output.log)
```

Without new process:

```
$ {cd ~; ls}
```

Arrays

```
print $array[2]

echo "${arr[2]}"
```

References

- ZSH Manual: https://zsh.sourceforge.io/Guide/zshguide.html
- ZSH tips: http://www.zzapper.co.uk/zshtips.html