

◀ (../06-job-control/)

Extra Unix Shell Material (../)

▶ (../08-environment-variables/)

Aliases and bash customization

? Overview

Teaching: 10 minutes min

Exercises: 0 min

Questions

- How do I customize my bash environment?

Objectives

- Create aliases.
- Add customizations to the `.bashrc` and `.bash_profile` files.
- Change the prompt in a bash environment.

Bash allows us to customize our environments to fill our own particular needs.

Aliases

Sometimes we need to use long commands that have to be typed over and over again. Fortunately, the `alias` command allows us to create shortcuts for these long commands.

As an example, let's create aliases for going up one, two, or three directories.

```
alias up='cd ..'
alias upup='cd ../..'
alias upupup='cd ../../..'
```

Let's try these commands out.

```
cd /usr/local/bin
upup
pwd
```

```
/usr
```

{ .output }

We can also remove a shortcut with `unalias`.

```
unalias upupup
```

If we create one of these aliases in a bash session, they will only last until the end of that session. Fortunately, bash allows us to specify customizations that will work whenever we begin a new bash session.

Bash customization files

Bash environments can be customized by adding commands to the `.bashrc`, `.bash_profile`, and `.bash_logout` files in our home directory. The `.bashrc` file is executed whenever entering interactive non-login shells whereas `.bash_profile` is executed for login shells. If the `.bash_logout` file exists, then it will be run after exiting a shell session.

Let's add the above commands to our `.bashrc` file.

```
echo "alias up='cd ..'" >> ~/.bashrc
tail -n 1 ~/.bashrc
```

```
alias up='cd ..'
```

We can execute the commands in `.bashrc` using `source`

```
source ~/.bashrc
cd /usr/local/bin
up
pwd
```

```
/usr/local
```

Having to add customizations to two files can be cumbersome. If we would like to always use the customizations in our `.bashrc` file, then we can add the following lines to our `.bash_profile` file.

```
if [ -f $HOME/.bashrc ]; then
    source $HOME/.bashrc
fi
```

Customizing your prompt

We can also customize our bash prompt by setting the `PS1` system variable. To set our prompt to be `$`, then we can run the command

```
export PS1="$ "
```

To set the prompt to `$` for all bash sessions, add this line to the end of `.bashrc`.

Further bash prompt customizations (<https://www.howtogeek.com/307701/how-to-customize-and-colorize-your-bash-prompt>) are possible. To have our prompt be `username@hostname[directory]:`, we would set

```
export PS1="\u@\h[\W]: "
```

where `\u` represents username, `\h` represents hostname, and `\w` represents the current directory.

! Key Points

- Aliases are used to create shortcuts or abbreviations
- The `.bashrc` and `.bash_profile` files allow us to customize our bash environment.
- The `PS1` system variable can be changed to customize your bash prompt.

◀ (../06-job-control/)

▶ (../08-environment-variables/)

Copyright © 2016 - 2019 Software Carpentry Foundation (<https://software-carpentry.org>)

Edit on GitHub (https://github.com/swcarpentry/shell-extras/edit/gh-pages/_episodes/07-aliases.md) / Source (<https://github.com/swcarpentry/shell-extras/>) / Contributing (<../contributing/>) / Cite (<../citation/>) / Contact (<mailto:lessons@software-carpentry.org>)