(/)

Curriculum
SE Foundations
Average: **108.76%**

0x03. Shell, init files, variables and expansions

DevOps **Shell** **Bash**

By: Julien Barbier

Weight: 1

Project over - took place from Feb 14, 2023 6:00 AM to Feb 15, 2023 6:00 AM

An auto review will be launched at the deadline

In a nutshell...

- **Auto QA review:** 70.0/70 mandatory & 20.0/20 optional
- **Altogether: 200.0%**
  - Mandatory: 100.0%
  - Optional: 100.0%
  - Calculation: 100.0% + (100.0% * 100.0%) == **200.0%**

About Bash projects

Unless stated, all your projects will be auto-corrected with Ubuntu 20.04 LTS.

Concepts

For this project, we expect you to look at this concept:

- Struggling with the sandbox? Try this: Using Docker & WSL on your local host (/concepts/100039)

Resources

Read or watch:

- Expansions (/rltoken/oXnzBjLBA9t9dr7WuftdmQ)
- Shell Arithmetic (/rltoken/PLSUQnBcKKU5eEgRfRDlug)
- Variables (/rltoken/SvdGNZJjKsPghzZEhaWu4Q)
- Shell initialization files (/rltoken/tqud57kjsSYgDfeZDlwl3g)
- The alias Command (/rltoken/1Z3nYPjmidqQJXcWQ9Fkug)
- Technical Writing (/rltoken/wYrZr3t3DeAE8PpYHYWGiw)

man or help:

- `printenv`
- `set`
- `unset`
- `export`
- `alias`
- `unalias`
- `.`
- `source`
- `printf`

Learning Objectives

At the end of this project, you are expected to be able to explain to anyone (/rltoken/d8LWxAXk9_gsvpPw3lCdwQ), **without the help of Google**:

General

- What happens when you type `$ ls -l *.txt`

Help

Shell Initialization Files

(/)

- What are the `/etc/profile` file and the `/etc/profile.d` directory
- What is the `~/.bashrc` file

Variables

- What is the difference between a local and a global variable
- What is a reserved variable
- How to create, update and delete shell variables
- What are the roles of the following reserved variables: HOME, PATH, PS1
- What are special parameters
- What is the special parameter `$?` ?

Expansions

- What is expansion and how to use them
- What is the difference between single and double quotes and how to use them properly
- How to do command substitution with `$()` and backticks

Shell Arithmetic

- How to perform arithmetic operations with the shell

The `alias` Command

- How to create an alias
- How to list aliases
- How to temporarily disable an alias

Other `help` pages

- How to execute commands from a file in the current shell

Copyright - Plagiarism

- You are tasked to come up with solutions for the tasks below yourself to meet with the above learning objectives.
- You will not be able to meet the objectives of this or any following project by copying and pasting someone else's work.
- You are not allowed to publish any content of this project.
- Any form of plagiarism is strictly forbidden and will result in removal from the program.

Requirements

General

- Allowed editors: `vi`, `vim`, `emacs`
- All your scripts will be tested on Ubuntu 20.04 LTS
- All your scripts should be exactly two lines long (`$ wc -l file` should print 2)
- All your files should end with a new line (why? (http://unix.stackexchange.com/questions/18743/whats-the-point-in-adding-a-new-line-to-the-end-of-a-file/18789))
- The first line of all your files should be exactly `#!/bin/bash`
- A `README.md` file, at the root of the folder of the project, describing what each script is doing
- You are not allowed to use `&&`, `||` or `;`
- You are not allowed to use `bc`, `sed` or `awk`
- All your files must be executable

More Info

Read your `/etc/profile`, `/etc/inputrc` and `~/.bashrc` files.

Look at some files in the `/etc/profile.d` directory.

Note: You do not have to learn about `awk`, `tar`, `bzip2`, `date`, `scp`, `ulimit`, `umask`, or shell scripting, yet.

Quiz questions

Great! You've completed the quiz successfully! Keep going! (Show quiz)

Tasks

0. <o> (/)

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Create a script that creates an alias.

- Name: `ls`
- Value: `rm *`

```
julien@ubuntu:/tmp/0x03$ ls
0-alias  file1  file2
julien@ubuntu:/tmp/0x03$ source ./0-alias
julien@ubuntu:/tmp/0x03$ ls
julien@ubuntu:/tmp/0x03$ \ls
julien@ubuntu:/tmp/0x03$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `0-alias`

Done! | Help | Check your code | Get a sandbox | QA Review

1. Hello you

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Create a script that prints `hello user`, where user is the current Linux user.

```
julien@ubuntu:/tmp/0x03$ id
uid=1000(julien) gid=1000(julien)
groups=1000(julien),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),113(lpadmin),128(sambashare)
julien@ubuntu:/tmp/0x03$ ./1-hello_you
hello julien
julien@ubuntu:/tmp/0x03$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `1-hello_you`

Done! | Help | Check your code | Get a sandbox | QA Review

2. The path to success is to take massive, determined action

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Add `/action` to the `PATH`. `/action` should be the last directory the shell looks into when looking for a program.

```
julien@ubuntu:/tmp/0x03$ echo $PATH
/home/julien/bin:/home/julien/.local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local
/games:/snap/bin
julien@ubuntu:/tmp/0x03$ source ./2-path
julien@ubuntu:/tmp/0x03$ echo $PATH
/home/julien/bin:/home/julien/.local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local
/games:/snap/bin:/action
julien@ubuntu:/tmp/0x03$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `2-path`

Done! | Help | Check your code | Get a sandbox | QA Review

3. If the path be beautiful, let us not ask where it leads

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Create a script that counts the number of directories in the `PATH` .

```
julien@ubuntu:/tmp/0x03$ echo $PATH
/home/julien/bin:/home/julien/.local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local
/games:/snap/bin
julien@ubuntu:/tmp/0x03$ . ./3-paths
11
julien@ubuntu:/tmp/0x03$
PATH=/home/julien/bin:/home/julien/.local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/
local/games:/snap/bin:::::/hello
julien@ubuntu:/tmp/0x03$ . ./3-paths
12
julien@ubuntu:/tmp/0x03$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `3-paths`

Done! | Help | Check your code | Get a sandbox | QA Review

4. Global variables

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Create a script that lists environment variables.

```
julien@ubuntu:/tmp/0x03$ source ./4-global_variables
CC=gcc
CDPATH=.:~:/usr/local:/usr:/
CFLAGS=-O2 -fomit-frame-pointer
COLORTERM=gnome-terminal
CXXFLAGS=-O2 -fomit-frame-pointer
DISPLAY=:0
DOMAIN=hq.garrels.be
e=
TOR=vi
FCEDIT=vi
FIGNORE=.o:~
G_BROKEN_FILENAMES=1
GDK_USE_XFT=1
GDMSESSION=Default
GNOME_DESKTOP_SESSION_ID=Default
GTK_RC_FILES=/etc/gtk/gtkrc:/nethome/franky/.gtkrc-1.2-gnome2
GWMCOLOR=darkgreen
GWMTERM=xterm
HISTFILESIZE=5000
history_control=ignoredups
HISTSIZE=2000
HOME=/nethome/franky
HOSTNAME=octarine.hq.garrels.be
INPUTRC=/etc/inputrc
IRCNAME=franky
JAVA_HOME=/usr/java/j2sdk1.4.0
LANG=en_US
LDFLAGS=-s
LD_LIBRARY_PATH=/usr/lib/mozilla:/usr/lib/mozilla/plugins
LESSCHARSET=latin1
LESS=-edfMQ
LESSOPEN=|/usr/bin/lesspipe.sh %s
LEX=flex
LOCAL_MACHINE=octarine
LOGNAME=franky
[...]
julien@ubuntu:/tmp/0x03$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `4-global_variables`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

5. Local variables

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Create a script that lists all local variables and environment variables, and functions.

```
julien@ubuntu:/tmp/0x03$ . ./5-local_variables
BASH=/bin/bash
BASHOPTS=checkwinsize:cmdhist:complete_fullquote:expand_aliases:extglob:extquote:force_fignore:histappend:interactive_comme
nts:progcomp:promptvars:sourcepath
BASH_ALIASES=()
BASH_ARGC=()
BASH_ARGV=()
BASH_CMDS=()
BASH_COMPLETION_COMPAT_DIR=/etc/bash_completion.d
BASH_LINENO=()
BASH_REMATCH=()
BASH_SOURCE=()
BASH_VERSINFO=([0]="4" [1]="3" [2]="46" [3]="1" [4]="release" [5]="x86_64-pc-linux-gnu")
BASH_VERSION='4.3.46(1)-release'
CLUTTER_IM_MODULE=xim
COLUMNS=133
COMPIZ_CONFIG_PROFILE=ubuntu
COMP_WORDBREAKS=$' \t\n"\'><=;|&(:'
DBUS_SESSION_BUS_ADDRESS=unix:abstract=/tmp/dbus-Fg27Lr20bq
DEFAULTS_PATH=/usr/share/gconf/ubuntu.default.path
DESKTOP_SESSION=ubuntu
[...]
julien@ubuntu:/tmp/0x03$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `5-local_variables`

Done! | Help | Check your code | Get a sandbox | QA Review

6. Local variable

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Create a script that creates a new local variable.

- Name: `BEST`
- Value: `School`

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `6-create_local_variable`

Done! | Help | Check your code | Get a sandbox | QA Review

7. Global variable

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Create a script that creates a new global variable.

- Name: `BEST`
- Value: `School`

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `7-create_global_variable`

Done! | Help | Check your code | Get a sandbox | QA Review

8. Every addition to true knowledge is an addition to human power

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Write a script that prints the result of the addition of 128 with the value stored in the environment variable `TRUEKNOWLEDGE`, followed by a new line.

```
julien@production-503e7013:~$ export TRUEKNOWLEDGE=1209
julien@production-503e7013:~$ ./8-true_knowledge | cat -e
1337$
julien@production-503e7013:~$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `8-true_knowledge`

Done! | Help | Check your code | Get a sandbox | QA Review

9. Divide and rule (/)

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Write a script that prints the result of `POWER` divided by `DIVIDE`, followed by a new line.

- `POWER` and `DIVIDE` are environment variables

```
julien@production-503e7013:~$ export POWER=42784
julien@production-503e7013:~$ export DIVIDE=32
julien@production-503e7013:~$ ./9-divide_and_rule | cat -e
1337$
julien@production-503e7013:~$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `9-divide_and_rule`

Done! | Help | Check your code | Get a sandbox | QA Review

10. Love is anterior to life, posterior to death, initial of creation, and the exponent of breath

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Write a script that displays the result of `BREATH` to the power `LOVE`

- `BREATH` and `LOVE` are environment variables
- The script should display the result, followed by a new line

```
julien@production-503e7013:~/$ export BREATH=4
julien@production-503e7013:~/$ export LOVE=3
julien@production-503e7013:~/$ ./10-love_exponent_breath
64
julien@production-503e7013:~/$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `10-love_exponent_breath`

Done! | Help | Check your code | Get a sandbox | QA Review

11. There are 10 types of people in the world -- Those who understand binary, and those who don't

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Write a script that converts a number from base 2 to base 10.

- The number in base 2 is stored in the environment variable `BINARY`
- The script should display the number in base 10, followed by a new line

```
julien@production-503e7013:~/$ export BINARY=10100111001
julien@production-503e7013:~/$ ./11-binary_to_decimal
1337
julien@production-503e7013:~/$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `11-binary_to_decimal`

Done! Help Check your code Get a sandbox QA Review

12. Combination

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Create a script that prints all possible combinations of two letters, except `oo`.

- Letters are lower cases, from `a` to `z`
- One combination per line
- The output should be alpha ordered, starting with `aa`
- Do not print `oo`
- Your script file should contain maximum 64 characters

```
julien@ubuntu:/tmp/0x03$ echo $((26 ** 2 -1))
675
julien@ubuntu:/tmp/0x03$ ./12-combinations | wc -l
675
julien@ubuntu:/tmp/0x03$
julien@ubuntu:/tmp/0x03$ ./12-combinations | tail -303 | head -10
oi
oj
ok
ol
om
on
op
oq
or
os
julien@ubuntu:/tmp/0x03$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `12-combinations`

Done! Help Check your code Get a sandbox QA Review

13. Floats

mandatory

Score: 100.0% (*Checks completed: 100.0%*)

Write a script that prints a number with two decimal places, followed by a new line.

The number will be stored in the environment variable `NUM`.

```
ubuntu@ip-172-31-63-244:~/0x03$ export NUM=0
ubuntu@ip-172-31-63-244:~/0x03$ ./13-print_float
0.00
ubuntu@ip-172-31-63-244:~/0x03$ export NUM=98
ubuntu@ip-172-31-63-244:~/0x03$ ./13-print_float
98.00
ubuntu@ip-172-31-63-244:~/0x03$ export NUM=3.14159265359
ubuntu@ip-172-31-63-244:~/0x03$ ./13-print_float
3.14
ubuntu@ip-172-31-63-244:~/0x03$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `13-print_float`

Done! Help Check your code Get a sandbox QA Review

14. Decimal to Hexadecimal (/)

#advanced

Score: 100.0% (*Checks completed: 100.0%*)

Write a script that converts a number from base 10 to base 16.

- The number in base 10 is stored in the environment variable `DECIMAL`
- The script should display the number in base 16, followed by a new line

```
julien@production-503e7013:~/$ export DECIMAL=16
julien@production-503e7013:~/$ ./100-decimal_to_hexadecimal
10
julien@production-503e7013:~/$ export DECIMAL=1337
julien@production-503e7013:~/$ ./100-decimal_to_hexadecimal | cat -e
539$
julien@production-503e7013:~/$ export DECIMAL=15
julien@production-503e7013:~/$ ./100-decimal_to_hexadecimal | cat -e
f$
julien@production-503e7013:~/$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `100-decimal_to_hexadecimal`

Done! | Help | Check your code | Get a sandbox | QA Review

15. Everyone is a proponent of strong encryption

#advanced

Score: 100.0% (*Checks completed: 100.0%*)

Write a script that encodes and decodes text using the rot13 encryption. Assume ASCII.

```
julien@production-503e7013:~/shell/fun_with_the_shell$ cat quote
"Everyone is a proponent of strong encryption."
- Dorothy E. Denning
julien@production-503e7013:~/shell/fun_with_the_shell$ ./101-rot13 < quote
"Rirelbar vf n cebcbarag bs fgebat rapelcgvba."
- Qbebgul R. Qraavat
julien@production-503e7013:~/shell/fun_with_the_shell$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `101-rot13`

Done! | Help | Check your code | Get a sandbox | QA Review

16. The eggs of the brood need to be an odd number

#advanced

Score: 100.0% (*Checks completed: 100.0%*)

Write a script that prints every other line from the input, starting with the first line.

```
ubuntu@ip-172-31-63-244:/$ \ls -1
bin
boot
dev
etc
home
initrd.img
lib
lib32
lib64
libx32
lost+found
media
mnt
opt
proc
root
run
sbin
srv
sys
t
#t#
t~
tmp
usr
var
vmlinuz
whoareyou
ubuntu@ip-172-31-63-244:/$ \ls -1 | ./102-odd
bin
dev
home
lib
lib64
lost+found
mnt
proc
run
srv
t
t~
usr
vmlinuz
ubuntu@ip-172-31-63-244:/$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `102-odd`

Done! | Help | Check your code | Get a sandbox | QA Review

17. I'm an instant star. Just add water and stir.

#advanced

Score: 100.0% (*Checks completed: 100.0%*)

Write a shell script that adds the two numbers stored in the environment variables `WATER` and `STIR` and prints the result.

- `WATER` is in base `water`
- `STIR` is in base `stir.`
- The result should be in base `bestchol`

```
julien@production-503e7013:~$ export WATER="ewwatratewa"
julien@production-503e7013:~$ export STIR="ti.itirtrtr"
julien@production-503e7013:~$ ./103-water_and_stir
shtbeolhc
julien@production-503e7013:~$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x03-shell_variables_expansions`
- File: `103-water_and_stir`

Done! Help Check your code Get a sandbox QA Review