



## COKE ZERO ADVENTURE

Instructions in BLUE | Concerns in RED

Commands are prefixed with \$

Additional Input is prefixed with €

Custom Names are *italicized* (Ex. *file.txt*)

Made with ❤️ by DotSlash

 **session.txt** - VERY IMPORTANT! Recorded using `$ script session.txt`

### Process Management

Show Linux distro (distributor, description, release, codename)

```
$ lsb_release -a
```

Show kernel version

```
$ uname -r
```

Show kernel (file) and its size

```
$ ls -lh /boot/vmlinuz-$(uname -r)
```

Show CPU information

```
$ cat /proc/cpuinfo
```

Run a process with lowest priority (least favorable)

```
$ nice -n 19 sleep 10
```

Run a process (> 60 sec), pause, and resume.

```
$ sleep 70
```

```
€ CTRL+Z
```

```
$ fg
```

To resume a process in background instead of foreground, run: (`$ bg`)  
Without any arguments, they operate on most recently suspended job.

Run a process (> 60 sec) in background, and terminate.

```
$ sleep 70 &
```

```
$ kill %1
```

%1 refers to the first background job started in the current shell session. To list all jobs and their number, run: (`$ jobs`)

NOTE: %1 is a reference to a job slot, NOT Process IDs (PIDs)!

Run three processes in background, and bring the second to the foreground.

```
$ sleep 70 &
```

```
$ sleep 70 &
```

```
$ sleep 70 &
```

```
$ fg %2
```

Run three processes in background, and terminate the first.

```
$ sleep 70 &  
$ sleep 70 &  
$ sleep 70 &  
$ kill %1
```

Run two processes, and connect them using a pipe.

```
$ cat /etc/passwd | grep home
```

Show uptime (elapsed time since booting)

```
$ uptime -p
```

## Memory Management

Show memory information

```
$ cat /proc/meminfo
```

Show amount of free and used memory in the system

```
$ free -h
```

## Storage Management

Create a file, rename, and delete.

```
$ touch myfile.txt  
$ mv myfile.txt samefile.txt  
$ rm samefile.txt
```

Create a folder, rename, and delete.

```
$ mkdir myfolder  
$ mv myfolder samefolder  
$ rm -r samefolder
```

Create a file and change permission so that only the owner can read

```
$ touch secrets.txt  
$ chmod 400 secrets.txt
```

Create a file and change both owner and group to "root"

```
$ touch myfile.txt  
$ sudo chown root:root myfile.txt
```

Create a folder and a symbolic link to the folder

```
$ mkdir myfolder  
$ ln -s myfolder mysymlink
```

Show disk space usage in human-readable format

```
$ df -h
```

Show file space (= size) usage of /usr/share/dict/american-english in human-readable format

```
$ du -h /usr/share/dict/american-english
```

If WAMERICAN is not yet installed, do so with apt (Ubuntu).

```
$ sudo apt install wamerican
```

Check and repair file system on hard disk

```
$ lsblk -f
```

```
$ sudo fsck -N /dev/sdX
```

FSCK opens up a whole can of worms requiring the know-how of partitions and the act of unmounting them. The -N flag only performs a DRY RUN on sdX which makes it safe to run on mounted partitions.

For additional information on FSCK please consult the MAN PAGE. **RTFM**

```
$ man fsck
```

Search for a given filename

```
$ sudo find / -name "myfile.txt"
```

The / (slash) refers to the root directory of the system, which means it will search the ENTIRE FILESYSTEM for the file. To search only within the current directory use the . (dot) in place of slash, which represents a relative path pointing to the current directory.

```
$ find . -name "myfile.txt"
```

## User Management

Change password

```
$ passwd
```

Show online users

```
$ who
```

Show current users on the system and what they are doing

```
$ w
```

Enable administrator (root or superuser) account

To enable root account on Ubuntu, run:

```
$ sudo passwd root
```

To promote a user to a superuser (give sudo access), run:

```
$ sudo usermod -aG sudo newuser
```

Add a new user and remove

```
$ sudo adduser newuser
```

```
$ sudo deluser newuser
```

adduser and deluser function as wrappers around the lower-level binaries provided by the passwd package (useradd and userdel).

## Network Management

Remote login to Linux (SSH)

```
$ ssh username@ip_address
```

Transfer a file to Linux (FTP)

```
$ ftp username@ip_address
```

Follow instructions on [cache111](#) to obtain Homosapiens! There's also an FTP snippet of session.txt provided at the end of this doc!

Show IP address of your computer

```
$ ip a
```

Show MAC address of your computer

```
$ ip link
```

The previous (`$ ip a`) also works but this is more readable.

Show IP address of google.com

```
$ dig google.com +short
```

Show connection speed to google.com

```
$ ping google.com
```

Assuming latency is what we are after.

DOWNLOAD/UPLOAD may require additional software to be installed.

Download a file at 1:00 AM tomorrow

```
$ sudo apt install at
```

```
$ sudo systemctl enable --now atd
```

```
$ echo "wget https://cache111.com/test.zip" | at 01:00
```

The above (using at) is easier, but below is the HACKER way.

It calculates the time and sleeps until then. RISKY but FUN!

```
$ nohup bash -c 'sleep $(( $(date -d "tomorrow 01:00" +%s) - $(date +%s) )) && wget https://cache111.com/test.zip' &
```

If you're connecting to a remote server, make sure to run (`$ date`).

The timezone on the server may not be the same as your current one.

Please note that the below topics are not covered in session.txt  
You people are probably comfy with the command line by now! :)

## Utilities (1)

Show the manual page of the "man" command

```
$ man man
```

Show only the first 10 lines of /etc/passwd

```
$ head -n 10 /etc/passwd
```

Show only the last 10 lines of /etc/passwd

```
$ tail -n 10 /etc/passwd
```

Show the first page of /usr/share/dict/american-english and scroll to the next

```
$ less /usr/share/dict/american-english
```

```
€ SPACE
```

```
€ q
```

Pressing SPACE will bring you to the next page, q to quit.

Show only the line containing "sys" in /etc/passwd

```
$ grep sys /etc/passwd
```

Show only the line ending with "land" in /usr/share/dict/american-english

```
$ grep 'land$' /usr/share/dict/american-english
```

Count the number of lines in /usr/share/dict/american-english

```
$ wc -l /usr/share/dict/american-english
```

Open /usr/share/dict/american-english in a text editor, and search for "microcomputer".

```
$ vim /usr/share/dict/american-english
```

```
€ /microcomputer
```

```
€ ENTER
```

```
€ :q
```

```
€ ENTER
```

Pressing / (slash) starts a search. Type microcomputer, press ENTER, and the cursor will jump to the first match. To quit type : (colon), followed by q, then ENTER. Alternatively, below is the shortcut.

Note that you'll still need to do :q to quit.

```
$ vim +'/microcomputer' /usr/share/dict/american-english
```

```
€ :q
```

```
€ ENTER
```

NOTE: If VIM is not yet installed, do so with apt (Ubuntu).

```
$ sudo apt install vim
```

NOTE: There's also nano! (easier I guess) But I'm a sucker for VIM.

Make a text file, put your firstname, save and display each character sequentially in hex.

```
$ echo "Firstname" > myname.txt
```

```
$ xxd myname.txt
```

Make a text file, put your firstname, encrypt with password protection.

```
$ echo "Firstname" > secret.txt
```

```
$ gpg -c secret.txt
```

Write a hello-world program in C, compile, and execute.

```
$ vim hello.c
```

```
€ i
```

```
#include <stdio.h>
```

```
int main(void) {
```

```
printf("Hello, world\n");
```

```
return 0;
```

```
}
```

```
€ ESC
```

```
€ :wq
```

```
€ ENTER
```

```
$ gcc -o hello hello.c
```

```
$ ./hello
```

In VIM, you start in NORMAL MODE. To type something you must go into INSERT MODE by pressing i. After typing everything in, press ESC to escape back into NORMAL MODE. Then type :wq (write & quit) followed by ENTER. Use gcc to compile hello.c into hello (binary) to execute. NOTE: If VIM is not yet installed, do so with apt (Ubuntu).

```
$ sudo apt install vim
```

NOTE: There's also nano! (easier I guess) But I'm a sucker for VIM.

Run a process that will continue after logout

```
$ nohup sleep 100 &
```

Shutdown in the next 10 minutes

```
$ sudo shutdown +10
```

Cancel the shutdown

```
$ sudo shutdown -c
```

## Utilities (2)

Make a folder with a file inside and compress it to foldername.tar.gz

```
$ mkdir myfolder
```

```
$ echo "Hello" > myfolder/myfile.txt
```

```
$ tar -czf foldername.tar.gz myfolder
```

Calculate the checksum of foldername.tar.gz

```
$ sha256sum foldername.tar.gz
```

Make a folder with a file inside and compress it to foldername.zip

```
$ mkdir myfolder
```

```
$ echo "Hello" > myfolder/myfile.txt
```

```
$ zip -r foldername.zip myfolder
```

Calculate the checksum of foldername.zip

```
$ sha256sum foldername.zip
```

Create two files, concatenate them together using "cat" ">" ">>".

```
$ echo "First File" > file1.txt
```

```
$ echo "Second File" > file2.txt
```

To overwrite/create combined.txt:

```
$ cat file1.txt file2.txt > combined.txt
```

To further append file1.txt:

```
$ cat file1.txt >> combined.txt
```

The > redirection is for overwriting.

The >> redirection is for appending at the end of the file.

Both can be used to create if the file is not present.

Split /usr/share/dict/american-english into 10 files (x00, x01, x02, ..., x09)

```
$ split -d -n 10 /usr/share/dict/american-english x
```

Show only the first column in /etc/passwd

```
$ cut -d: -f1 /etc/passwd
```

Show printer queue

```
$ lpq
```

Show current date & time

```
$ date
```

Show calendar

```
$ cal
```

The cal command in Ubuntu is supplied by the ncal package, if not yet installed, do so with apt.

```
$ sudo apt install ncal
```

## FTP Snippet from session.txt (Homo sapiens)

```
jaiyen@cokezero:~$ ftp -p ftp.broadinstitute.org
```

Connected to ftp02.broadinstitute.org.

220 FTP Server ready.

```
Name (ftp.broadinstitute.org:root): gsapubftp-anonymous
```

331 Anonymous login ok, send your complete email address as your password

```
Password:
```

230 Anonymous access granted, restrictions apply

Remote system type is UNIX.

Using binary mode to transfer files.

No entry for terminal type "xterm-ghostty";  
using dumb terminal settings.

```
ftp> cd bundle
```

250 CWD command successful

```
ftp> cd hg38
```

250 CWD command successful

```
ftp> bi
```

200 Type set to I

```
ftp> prompt
```

Interactive mode off.

```
ftp> get Homo_sapiens_assembly38.dict
```

local: Homo\_sapiens\_assembly38.dict remote:

Homo\_sapiens\_assembly38.dict

229 Entering Extended Passive Mode (|||63534|)

150 Opening BINARY mode data connection for

Homo\_sapiens\_assembly38.dict (581712 bytes)

0% | | 0 0.00 KiB/s

--:-- ETA

66% |\*\*\*\*\*| 378 KiB 378.92 KiB/s

00:00 ETA

100% |\*\*\*\*\*| 568 KiB 406.45 KiB/s

00:00 ETA

226 Transfer complete

581712 bytes received in 00:01 (351.29 KiB/s)

```
ftp> bye
```

221 Goodbye.



## **PATHS**

### ABSOLUTE PATHS

- Example: /etc/passwd
- Begins with / (slash)
- Starts from the root directory, always constant.

### RELATIVE PATHS

- Example: ./myfile.txt
- Begins with . (dot) , .. (double dot) , filename , etc.
- Depends on the current working directory (\$PWD), not constant.

### SHORTCUTS

- Example: ~ (Tilde)
- Common Misunderstanding (for me too, in the past)
- When employed, it is expanded by the shell (bash, zsh, etc.). It is not a valid path on its own.
- ~ (Tilde) is the most common one, it is user-sensitive and expands to be the home directory of the current user.
- Tilde Example: ~/file.txt may expand to /home/jaiyen/file.txt

## **Background Processes**

Regarding running background processes, you might have come across both nohup and & (ampersand) at some point. THEY ARE DIFFERENT!

### nohup

Ex. \$ nohup some\_really\_long\_task.sh

- Is prefixed at the start of the command
- Ignores hangup signal (SIGHUP)
- Runs in Foreground (Output -> nohup.out)
- TLDR; SCREEN BUSY BUT SURVIVES LOGOUT

### & (ampersand)

Ex. \$ some\_really\_long\_task.sh &

- Is appended at the end of the command
- DOES NOT ignore hangup signal (SIGHUP)
- Runs in Background (Output -> stdout)
- TLDR; SCREEN FREE BUT WILL DIE WHEN LOGOUT

### TOGETHER

Ex. \$ nohup some\_really\_long\_task.sh &

- Ignores hangup signal (SIGHUP)
- Runs in Background (Output -> nohup.out , if not redirected)
- TLDR; SCREEN FREE AND SURVIVES LOGOUT

IF YOU NOTICE ANY MISTAKES PLEASE NOTIFY ME OR OUR COLLEAGUES  
I DOUBLE CHECKED BUT AM NOT IMMUNE TO MISTAKES AND SKILL ISSUES