

## Iterate with "for"

```
$ for i in Europe/London Europe/Paris Europe/Madrid ; do
>   showtime America/Los_Angeles $i 11:45
> done
America/Los_Angeles 11:45 is Europe/London Fri 21 Feb 2020 07:45:00 PM GMT
America/Los_Angeles 11:45 is Europe/Paris Fri 21 Feb 2020 08:45:00 PM CET
America/Los_Angeles 11:45 is Europe/Madrid Fri 21 Feb 2020 08:45:00 PM CET
$ TZONES='America/Los_Angeles America/New_York Europe/London'
$ for tzin in $TZONES ; do # For all input time zones
>   for tzout in $TZONES ; do # For all output time zones
>     showtime $tzin $tzout 11:45 # Show the time
>   done
> done
America/Los_Angeles 11:45 is America/Los_Angeles Fri 21 Feb 2020 11:45:00 AM PST
America/Los_Angeles 11:45 is America/New_York Fri 21 Feb 2020 02:45:00 PM EST
America/Los_Angeles 11:45 is Europe/London Fri 21 Feb 2020 07:45:00 PM GMT
America/New_York 11:45 is America/Los_Angeles Fri 21 Feb 2020 08:45:00 AM PST
America/New_York 11:45 is America/New_York Fri 21 Feb 2020 11:45:00 AM EST
America/New_York 11:45 is Europe/London Fri 21 Feb 2020 04:45:00 PM GMT
Europe/London 11:45 is America/Los_Angeles Fri 21 Feb 2020 03:45:00 AM PST
Europe/London 11:45 is America/New_York Fri 21 Feb 2020 06:45:00 AM EST
Europe/London 11:45 is Europe/London Fri 21 Feb 2020 11:45:00 AM GMT
```

## Conditionals with "if"

```
$ touch sourcefile # Create two files; source is older than destination
$ touch destfile
$ if test sourcefile -nt destfile ; then # Refresh if needed
>   cp sourcefile destfile
>   echo Refreshed destfile
> fi
$ touch sourcefile # Make source newer than destination
$ if [ sourcefile -nt destfile ] ; then # Alternative test syntax
>   cp sourcefile destfile
>   echo Refreshed destfile
> fi
Refreshed destfile
$ if [ sourcefile -nt destfile ] ; then
>   cp sourcefile destfile
>   echo Refreshed destfile
> else
>   echo destfile is up to date
> fi
destfile is up to date
$
```

## Loops with "while"

```
$ cd /etc
$ ls | # List all directory entries
> while read name ; do # For every entry
>   if [ -f $name -a -r $name ] ; then # If regular file and readable
>     echo -n "$name " # Display its name
>     expr $(wc -c <$name) / $(wc -l <$name) # Display average characters per line
>   fi
> done |
> head # Display first ten lines
adduser.conf 35
bash.bashrc 34
bash_completion 45
bindresvport.blacklist 25
```

```
ca-certificates.conf 42
colord.conf 31
crontab 48
debconf.conf 35
debian_version 4
deluser.conf 30
$
```

## Apply command with "xargs"

```
$ find . -type f | # Output the name of all files
> xargs cat | # Combine them by applying cat
> wc -l # Count number of lines
546986
$ find '/cygdrive/c/Program Files' -type f | # Output names under program files
> xargs stat -c '%Y %n' | # Output modification time (s) and name
> sort -n | # Output in numeric order
> head -1 # Output oldest one
stat: ./Application: No such file or directory
stat: Verifier/vrfauto.dll: No such file or directory
stat: ./Application: No such file or directory
$ find '/cygdrive/c/Program Files' -type f -print0 | # Terminate names with \0
> xargs -0 stat -c '%Y %n' | # Process input terminated with \0
> sort -n |
> head -1
947862022 ./Ghostgum/gsview/LICENCE
$ ls -l ./Ghostgum/gsview/LICENCE
-rw-r--r-- 1 dds None 11173 Jan 14 2000 ./Ghostgum/gsview/LICENCE
$
```

## Select with "case"

```
$ uname # Operating system details
Linux
$ case $(uname) in
> Linux)
> alias s=gnome-open
> alias cpt='pwd | xsel --clipboard'
> ;;
> Darwin)
> alias s=open
> alias cpt='pwd | pbcopy'
> ;;
> CYGWIN*)
> alias s=cygstart
> alias cpt='cygpath -w `pwd` | tr -d \\n >/dev/clipboard'
> ;;
> esac
$ alias
alias cpt='pwd | xsel --clipboard'
alias s='gnome-open'
$
```