

Prefer redirection to pipes

```
$ cat file | command # Wasteful execution of cat
$ command <file # A redirection is all that's needed
$
```

Test command, not its exit code

```
$ command
$ if [ $? -ne 0 ] ; then # Verbose exit variable test
>   echo Error >&2
> fi
$ if ! command ; then # A simple negation will do
>   echo Error >&2
> fi
$
```

Use the sed and awk predicates

```
$ grep pattern | awk '{ ... }' # Unneeded use of grep
$ awk '/pattern/ { ... }' # Simply prepend pattern
$ grep pattern | sed '...' # Unneeded use of grep
$ sed '/pattern/ { ... }' # Simply prepend pattern
$
```

Grep can recurse directories

```
$ grep pattern afile | wc -l # Count matches
$ grep -c pattern afile # Modern count matches
$ find . -type f | xargs grep pattern # Recursive search
$ grep -r pattern . # Modern recursive search
$
```

Prefer wildcards to ls

```
$ echo $(ls) # This is the same
$ ls # As a simple invocation
$ for i in $(ls) ; do # The ls here
>   ...
> done
$ for i in * ; do # can be replaced by a wildcard
>   ...
> done
$
```

Replace awk with cut

```
$ head -2 /etc/passwd |
> awk -F: '{print $1, $7}' # Print fields 1 and 7
root /bin/bash
daemon /usr/sbin/nologin
$ head -2 /etc/passwd |
> cut -d : -f 1,7 # More efficient way to print fields 1 and 7
root:/bin/bash
daemon:/usr/sbin/nologin
$
```

Replace sed with expr

```
$ echo $LANG
en_US.UTF-8
$ echo $LANG |
```

```
> sed 's/.*\.\(.*\)\/\1/' # Isolate encoding
UTF-8
$ expr "$LANG" : '.*\.\(.*\)' # More efficient way to isolate encoding
UTF-8
$
```

Process find's output

```
$ ls -ld */core # Find files named core; might not fit
$ find . -name core | # Find files named core
> while read filename ; do
>   ls -ld "$filename"
> done
$ find . -name core \
> -exec ls -ld '{}' \; # Execute ls for each found file
$ find . -name core -print0 |
> xargs -0 ls -ld # Execute ls in batches
$
```