

# 1 Shell CheatSheet

## LANGUAGES

- PDF Link: [cheatsheet-shell-A4.pdf](#), Category: languages
- Blog URL: <https://cheatsheet.dennyzhang.com/cheatsheet-shell-A4>
- Related posts: Vim CheatSheet, #denny-cheatsheets

File me Issues or star this repo.

## 1.1 Basic

| Name                                   | Comment  |
|--|--|
| Redirect stdout/stderr                 | <code>ls /tmp &gt;/dev/null 2&gt;&amp;1</code>                                     |
| Deal with filename                     | <code>basename \$f, dirname \$f</code>   |
| Use timeout: avoid command hang        | <code>timeout 10 sh -c 'ls -lt'</code>   |
| Restart shell without killing terminal | <code>exec -l \$SHELL</code>   |
| Run sub-shell                          | <code>echo \$BASH_SUBSHELL; ( echo "Running in subshell: \$BASH_SUBSHELL" )</code> |
| Set pipefail                           | <code>set -ueoxv pipefail, set +ueoxv pipefail</code>                              |
| Run static code check                  | Link: <a href="#">shellcheck</a>   |

## 1.2 Shell script

| Name                                   | Comment   |
|--|---|
| Trap exit signal                       | <code>code/trap-exit.sh</code>  |
| Shell retry                            | <code>code/shell-retry.sh</code>  |
| Check if a string contains a substring | <code>code/string-contains.sh</code>                                      |
| Check if a string in a list            | <code>code/string-in-list.sh</code> , Link: <a href="#">stackoverflow</a> |
| Log with timestamp                     | <code>code/log-with-timestamp.sh</code>                                   |
| Quit if current user is not root       | <code>code/assert-user-root.sh</code>                                     |
| Set -x on fly                          | <code>code/restore-debug-output.sh</code>                                 |

## 1.3 Environment variables

| Name                           | Comment                            |
|--------------------------------|------------------------------------|
| List all environment variables | <code>export</code>                |
| Define a new env variable      | <code>export NAME1="value1"</code> |

## 1.4 zsh

| Name                          | Comment  |
|-------------------------------|--|
| Disable all zsh's autocorrect | In <code>~/.zshrc</code> , <code>unsetopt correct_all</code>   |
| Disable a given autocorrect   | In <code>~/.zshrc</code> , alias <code>ssh='nocorrect ssh'</code> . <code>zsh<sub>disable</sub></code> |

## 1.5 GNU tools

### 1.5.1 Echo string

| Name                 | Comment  |
|----------------------|--|
| Echo red text        | <code>echo -e "hello,[0;31m there [0;31m"</code>                     |
| Echo multiple lines  | <code>echo -e "hello,"</code>  |
| Echo bold text       | <code>echo -e hello, "\033[1mThis is bold text.\033[0m"</code>       |
| Echo underlined text | <code>echo -e hello, "\033[4mThis is underlined text.\033[0m"</code> |

## 1.6 Shell Basic

### 1.6.1 cd

| Name                     | Comment  |
|--------------------------|--|
| Go to given folder       | <code>cd /var/log/</code>  |
| Go to folder in subshell | <code>(cd /var/log/ &amp;&amp; ls)</code> After this, PWD won't be changed |
| Go to home               | <code>cd</code>  |
| Go to parent folder      | <code>cd ..</code>   |
| Go to previous folder    | <code>cd -</code>  |

### 1.6.2 Numeric

| Name               | Comment  |
|--------------------|--|
| <code>*</code>     | <code>expr 5 \* 4</code>   |
| <code>+</code>     | <code>let z=x+y, z=\$((x+y))</code>                              |
| <code>==</code>    | <code>int1 -eq int2, [ \$? -eq 0 ] &amp;&amp; echo "good"</code> |
| <code>&gt;=</code> | <code>int1 -ge =int2</code>                                      |
| <code>&gt;</code>  | <code>int1 -gt =int2</code>                                      |
| <code>&lt;=</code> | <code>int1 -le =int2</code>                                      |
| <code>&lt;</code>  | <code>int1 -lt =int2</code>                                      |
| <code>!=</code>    | <code>int1 -ne =int2</code>                                      |

### 1.6.3 xargs

```
# Run grep for files filtered by find
find /var/log -name "*.log" | xargs grep -i error
```

```
# Loop with pipes
cat /etc/passwd | awk -F':' '{print $1}' | xargs -I{} sudo -l -U {} | grep -v "not allowed to"
```

## 1.7 Scripts

- Compare command output

```
[ 0 -eq $(find ./data -name "*.txt" -type f -print | wc -l) ]
```

- get ip from eth0

```
/sbin/ifconfig eth0 | grep 'inet addr:' | cut -d: -f2 | awk '{ print $1}'
```

## 1.8 More Resources

License: Code is licensed under MIT License.