

ICAT Linux CheatSheet

Permissions

Ingredience

short	long	file	dir
r	read	read	is shown
w	write	write	create stuff in it
x	execute	execute it	enter it

ls-command

```
-rw-r--r-- 1 kniepbert staff 10 4 Aug 08:24 unit1.pdf
```

Misc

Oneliner

To concatenate commands in one line use **&&** and ;
`cmd1 && cmd2` fires cmd2 if cmd1 was successfull
`cmd1 ; cmd2` fires cmd2 whatever result cmd1 has

Errorcode

Every command gives back an Errorcode between 0-255. If everything went alright its 0. Its stored in the Variable **\$?**.

```
$ touch test ; echo $?  
0  
$ ls test ; echo $?  
ls: test: No such file or directory  
1
```

Within your own script you could create an errorcode with
exit VALUE

variables

You could assign values or stdout to variables. Don't use blanks!

```
$ var=1 ; echo ${var}  
1  
$ var = 1  
-bash: var: command not found  
$ var="Hello World" ; echo ${var}  
Hello World  
$ ls  
unit1.tex unit2.tex unit3.tex  
$ var='ls '  
$ echo ${var}  
unit1.tex unit2.tex unit3.tex
```

Compare

if-then-else

To check a condition and act according to the result use if-then-else:

```
if [ CONDITION ]  
then  
    CONSEQUENCE  
else
```

ALTERNATIVE

```
fi
```

if there is only one command after then/else

```
if [ CONDITION ]  
then CONSEQUENCE  
else ALTERNATIVE  
fi
```

for linebreaks use ;

```
if [ CONDITION ]; then cmd1; else cmd2; fi
```

conditions

use **test** to check and compare. Have a look at the manpage to list all possible checks.

```
$ if [ -e test ];then echo '1';else echo '0';fi  
no  
$ touch test  
$ if [ -e test ];then echo '1';else echo '0';fi  
yes  
$ if [ 1 -eq 1 ];then echo '1';else echo '0';fi  
yes  
$ if [ 2 -eq 1 ];then echo '1';else echo '0';fi  
no  
$ if [ 'Hi'=='Hi' ];then echo '1';else echo '0';fi  
yes
```