

Test file properties

```
$ test -d / && echo Directory # Test if directory
Directory
$ test -f / && echo File # Test if file
$ test -f /etc/passwd && echo File # Test if file
File
$ test hi = there && echo Same # Test if strings equal
$ test hi = hi && echo Same
Same
$ test hi != there && echo Different # Test if strings different
Different
$ test -z "" && echo Empty # Test if string empty
Empty
$ test -n 'a string' && echo Non-empty # Test if string non-empty
Non-empty
$ test 32 -eq 42 && echo Equal # Test if integers equal
$ test 42 -eq 42 && echo Equal # Test if integers equal
Equal
$ test 32 -lt 50 && echo Less than # Test if integer less than other
Less than
$ test . -nt / && echo . is newer than / # Test if file newer than other
. is newer than /
$ test -w / && echo Writable # Test if writable
Writable
$ [ -w / ] && echo Writable # Synonym for test
Writable
$ if [ -d /etc/bash_completion.d ] ; then # Script use
>   echo $(ls /etc/bash_completion.d | wc -l) completion scripts installed
> fi
5 completion scripts installed
$
```

Evaluate expressions

```
$ expr 1 + 2 # Add
3
$ expr 10 - 2 # Subtract
8
$ expr 2 \* 10 # Multiply (escape special character)
20
$ expr 12 \% 5 # Remainder
2
$ expr 10 \< 50 # Compare numbers
1
$ expr 5 = 12 # Test of equality
0
$ expr John \> Mary # Compare strings
0
$ expr 1 + 20 \* 2 # Operator precedence is the usual one
41
$ expr \( 1 + 20 \) \* 2 # Use brackets to change it
42
$ expr 'To be or not to be' : '[^]*' # Chars matched by regular expression
2
$ expr 'To be or not to be' : '\([^]*\)' # Matched part
To
$ expr length 'To be or not to be' # String length
18
$ expr substr 'To be or not to be' 4 2 # Substring of 2 from 4
be
$ expr '' \| b # Short-circuit OR (first part failed)
```

```
b
$ expr 0 \| b # Short-circuit OR (first part failed)
b
$ expr 0 \& b # Short-circuit AND (first part failed)
0
$ expr a \& b # Short-circuit AND (first part succeeded)
a
$
```

Shell built-ins

```
$ i=0
$ while [ $i -lt 10 ] ; do
>   echo $i
>   i=$(expr $i + 1)
> done
0
1
2
3
4
5
6
7
8
9
$ i=0
$ while [[ i -lt 10 ]] ; do
>   echo $i
>   i=$((i + 1))
> done
0
1
2
3
4
5
6
7
8
9
$ i=0
$ time while [ $i -lt 1000 ] ; do   i=$(expr $i + 1); done

real    0m3.255s
user    0m0.663s
sys     0m1.474s
$ i=0
$ time while [[ $i -lt 1000 ]] ; do   i=$((i + 1)); done

real    0m0.042s
user    0m0.032s
sys     0m0.000s
$
```