

Bash conditional statements

WE'LL COVER THE FOLLOWING ^

- Bash if-else
- Bash `case` statement

Bash if-else

To compare integers we use can the following operators:

Operator	Description
<code>-eq</code>	Equal to
<code>-ne</code>	Not equal to
<code>-gt</code>	Greater than
<code>-ge</code>	Greater than OR equal to
<code>-lt</code>	less than
<code>-le</code>	less than OR equal to

The following example shows how to use the number comparison operators in an if statement.

main.sh

```
bash bash-ifelse.sh 0
```





Bash `elif` is short for `else if` which can allow us to select one of many blocks of code to execute by testing two or more conditional expressions. The `If-elif-else` syntax is given below:

```
if [ conditional expression1 ]
then
    statement 1
    statement 2
    ..
elif [ conditional expression2 ]
then
    statement 3
    statement 4
    ..
else
    statement 5
fi
```

Bash file test operators:

File Test	Operator Description
<code>-e</code>	File exists (this could be regular file, directory, block device, character device, etc.,)
<code>-f</code>	It's a regular file (for example: <code>/etc/shadow</code>)
<code>-d</code>	It's a directory (for example: <code>/var</code>)
<code>-b</code>	It's a block device (for example: <code>/dev/sdb</code>)

`-c`

It's a character device (for example: `/dev/tty2`)

`-s`

File is not empty

`-r`

File read permission

`-w`

File write permission

`-x`

File execute permission

`-u`

suid set on the file

`-g`

sgid set on the file

`-k`

Sticky bit set on the file

`-p`

It's a pipe

`-S`

It's a socket

`-h`

It's a symbolic link

`-t`

Checks whether the given FD is opened in a terminal.

`-O`

You own the file

`-G`

File group id and my group id are the same.

`-N`

Did the file got modified since last read?

`file-a -nt file-b`

file-a is newer than file-b

`file-a -ot file-b`

File1 is older than file2

`file-a -ef file-b`

Both file1 and file2 are hard linked
to the same file





Example:

main.sh
bashdircheck.sh

```
#!/bin/bash

directory=$1

# bash check if directory exists
if [ -d $directory ]; then
    echo "Directory exists!"
else
    echo "Directory does not exists!"
fi
```



Bash `case` statement

In the following Bash `case` statements, if value of the “var” matches “pattern1”, it will execute `command1`, `command2`, and any other commands in the “pattern1” block.

```
case var in
    pattern1 )
        command1
        command2
        ...
    ;;
    pattern2 )
        command3
        command4
        ...
    ;;
esac
```

Example:

main.sh

```
bash bash-case.sh
```



bash-case.sh

