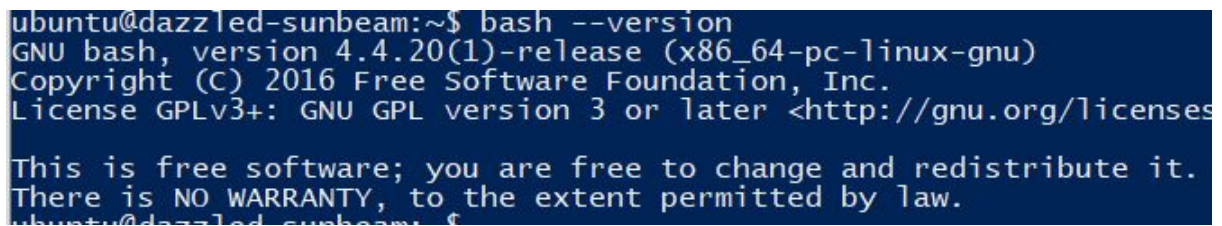


# Bash scripting

```
$ bash --version
```

A terminal window with a dark blue background and white text. The prompt is 'ubuntu@dazzled-sunbeam:~\$'. The command 'bash --version' has been entered. The output shows the GNU bash version (4.4.20(1)-release), copyright information (© 2016 Free Software Foundation, Inc.), the license (GPLv3+), and a URL to the GNU GPL license. It also includes a statement about free software and warranty.

```
ubuntu@dazzled-sunbeam:~$ bash --version
GNU bash, version 4.4.20(1)-release (x86_64-pc-linux-gnu)
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses
This is free software; you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
ubuntu@dazzled-sunbeam:~$
```

## Common bash commands:

```
$ ls - List directory contents
$ echo - Prints text to the terminal windows
$ touch - Creates a file
$ mkdir - Create a directory
$ pwd - Print working directory
$ cd - Change directory
$ mv - Move or rename directory
$ less - View the contents of a text file (no edit)
$ cat - Read a file, create a file and concatenate files
$ chmod - Sets the file permissions flag on a file or folder
$ exit - Closes terminal, log you out of remote SSH access session, end
execution of a shell script
$ history - List your most recent commands
$ clear - Clear your terminal window
$ cp - Copy files and directories
$ kill - Terminate stalled processes
```

### Første script:

\$ nano hello.sh

```
#!/bin/bash
echo "Hello World"
```

\$ bash hello.sh

Hello World

### Brug af comment:

\$ nano comment.sh

```
#!/bin/bash

# Add two numeric value
((sum=25+35))

#Print the result
echo $sum
```

\$ bash comment.sh

60

### While loop:

\$ nano while.sh

```
#!/bin/bash
valid=true
count=1
while [ $valid ]
do
echo $count
if [ $count -eq 5 ];
then
break
fi
((count++))
done
```

\$ bash while.sh

1  
2  
3  
4  
5

### For loop:

\$ nano for.sh

```
#!/bin/bash
for (( counter=10; counter>0; counter-- ))
do
echo -n "$counter "
done
printf "\n"
```

\$ bash for.sh

10 9 8 7 6 5 4 3 2 1

### Get User Input:

\$ nano user\_input.sh

```
#!/bin/bash
echo "Enter Your Name"
read name
echo "Welcome $name to bash scripting"
```

\$ bash user\_input.sh

Enter Your Name

Mads

Welcome Mads to bash scripting

### If statement:

\$ nano simple\_if.sh

```
#!/bin/bash
n=10
if [ $n -lt 10 ];
then
echo "It is an one digit number"
else
echo "It is a two digit number"
fi
```

\$ bash simple\_if.sh

It is a two digit number