

Изучите принципы действия команд

```
WHICH(1)                                General Commands Manual                                WHICH(1)

NAME
    which - locate a command

SYNOPSIS
    which [-a] filename ...

DESCRIPTION
    which returns the pathnames of the files (or links) which would be executed in the current environment, had
    its arguments been given as commands in a strictly POSIX-conformant shell. It does this by searching the PATH
    for executable files matching the names of the arguments. It does not canonicalize path names.

OPTIONS
    -a    print all matching pathnames of each argument

EXIT STATUS
    0      if all specified commands are found and executable
    1      if one or more specified commands is nonexistent or not executable
    2      if an invalid option is specified

Debian                                     29 Jun 2016                                     WHICH(1)
Manual page which(1) line 1/24 (END) (press h for help or q to quit)
```

```
evgentoph@DESKTOP-TH92VI3: ~
PS(1)                                User Commands                                PS(1)

NAME
    ps - report a snapshot of the current processes.

SYNOPSIS
    ps [options]

DESCRIPTION
    ps displays information about a selection of the active processes. If you want a repetitive update of the
    selection and the displayed information, use top instead.

    This version of ps accepts several kinds of options:

    1  UNIX options, which may be grouped and must be preceded by a dash.
    2  BSD options, which may be grouped and must not be used with a dash.
    3  GNU long options, which are preceded by two dashes.

    Options of different types may be freely mixed, but conflicts can appear. There are some synonymous options,
    which are functionally identical, due to the many standards and ps implementations that this ps is compatible
    with.

    Note that ps -aux is distinct from ps aux. The POSIX and UNIX standards require that ps -aux print all
    processes owned by a user named x, as well as printing all processes that would be selected by the -a option.
    If the user named x does not exist, this ps may interpret the command as ps aux instead and print a warning.
    This behavior is intended to aid in transitioning old scripts and habits. It is fragile, subject to change,
    and thus should not be relied upon.

    By default, ps selects all processes with the same effective user ID (euid=EUID) as the current user and

Manual page ps(1) line 1 (press h for help or q to quit)
```

```

KILL(1)                                     User Commands                                     KILL(1)

NAME
    kill - send a signal to a process

SYNOPSIS
    kill [options] <pid> [...]

DESCRIPTION
    The default signal for kill is TERM. Use -l or -L to list available signals. Particularly useful signals include HUP, INT, KILL, STOP, CONT, and 0. Alternate signals may be specified in three ways: -9, -SIGKILL or -KILL. Negative PID values may be used to choose whole process groups; see the PGID column in ps command output. A PID of -1 is special; it indicates all processes except the kill process itself and init.

OPTIONS
    <pid> [...]
        Send signal to every <pid> listed.

    -<signal>
    -s <signal>
    --signal <signal>
        Specify the signal to be sent. The signal can be specified by using name or number. The behavior of signals is explained in signal(7) manual page.

    -q, --queue <value>
        Use sigqueue(3) rather than kill(2) and the value argument is used to specify an integer to be sent with the signal. If the receiving process has installed a handler for this signal using the SA_SIGINFO flag to sigaction(2), then it can obtain this data via the si_value field of the siginfo_t structure.

Manual page kill(1) line 1 (press h for help or q to quit)

```

```

LN(1)                                       User Commands                                       LN(1)

NAME
    ln - make links between files

SYNOPSIS
    ln [OPTION]... [-I] TARGET LINK_NAME
    ln [OPTION]... TARGET
    ln [OPTION]... TARGET... DIRECTORY
    ln [OPTION]... -t DIRECTORY TARGET...

DESCRIPTION
    In the 1st form, create a link to TARGET with the name LINK_NAME. In the 2nd form, create a link to TARGET in the current directory. In the 3rd and 4th forms, create links to each TARGET in DIRECTORY. Create hard links by default, symbolic links with --symbolic. By default, each destination (name of new link) should not already exist. When creating hard links, each TARGET must exist. Symbolic links can hold arbitrary text; if later resolved, a relative link is interpreted in relation to its parent directory.

    Mandatory arguments to long options are mandatory for short options too.

    --backup[=CONTROL]
        make a backup of each existing destination file

    -b
        like --backup but does not accept an argument

    -d, -F, --directory
        allow the superuser to attempt to hard link directories (note: will probably fail due to system restrictions, even for the superuser)

Manual page ln(1) line 1 (press h for help or q to quit)

```

```
evgentoph@DESKTOP-TH92VI3: ~
STAT(1)                                User Commands                                STAT(1)

NAME
    stat - display file or file system status

SYNOPSIS
    stat [OPTION]... FILE...

DESCRIPTION
    Display file or file system status.

    Mandatory arguments to long options are mandatory for short options too.

    -L, --dereference
        follow links

    -f, --file-system
        display file system status instead of file status

    --cached=MODE
        specify how to use cached attributes; useful on remote file systems. See MODE below

    -c --format=FORMAT
        use the specified FORMAT instead of the default; output a newline after each use of FORMAT

    --printf=FORMAT
        like --format, but interpret backslash escapes, and do not output a mandatory trailing newline; if you
        want a newline, include \n in FORMAT

Manual page stat(1) line 1 (press h for help or q to quit)
```

```
evgentoph@DESKTOP-TH92VI3: ~
PING(8)                                iputils                                PING(8)

NAME
    ping - send ICMP ECHO_REQUEST to network hosts

SYNOPSIS
    ping [-aAbBdDfHhLnOqrRUVvV46] [-c count] [-F flowlabel] [-i interval] [-I interface] [-l preload] [-m mark]
        [-M pmtudisc option] [-N nodeinfo option] [-w deadline] [-W timeout] [-p pattern] [-Q tos]
        [-s packetsize] [-S sndbuf] [-t ttl] [-T timestamp option] [hop...] {destination}

DESCRIPTION
    ping uses the ICMP protocol's mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or
    gateway. ECHO_REQUEST datagrams ("pings") have an IP and ICMP header, followed by a struct timeval and then an
    arbitrary number of "pad" bytes used to fill out the packet.

    ping works with both IPv4 and IPv6. Using only one of them explicitly can be enforced by specifying -4 or -6.

    ping can also send IPv6 Node Information Queries (RFC4620). Intermediate hops may not be allowed, because IPv6
    source routing was deprecated (RFC5095).

OPTIONS
    -4
        Use IPv4 only.

    -6
        Use IPv6 only.

    -a
        Audible ping.

Manual page ping(8) line 1 (press h for help or q to quit)
```

CHMOD(1)

User Commands

CHMOD(1)

NAME

chmod - change file mode bits

SYNOPSIS

```
chmod [OPTION]... MODE[,MODE]... FILE...
chmod [OPTION]... OCTAL-MODE FILE...
chmod [OPTION]... --reference=RFILE FILE...
```

DESCRIPTION

This manual page documents the GNU version of chmod. chmod changes the file mode bits of each given file according to mode, which can be either a symbolic representation of changes to make, or an octal number representing the bit pattern for the new mode bits.

The format of a symbolic mode is [ugoa...][[-+]=[perms...].], where perms is either zero or more letters from the set rwxXst, or a single letter from the set ugo. Multiple symbolic modes can be given, separated by commas.

A combination of the letters ugoa controls which users' access to the file will be changed: the user who owns it (u), other users in the file's group (g), other users not in the file's group (o), or all users (a). If none of these are given, the effect is as if (a) were given, but bits that are set in the umask are not affected.

The operator + causes the selected file mode bits to be added to the existing file mode bits of each file; - causes them to be removed; and = causes them to be added and causes unmentioned bits to be removed except that a directory's unmentioned set user and group ID bits are not affected.

The letters rwxXst select file mode bits for the affected users: read (r), write (w), execute (or search for

Manual page chmod(1) line 1 (press h for help or q to quit)

DD(1)

User Commands

DD(1)

NAME

dd - convert and copy a file

SYNOPSIS

```
dd [OPERAND]...
dd OPTION
```

DESCRIPTION

Copy a file, converting and formatting according to the operands.

bs=BYTES

read and write up to BYTES bytes at a time (default: 512); overrides ibs and obs

cbs=BYTES

convert BYTES bytes at a time

conv=CONVS

convert the file as per the comma separated symbol list

count=N

copy only N input blocks

ibs=BYTES

read up to BYTES bytes at a time (default: 512)

if=FILE

read from FILE instead of stdin

Manual page dd(1) line 1 (press h for help or q to quit)

```
evgentoph@DESKTOP-TH92VI3: ~
CHOWN(1)                                User Commands                                CHOWN(1)

NAME
    chown - change file owner and group

SYNOPSIS
    chown [OPTION]... [OWNER][:[GROUP]] FILE...
    chown [OPTION]... --reference=RFILE FILE...

DESCRIPTION
    This manual page documents the GNU version of chown. chown changes the user and/or group ownership of each
    given file. If only an owner (a user name or numeric user ID) is given, that user is made the owner of each
    given file, and the files' group is not changed. If the owner is followed by a colon and a group name (or nu-
    meric group ID), with no spaces between them, the group ownership of the files is changed as well. If a colon
    but no group name follows the user name, that user is made the owner of the files and the group of the files
    is changed to that user's login group. If the colon and group are given, but the owner is omitted, only the
    group of the files is changed; in this case, chown performs the same function as chgrp. If only a colon is
    given, or if the entire operand is empty, neither the owner nor the group is changed.

OPTIONS
    Change the owner and/or group of each FILE to OWNER and/or GROUP. With --reference, change the owner and
    group of each FILE to those of RFILE.

    -c, --changes
        like verbose but report only when a change is made

    -f, --silent, --quiet
        suppress most error messages

Manual page chown(1) line 1 (press h for help or q to quit)
```

```
evgentoph@DESKTOP-TH92VI3: ~
YES(1)                                User Commands                                YES(1)

NAME
    yes - output a string repeatedly until killed

SYNOPSIS
    yes [STRING]...
    yes OPTION

DESCRIPTION
    Repeatedly output a line with all specified STRING(s), or 'y'.

    --help display this help and exit

    --version
        output version information and exit

AUTHOR
    Written by David MacKenzie.

REPORTING BUGS
    GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
    Report any translation bugs to <https://translationproject.org/team/>

COPYRIGHT
    Copyright © 2020 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later
    <https://gnu.org/licenses/gpl.html>.
    This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent per-
    mitted by law.

Manual page yes(1) line 1 (press h for help or q to quit)
```

```
evgentoph@DESKTOP-TH92VI3: ~
uname(1)                                User Commands                                uname(1)

NAME
    uname - print system information

SYNOPSIS
    uname [OPTION]...

DESCRIPTION
    Print certain system information.  With no OPTION, same as -s.

    -a, --all
        print all information, in the following order, except omit -p and -i if unknown:

    -s, --kernel-name
        print the kernel name

    -n, --nodename
        print the network node hostname

    -r, --kernel-release
        print the kernel release

    -v, --kernel-version
        print the kernel version

    -m, --machine
        print the machine hardware name

Manual page uname(1) line 1 (press h for help or q to quit)
```

```
evgentoph@DESKTOP-TH92VI3: ~
grep(1)                                User Commands                                GREP(1)

NAME
    grep, egrep, fgrep, rgrep - print lines that match patterns

SYNOPSIS
    grep [OPTION...] PATTERNS [FILE...]
    grep [OPTION...] -e PATTERNS ... [FILE...]
    grep [OPTION...] -f PATTERN FILE ... [FILE...]

DESCRIPTION
    grep searches for PATTERNS in each FILE. PATTERNS is one or more patterns separated by newline characters,
    and grep prints each line that matches a pattern. Typically PATTERNS should be quoted when grep is used in a
    shell command.

    A FILE of "-" stands for standard input. If no FILE is given, recursive searches examine the working
    directory, and nonrecursive searches read standard input.

    In addition, the variant programs egrep, fgrep and rgrep are the same as grep -E, grep -F, and grep -r,
    respectively. These variants are deprecated, but are provided for backward compatibility.

OPTIONS
    Generic Program Information
        --help Output a usage message and exit.

        -V, --version
            Output the version number of grep and exit.

    Pattern Syntax

Manual page grep(1) line 1 (press h for help or q to quit)
```

Создайте рабочую директорию и перейдите в нее

```
evgentoph@DESKTOP-TH92VI3:~$ mkdir dir2
evgentoph@DESKTOP-TH92VI3:~$ cd dir2
evgentoph@DESKTOP-TH92VI3:~/dir2$ _
```

Определить версию

```
evgentoph@DESKTOP-TH92VI3:~/dir2$ uname --version
uname (GNU coreutils) 8.32
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Written by David MacKenzie.
```

Запись в файл

```
evgentoph@DESKTOP-TH92VI3:~/dir2$ touch file1
evgentoph@DESKTOP-TH92VI3:~/dir2$ uname --version >> file1
evgentoph@DESKTOP-TH92VI3:~/dir2$ cat file1
uname (GNU coreutils) 8.32
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.


Written by David MacKenzie.
evgentoph@DESKTOP-TH92VI3:~/dir2$
```

PATH и запись в тот же файл

```
evgentoph@DESKTOP-TH92VI3:~/dir2$ echo $PATH >> file1
evgentoph@DESKTOP-TH92VI3:~/dir2$ cat file1
uname (GNU coreutils) 8.32
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Written by David MacKenzie.
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/mnt/c/Program Files/WindowsApps/CanonicalGroupLimited.Ubuntu_2204.1.7.0_x64__79rhkp1fndgsc:/mnt/c/Program Files (x86)/Razer Chroma SDK/bin:/mnt/c/Program Files/Razer Chroma SDK/bin:/mnt/c/Program Files (x86)/Razer/ChromaBroadcast/bin:/mnt/c/Program Files/Razer/ChromaBroadcast/bin:/mnt/c/ProgramData/Oracle/Java/javapath:/mnt/c/Windows/system32:/mnt/c/Windows:/mnt/c/Windows/System32/Wbem:/mnt/c/Windows/System32/WindowsPowerShell/v1.0:/mnt/c/Windows/System32/OpenSSH:/mnt/c/Program Files (x86)/NVIDIA Corporation/PhysX/Common:/mnt/c/Program Files/NVIDIA Corporation/NVIDIA NvDLISR:/mnt/c/Windows/system32/config/systemprofile/AppData/Local/Microsoft/WindowsApps:/mnt/d/python/Scripts:/mnt/d/python:/mnt/c/Python1337/Scripts:/mnt/c/Python1337:/mnt/c/Users/step/AppData/Local/Microsoft/WindowsApps:/mnt/d/VSCode/Microsoft VS Code/bin:/mnt/d/pyCharm/PyCharm Community Edition 2022.2.2/bin:/snap/bin
evgentoph@DESKTOP-TH92VI3:~/dir2$
```

Определение архитектуры процессора

 evgentoph@DESKTOP-TH92VI3: ~/dir2

```
evgentoph@DESKTOP-TH92VI3:~/dir2$ uname > file1
evgentoph@DESKTOP-TH92VI3:~/dir2$ cat file1
Linux
evgentoph@DESKTOP-TH92VI3:~/dir2$ _
```

Изменения прав и полные сведения

```

evgentoph@DESKTOP-TH92VI3:~/dir2$ touch file2 file3
evgentoph@DESKTOP-TH92VI3:~/dir2$ ls
file1 file2 file3
evgentoph@DESKTOP-TH92VI3:~/dir2$ chmod u-rw file2
evgentoph@DESKTOP-TH92VI3:~/dir2$ cat file2
cat: file2: Permission denied
evgentoph@DESKTOP-TH92VI3:~/dir2$ chmod u+rw file2
evgentoph@DESKTOP-TH92VI3:~/dir2$ cat file2
evgentoph@DESKTOP-TH92VI3:~/dir2$ echo gg >> file2
evgentoph@DESKTOP-TH92VI3:~/dir2$ cat file2
gg
evgentoph@DESKTOP-TH92VI3:~/dir2$ ls -l
total 4
-rw-r--r-- 1 evgentoph evgentoph 6 Dec 17 02:10 file1
-rw-r--r-- 1 evgentoph evgentoph 3 Dec 17 02:15 file2
-rw-r--r-- 1 evgentoph evgentoph 0 Dec 17 02:12 file3
evgentoph@DESKTOP-TH92VI3:~/dir2$ chmod o-r file3
evgentoph@DESKTOP-TH92VI3:~/dir2$ ls -l
total 4
-rw-r--r-- 1 evgentoph evgentoph 6 Dec 17 02:10 file1
-rw-r--r-- 1 evgentoph evgentoph 3 Dec 17 02:15 file2
-rw-r----- 1 evgentoph evgentoph 0 Dec 17 02:12 file3
evgentoph@DESKTOP-TH92VI3:~/dir2$ █

```

Количество процессов

```

evgentoph@DESKTOP-TH92VI3:~/dir2$ ps
  PID TTY          TIME CMD
   11 tty1      00:00:00 bash
  327 tty1      00:00:00 bash
  377 tty1      00:00:00 ps
evgentoph@DESKTOP-TH92VI3:~/dir2$ █

```

Shell

```

evgentoph@DESKTOP-TH92VI3:~/dir2$ shell
Command 'shell' not found, did you mean:
  command 'rshell' from deb pyboard-rshell (0.0.31-0ubuntu1)
  command 'bshell' from deb avahi-ui-utils (0.8-5ubuntu5)
  command 'jshell' from deb openjdk-11-jdk-headless (11.0.17+8-1ubuntu2~22.04)
  command 'jshell' from deb openjdk-17-jdk-headless (17.0.5+8-2ubuntu1~22.04)
  command 'jshell' from deb openjdk-18-jdk-headless (18.0.2+9-2~22.04)
  command 'jshell' from deb openjdk-19-jdk-headless (19.0.1+10-1ubuntu1~22.04)
  command 'spell' from deb spell (1.0-24.2)
  command 'shelr' from deb shelr (0.16.3-2.1)
Try: sudo apt install <deb name>
evgentoph@DESKTOP-TH92VI3:~/dir2$ █

```

Nano


```
evgentoph@DESKTOP-TH92VI3: ~/dir2
GNU nano 6.2
Зима недаром элится,
Прошла ее пора-
Весна в окно стучится
И гонит со двора.

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo      M-A Set Mark
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo      M-6 Copy
```

Сведения

```
evgentoph@DESKTOP-TH92VI3: ~/dir2
evgentoph@DESKTOP-TH92VI3:~/dir2$ ls
file1 file2 file3 nano.398.save
evgentoph@DESKTOP-TH92VI3:~/dir2$ echo /etc/passwd
/etc/passwd
evgentoph@DESKTOP-TH92VI3:~/dir2$
```

Help

```
evgentoph@DESKTOP-TH92VI3: ~/dir2
break [n]
builtin [shell-builtin [arg ...]]
caller [expr]
case WORD in [PATTERN] [PATTERN]...) COMMANDS ;;... esac
cd [-L][-P [-e]] [-@] [dir]
command [-pVv] command [arg ...]
compgen [-abcdefgjkuv] [-o option] [-A action] [-G globp]
complete [-abcdefgjkuv] [-pr] [-DEI] [-o option] [-A act]
comptop [-o|+o option] [-DEI] [name ...]
continue [n]
coproc [NAME] command [redirections]
declare [-aAfFgiIlNrTux] [-p] [name[=value] ...]
dirs [-clpv] [+N] [-N]
disown [-h] [-ar] [jobspec ... | pid ...]
echo [-neE] [arg ...]
enable [-a] [-dnps] [-f filename] [name ...]
eval [arg ...]
exec [-cl] [-a name] [command [argument ...]] [redirections]
exit [n]
export [-fn] [name[=value] ...] or export -p
false
fc [-e ename] [-lnr] [first] [last] or fc -s [pat=rep] [c]
fg [job_spec]
for NAME [in WORDS ...] ; do COMMANDS; done
for (( exp1; exp2; exp3 )) ; do COMMANDS; done
function name { COMMANDS ; } or name () { COMMANDS ; }
getopts optstring name [arg ...]
hash [-lr] [-p pathname] [-dt] [name ...]
help [-dms] [pattern ...]
printf [-v var] format [arguments]
pushd [-n] [+N | -N | dir]
pwd [-LP]
read [-ers] [-a array] [-d delim] [-i text] [-n nchars]
readarray [-d delim] [-n count] [-O origin] [-s count]
readonly [-aAf] [name[=value] ...] or readonly -p
return [n]
select NAME [in WORDS ...] ; do COMMANDS; done
set [-abefhkmnptuvxBCHP] [-o option-name] [--] [arg ...]
shift [n]
shopt [-pqsu] [-o] [optname ...]
source filename [arguments]
suspend [-f]
test [expr]
time [-p] pipeline
times
trap [-lp] [[arg] signal_spec ...]
true
type [-afptP] name [name ...]
typeset [-aAfFgiIlNrTux] [-p] name[=value] ...
ulimit [-SHabcdefiklmnpqrstuvxPT] [limit]
umask [-p] [-S] [mode]
unalias [-a] name [name ...]
unset [-f] [-v] [-n] [name ...]
until COMMANDS; do COMMANDS; done
variables - Names and meanings of some shell variables
wait [-fn] [-p var] [id ...]
while COMMANDS; do COMMANDS; done
{ COMMANDS ; }
```

If,for,while

```

evgentoph@DESKTOP-TH92VI3: ~/dir2
evgentoph@DESKTOP-TH92VI3:~/dir2$ help if
if: if COMMANDS; then COMMANDS; [ elif COMMANDS; then COMMANDS; ]... [ else COMMANDS; ] fi
Execute commands based on conditional.

The `if COMMANDS' list is executed. If its exit status is zero, then the
`then COMMANDS' list is executed. Otherwise, each `elif COMMANDS' list is
executed in turn, and if its exit status is zero, the corresponding
`then COMMANDS' list is executed and the if command completes. Otherwise,
the `else COMMANDS' list is executed, if present. The exit status of the
entire construct is the exit status of the last command executed, or zero
if no condition tested true.

Exit Status:
Returns the status of the last command executed.
evgentoph@DESKTOP-TH92VI3:~/dir2$ help for
for: for NAME [in WORDS ... ] ; do COMMANDS; done
Execute commands for each member in a list.

The `for' loop executes a sequence of commands for each member in a
list of items. If `in WORDS ...;' is not present, then `in "$@"' is
assumed. For each element in WORDS, NAME is set to that element, and
the COMMANDS are executed.

Exit Status:
Returns the status of the last command executed.
evgentoph@DESKTOP-TH92VI3:~/dir2$ help while
while: while COMMANDS; do COMMANDS; done
Execute commands as long as a test succeeds.

Expand and execute COMMANDS as long as the final command in the
`while' COMMANDS has an exit status of zero.

Exit Status:
Returns the status of the last command executed.
evgentoph@DESKTOP-TH92VI3:~/dir2$

```

Присваивание нового значения переменой

```

evgentoph@DESKTOP-TH92VI3:~/dir2$ echo hello
hello
evgentoph@DESKTOP-TH92VI3:~/dir2$ hello="Hello world!"
evgentoph@DESKTOP-TH92VI3:~/dir2$ echo $hello
Hello world!
evgentoph@DESKTOP-TH92VI3:~/dir2$ _

```

Random

```

evgentoph@DESKTOP-TH92VI3:~/dir2$ RANDOM
RANDOM: command not found
evgentoph@DESKTOP-TH92VI3:~/dir2$ echo $RANDOM
24773
evgentoph@DESKTOP-TH92VI3:~/dir2$

```

```

evgentoph@DESKTOP-TH92VI3: ~/dir2
evgentoph@DESKTOP-TH92VI3:~/dir2$
evgentoph@DESKTOP-TH92VI3:~/dir2$ #!/bin/bash
evgentoph@DESKTOP-TH92VI3:~/dir2$ echo "Hello World"
Hello World
evgentoph@DESKTOP-TH92VI3:~/dir2$ _

```

Удаление директории

```
evgentoph@DESKTOP-TH92VI3:~/dir2$ cd
evgentoph@DESKTOP-TH92VI3:~$ rmdir dir2
rmdir: failed to remove 'dir2': Directory not empty
evgentoph@DESKTOP-TH92VI3:~$ cd dir2
evgentoph@DESKTOP-TH92VI3:~/dir2$ ls
file1
evgentoph@DESKTOP-TH92VI3:~/dir2$ rm file1
evgentoph@DESKTOP-TH92VI3:~/dir2$ cd
evgentoph@DESKTOP-TH92VI3:~$ rmdir dir2
evgentoph@DESKTOP-TH92VI3:~$ _
```