

Операционные системы

Лабораторная работа №5

Гульдяев Тихон Дмитриевич

Содержание

1	Цель работы	6
2	Выполнение лабораторной работы	7
3	Выводы	19
4	Ответы на контрольные вопросы	20
	Список литературы	27

Список таблиц

Список иллюстраций

2.1	домашний каталог	7
2.2	Переход в каталог /tmp	7
2.3	Вывод на экран содержимое каталога /tmp	8
2.4	Опция -l команды ls	8
2.5	Опция -a команды ls	8
2.6	Подкаталог cron	8
2.7	содержимое каталога /var/spool/cron/ и каталога /var/spool/	9
2.8	Владельцы файлов и подкаталогов в var/spool/ и var/spool/cron/	9
2.9	Создание newdir	9
2.10	Создание morefun	9
2.11	Создание letters, memos, misk	10
2.12	Удаление letters, memos, misk	10
2.13	Попытка удаления newdir	10
2.14	Удаление morefun	10
2.15	Флаг -R ls	10
2.16	Флаг -l команды ls	11
2.17	Флаг -t команды ls	11
2.18	Команда help cd	12
2.19	Команда man pwd	13
2.20	Команда man mkdir	14
2.21	Команда man rmdir	15
2.22	Команда man rm	16

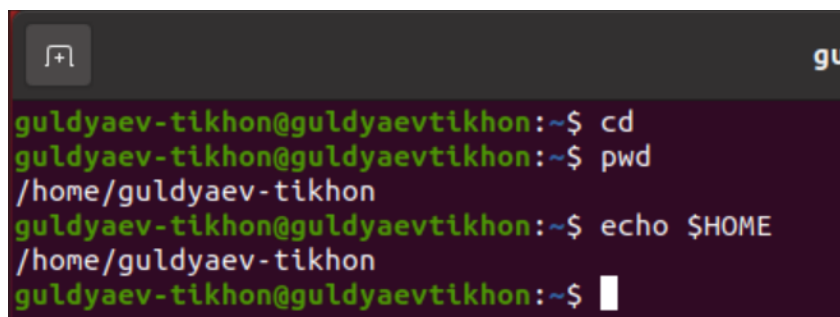
2.23 Модификация команды ls флагом -l	17
2.24 Модификация команды df флагом -h	18

1. Цель работы

Приобретение практических навыков взаимодействия пользователя с системой посредством командной строки.

2. Выполнение лабораторной работы

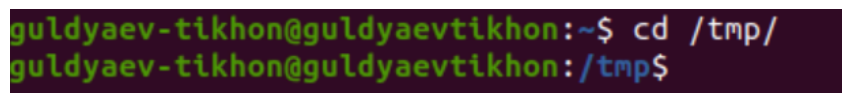
Определяю полное имя моего домашнего каталога. (рис. 2.1).

A terminal window with a dark background and green text. The prompt is 'guldyaev-tikhon@guldyaevtikhon:~\$'. The user enters 'cd', then 'pwd', which outputs '/home/guldyaev-tikhon'. Then the user enters 'echo \$HOME', which also outputs '/home/guldyaev-tikhon'. The prompt is now 'guldyaev-tikhon@guldyaevtikhon:~\$' followed by a cursor.

```
guldyaev-tikhon@guldyaevtikhon:~$ cd
guldyaev-tikhon@guldyaevtikhon:~$ pwd
/home/guldyaev-tikhon
guldyaev-tikhon@guldyaevtikhon:~$ echo $HOME
/home/guldyaev-tikhon
guldyaev-tikhon@guldyaevtikhon:~$
```

Рис. 2.1: домашний каталог

Перехожу в каталог /tmp. (рис. 2.2).

A terminal window with a dark background and green text. The prompt is 'guldyaev-tikhon@guldyaevtikhon:~\$'. The user enters 'cd /tmp/'. The prompt changes to 'guldyaev-tikhon@guldyaevtikhon:/tmp\$'.

```
guldyaev-tikhon@guldyaevtikhon:~$ cd /tmp/
guldyaev-tikhon@guldyaevtikhon:/tmp$
```

Рис. 2.2: Переход в каталог /tmp

Вывожу на экран содержимое каталога /tmp.(рис. 2.3).


```
gulyaev-tikhon@gulyaev-tikhon:/tmp$ cd /var/spool/cron/
gulyaev-tikhon@gulyaev-tikhon:~$ ls
crontabs
gulyaev-tikhon@gulyaev-tikhon:~$ ls /var/spool/
anacron  cron  cups  libreoffice  mail  rsyslog
gulyaev-tikhon@gulyaev-tikhon:~$
```

Рис. 2.7: содержимое каталога `/var/spool/cron/` и каталога `/var/spool/`

Определяю, кто является владельцем файлов и подкаталогов в `var/spool/` и `var/spool/cron/` (рис. 2.8).

```
guldyaev-tikhon@guldyaev-tikhon:~$ ls -l /var/spool/ | awk '{print $3}'
root
root
root
root
root
syslog
guldyaev-tikhon@guldyaev-tikhon:~$ ls -l /var/spool/cron/ | awk '{print $3}'
root
guldyaev-tikhon@guldyaev-tikhon:~$
```

Рис. 2.8: Владельцы файлов и подкаталогов в `var/spool/` и `var/spool/cron/`

Создаю каталог с именем newdir (рис. 2.9).

gulyaev-tikhongulyaev@tikhon: ~\$	mkdir newdir		
gulyaev-tikhongulyaev@tikhon: ~\$	ls		
academic-laboratory-report-template-master	file.txt	snap	Документы
Desktop	google-chrome-stable_current_and64.deb	study_2022-2023_os-intro	Загрузки
Downloads	labs.os	test_files	Избранное
004_linuxmonitoring_v1.0.0.tar.gz	newdir	work	Музыка
	Pictures	Видео	Общедоступные

Рис. 2.9: Создание newdir

В каталоге ~/newdir создаю новый каталог с именем morefun. (рис. 2.10).

```
guldyaev-tikhon@guldyaevtikhon:~$ mkdir newdir/morefun
guldyaev-tikhon@guldyaevtikhon:~$ ls newdir/
morefun
guldyaev-tikhon@guldyaevtikhon:~$
```

Рис. 2.10: Создание morefun

В домашнем каталоге создаю одной командой три новых каталога с именами letters, memos, misk.(рис. 2.11).

```
gulyaev-tikhon@gulyaev-tikhon:~$ mkdir memos letters misk
gulyaev-tikhon@gulyaev-tikhon:~$ ls
academic-laboratory-report-template-master  file.txt      misk      text_files  Изображения
Desktop                                     google-chrome-stable_current_amd64.deb  newdir     Pictures    work        Музыка
D04_LinuxMonitoring_v2.0-0                labs_05       letters    snap        study_2022-2023_os-Intro  Видео
D04_LinuxMonitoring_v2.0-0.tar.xz         memos        study_2022-2023_os-Intro  Загрузки   'Рабочий стол'  Общедоступные
Downloads                                  Pictures      work        Изображения  Музыка
gulyaev-tikhon@gulyaev-tikhon:~$
```

Рис. 2.11: Создание letters, memos, misk

Удаляю эти каталоги одной командой. (рис. 2.12).

```
gulyaev-tikhon@gulyaev-tikhon:~$ rm -rf memos/ misk/ letters/
gulyaev-tikhon@gulyaev-tikhon:~$ ls
academic-laboratory-report-template-master  file.txt      snap      Документы  'Рабочий стол'
Desktop                                     google-chrome-stable_current_amd64.deb  study_2022-2023_os-Intro  Загрузки   Изображения  Музыка
D04_LinuxMonitoring_v2.0-0                labs_05       text_files  Изображения  Музыка
D04_LinuxMonitoring_v2.0-0.tar.xz         newdir        work        Изображения  Музыка
Downloads                                  Pictures      work        Изображения  Музыка
gulyaev-tikhon@gulyaev-tikhon:~$
```

Рис. 2.12: Удаление letters, memos, misk

Пробую удалить newdir командой rm и проверяю это.(рис. 2.13). Безуспешно, ибо без дополнительных флагов команда rm для удаления директорий не подходит

```
gulyaev-tikhon@gulyaev-tikhon:~$ rm newdir/
rm: невозможно удалить 'newdir/': Это каталог
gulyaev-tikhon@gulyaev-tikhon:~$
```

Рис. 2.13: Попытка удаления newdir

Удаляю эти каталог morefun командой rmdir и проверяю это. (рис. 2.14).

```
gulyaev-tikhon@gulyaev-tikhon:~$ rmdir newdir/morefun/
gulyaev-tikhon@gulyaev-tikhon:~$ ls newdir/
gulyaev-tikhon@gulyaev-tikhon:~$
```

Рис. 2.14: Удаление morefun

С помощью команды man определяю, какую опцию команды ls нужно использовать для просмотра содержимое не только указанного каталога, но и подкаталогов, входящих в него. (рис. 2.15).

```
-R, --recursive
    list subdirectories recursively
```

Рис. 2.15: Флаг -R ls

С помощью команды `man` определяю набор опций команды `ls`, позволяющий отсортировать по времени последнего изменения выводимый список содержимого каталога с развёрнутым описанием файлов. (рис. 2.16) и (рис. 2.17) Это флаги `-l` и `-t`, пример команды: `ls -tl`

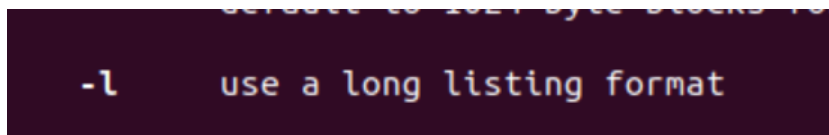


Рис. 2.16: Флаг `-l` команды `ls`

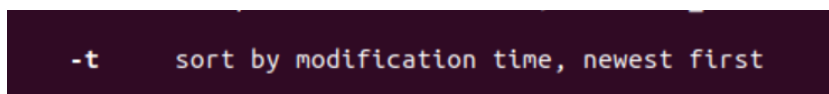


Рис. 2.17: Флаг `-t` команды `ls`

Используя команду `man` для просмотра описания следующих команд: `cd`, `pwd`, `mkdir`, `rmdir`, `rm`.

Команда `cd` не имеет своей `man`-страницы, поскольку она является встроенной командой оболочки (`shell built-in`), а не отдельным исполняемым файлом. Вместо этого можно прочитать информацию о команде `cd` с помощью команды `help cd` в терминале. (рис. 2.18).

Команда `cd` используется для смены текущего рабочего каталога в терминале.

- `-L`: По умолчанию команда `cd` следует символическим ссылкам. Опция `-L` указывает на то, что нужно следовать физическому пути (ссылки не разыменовываются).
- `-P`: Опция `-P` указывает на то, что нужно следовать логическому пути, то есть разыменовывать символические ссылки и следовать по физическому пути.
- `-e`: Опция `-e` позволяет использовать команду `cd` с переменными окружения. В этом случае переменная `HOME` используется как каталог назначения, если заданный каталог не существует.
- `-@`: Опция `-@` используется для показа атрибутов расширенных атрибутов файла или директории в терминале macOS. В Linux эта опция может отображать количество жестких ссылок на файл или директорию.

```

guldyaev-tikhon@guldyaevtikhon:~$ help cd
cd: cd [-L|[-P [-e]] [-@]] [karalor]
    Change the shell working directory.

    Change the current directory to DIR.  The default DIR is the value of the
    HOME shell variable.

    The variable CDPATH defines the search path for the directory containing
    DIR.  Alternative directory names in CDPATH are separated by a colon (:).
    A null directory name is the same as the current directory.  If DIR begins
    with a slash (/), then CDPATH is not used.

    If the directory is not found, and the shell option 'cdable_vars' is set,
    the word is assumed to be a variable name.  If that variable has a value,
    its value is used for DIR.

    Options:
    -L      force symbolic links to be followed: resolve symbolic
            links in DIR after processing instances of '..'
    -P      use the physical directory structure without following
            symbolic links: resolve symbolic links in DIR before
            processing instances of '..'
    -e      if the -P option is supplied, and the current working
            directory cannot be determined successfully, exit with
            a non-zero status
    -@      on systems that support it, present a file with extended
            attributes as a directory containing the file attributes

    The default is to follow symbolic links, as if '-L' were specified.
    '..' is processed by removing the immediately previous pathname component
    back to a slash or the beginning of DIR.

    Exit Status:
    Returns 0 if the directory is changed, and if $PWD is set successfully when
    -P is used; non-zero otherwise.
guldyaev-tikhon@guldyaevtikhon:~$

```

Рис. 2.18: Команда help cd

Команда man pwd.(рис. 2.19).

Команда pwd используется для вывода текущего рабочего каталога в терминале.

- -L – показывает символические ссылки в пути, которые могут привести к другому каталогу, тогда как опция -P использует фактический путь до каталога, следуя по символическим ссылкам;
- -P – выдает физический путь до текущего рабочего каталога (без символических ссылок);
- -h – аналогичен опции -L;
- -help – выдает краткую справку о команде pwd;
- -version – выдает информацию о версии команды pwd.

```
PWD(1) User Commands
NAME
  pwd - print name of current/working directory
SYNOPSIS
  pwd [OPTION]...
DESCRIPTION
  Print the full filename of the current working directory.

  -L, --logical
        use PWD from environment, even if it contains symlinks

  -P, --physical
        avoid all symlinks

  --help display this help and exit

  --version
        output version information and exit

  If no option is specified, -P is assumed.

  NOTE: your shell may have its own version of pwd, which usually supersedes the
  one in this manual. For details about the options it supports.
```

Рис. 2.19: Команда man pwd

Команда man mkdir. (рис. 2.20).

Команда mkdir используется для создания новых директорий (папок) в файловой системе.

Основные опции команды mkdir:

- -p, --parents: позволяет создавать вложенные директории. Если указан путь, который содержит несколько директорий, а некоторые из этих директорий еще не существуют, то эта опция создаст все необходимые директории.
- -m, --mode: позволяет задать права доступа к создаваемым директориям. Права доступа задаются в восьмеричной системе счисления. Например, значение 777 означает, что у всех пользователей есть полные права на чтение, запись и исполнение файлов в данной директории.
- -v, --verbose: позволяет выводить информацию о созданных директориях.
- -Z, --context: позволяет задать контекст безопасности SELinux для создаваемых директорий.
- --help: выводит справочную информацию о команде.
- --version: выводит информацию о версии команды.

```
MKDIR(1) User Commands
NAME
    mkdir - make directories
SYNOPSIS
    mkdir [OPTION]... DIRECTORY...
DESCRIPTION
    Create the DIRECTORY(ies), if they do not already exist.

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

    -m, --mode=MODE
        set file mode (as in chmod), not a=rwx - umask
    -p, --parents
        no error if existing, make parent directories as needed
    -v, --verbose
        print a message for each created directory
    -Z
        set SELinux security context of each created directory to the default type
    --context[=CTX]
        like -Z, or if CTX is specified then set the SELinux or SMACK security context to CTX
    --help
        display this help and exit
    --version
        output version information and exit
AUTHOR
```

Рис. 2.20: Команда man mkdir

Команда man rmdir. (рис. 2.21). Команда rmdir используется для удаления пустых каталогов. Она не может удалить каталоги, содержащие файлы или другие каталоги.

Основные опции команды rmdir:

- `-ignore-fail-on-non-empty`: игнорирует ошибки при попытке удалить непустой каталог.
- `-p, --parents`: позволяет удалить указанный каталог и все его родительские каталоги, если они пустые.
- `-v, --verbose`: выводит информацию о каждом удаляемом каталоге.
- `--help`: выводит справочную информацию о команде.
- `--version`: выводит информацию о версии команды.

```
RMDIR(1) User Commands
NAME
  rmdir - remove empty directories
SYNOPSIS
  rmdir [OPTION]... DIRECTORY...
DESCRIPTION
  Remove the DIRECTORY(ies), if they are empty.
  --ignore-fail-on-non-empty
      ignore each failure that is solely because a directory
      is non-empty
  -p, --parents
      remove DIRECTORY and its ancestors; e.g., 'rmdir -p a/b/c' is similar to 'rmdir a/b/c a/b a'
  -v, --verbose
      output a diagnostic for every directory processed
  --help
      display this help and exit
  --version
      output version information and exit
```

Рис. 2.21: Команда man rmdir

Команда man rm. (рис. 2.22).

Команда rm в Ubuntu используется для удаления файлов или директорий. Основные опции команды rm:

- -f, --force: Игнорировать несуществующие файлы и не запрашивать подтверждения удаления.
- -i: Запрашивать подтверждение перед удалением каждого файла.
- -I: Запросить подтверждение, если больше двух файлов удаляются или каталог удаляется рекурсивно.
- --interactive: Запрашивать подтверждение перед удалением каждого файла, аналогично -i.
- --one-file-system: Не удалить файлы из файловых систем, отличных от той, на которой находится указанный файл.
- --no-preserve-root: Не предотвращать удаление "/" (корневого) каталога.
- --preserve-root: Предотвращает удаление "/" (корневого) каталога.
- -r, -R, --recursive: Удалять каталоги и их содержимое рекурсивно.
- -d, --dir: Удалить только каталоги, даже если они не пусты.
- -v, --verbose: Подробно отображать информацию о процессе удаления.
- --help: Отображает справочную информацию о команде rm.
- --version: Отображает информацию о версии команды rm.

```
RM(1)                                User Commands                                RM(1)
NAME
rm - remove files or directories
SYNOPSIS
rm [OPTION]... [FILE]...
DESCRIPTION
This manual page documents the GNU version of rm.  rm removes each specified file.  By default, it does not remove directories.
If the -I or --interactive=once option is given, and there are more than three files or the -f, -R, or --recursive are given, then rm prompts
the user for whether to proceed with the entire operation.  If the response is not affirmative, the entire command is aborted.
Otherwise, if a file is unwritable, standard input is a terminal, and the -f or --force option is not given, or the -i or --interactive=always
option is given, rm prompts the user for whether to remove the file.  If the response is not affirmative, the file is skipped.
OPTIONS
Remove (unlink) the FILE(s).
-f, --force
    ignore nonexistent files and arguments, never prompt
-i
    prompt before every removal
-I
    prompt once before removing more than three files, or when removing recursively; less intrusive than -i, while still giving protection
    against most mistakes
--interactive=[WHEN]
    prompt according to WHEN: never, once (-I), or always (-i); without WHEN, prompt always
--one-file-system
    when removing a hierarchy recursively, skip any directory that is on a file system different from that of the corresponding command line
    argument
--no-preserve-root
    do not treat '/' specially
--preserve-root[=all]
    do not remove '/' (default); with 'all', reject any command line argument on a separate device from its parent
-r, -R, --recursive
    remove directories and their contents recursively
-d, --dir
    remove empty directories
-v, --verbose
    explain what is being done
--help
    display this help and exit
--version
    output version information and exit
```

Рис. 2.22: Команда man rm

Используя информацию, полученную при помощи команды history, выполняю модификацию(флаг -l) и исполнение команды ls. (рис. 2.23).


```

511 ls
512 clera
513 clear
514 hi
515 clear
516 history
guldyaeв-tikhon@guldyaeвтikhon:~$ !511 -l
ls -l
total 125436
drwxrwxr-x 4 guldyaeв-tikhon guldyaeв-tikhon 4096 фев 7 09:16 academic-la
poratory-report-template-master
drwxr-xr-x 3 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 10 20:37 Desktop
drwxrwxr-x 9 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 19 09:50 D04_LinuxMo
toring_v2.0-0
-rw-rw-r-- 1 guldyaeв-tikhon guldyaeв-tikhon 34423272 мар 19 11:26 D04_LinuxMo
toring_v2.0-0.tar.xz
drwxr-xr-x 3 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 10 22:09 Downloads
-rwxr--r-- 1 guldyaeв-tikhon guldyaeв-tikhon 0 мар 3 22:06 file.txt
-rw-rw-r-- 1 guldyaeв-tikhon guldyaeв-tikhon 93937560 мар 4 12:06 google-chro
ne-stable_current_amd64.deb
drwxrwxr-x 3 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 10 20:20 labs_05
drwxrwxr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 31 03:26 newdir
drwxr-xr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 3 21:06 Pictures
drwx----- 4 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 10 20:48 snap
drwxrwxr-x 7 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 29 20:44 study_2022-
2023_os-intro
drwxrwxr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 3 21:24 text_files
drwxrwxr-x 3 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 10 20:47 work
drwxr-xr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 24 12:39 Видео
drwxr-xr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 24 12:39 Документы
drwxr-xr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 24 12:39 Загрузки
drwxr-xr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 31 02:32 Изображения
drwxr-xr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 24 12:39 Музыка
drwxr-xr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 24 12:39 Общедоступн
e
drwxr-xr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 24 17:54 'Рабочий сто
p'
drwxr-xr-x 2 guldyaeв-tikhon guldyaeв-tikhon 4096 мар 24 12:39 Шаблоны
guldyaeв-tikhon@guldyaeвтikhon:~$

```

Рис. 2.23: Модификация команды ls флагом -l

Используя информацию, полученную при помощи команды history, выполняю модификацию(флаг -h) и исполнение команды df. (рис. 2.24).

```

508 df
509 df -h
510 clear
511 ls
512 clera
513 clear
514 hl
515 clear
516 history
517 ls -l
518 clear
519 df
520 clear
521 history
522 df -f
523 clear
524 history
guldyayev-tikhon@guldyayevtikhon:~$ !508 -h
df -h
Файл.система  Размер  Использовано  Дост  Использовано%  Смонтировано в
udev          1,9G      0             1,9G      0%             /dev
tmpfs         393M      1,4M          392M      1%             /run
/dev/sda5     39G       15G           22G       41%            /
tmpfs         2,0G      21M           1,9G      2%             /dev/shm
tmpfs         5,0M      4,0K          5,0M      1%             /run/lock
tmpfs         2,0G      0             2,0G      0%             /sys/fs/cgroup
/dev/loop0    56M       56M           0          100%            /snap/core18/2714
/dev/loop1    128K      128K          0          100%            /snap/bare/5
/dev/loop2    56M       56M           0          100%            /snap/core18/2721
/dev/loop3    64M       64M           0          100%            /snap/core20/1852
/dev/loop8    50M       50M           0          100%            /snap/snapd/18596
/dev/loop9    37M       37M           0          100%            /snap/gh/502
/dev/loop7    92M       92M           0          100%            /snap/gtk-common-themes/1535
/dev/loop4    347M      347M          0          100%            /snap/gnome-3-38-2004/115
/dev/loop6    46M       46M           0          100%            /snap/snap-store/638
/dev/loop5    64M       64M           0          100%            /snap/core20/1828
/dev/loop11   347M      347M          0          100%            /snap/gnome-3-38-2004/119
/dev/loop10   50M       50M           0          100%            /snap/snapd/18357
/dev/loop12   55M       55M           0          100%            /snap/snap-store/558
/dev/sda1     511M      4,0K          511M      1%             /boot/efi
tmpfs         393M      28K           393M      1%             /run/user/1000
/dev/sr0      51M       51M           0          100%            /media/guldyayev-tikhon/VBox_GAs_7.0.2
guldyayev-tikhon@guldyayevtikhon:~$

```

Рис. 2.24: Модификация команды df флагом -h

3. Выводы

Я приобрел практические навыки взаимодействия пользователя с системой посредством командной строки.

4. Ответы на контрольные вопросы

1. Что такое командная строка? Командная строка - это интерфейс для взаимодействия с операционной системой через ввод команд в текстовом формате. Она позволяет пользователю управлять компьютером и выполнять различные задачи, такие как создание, копирование, перемещение или удаление файлов и директорий, управление процессами, настройка системы и многое другое. Командная строка является более гибким и мощным способом работы с компьютером по сравнению с графическим интерфейсом пользователя (GUI), так как позволяет выполнять более сложные и настраиваемые задачи с помощью команд и опций.

2. При помощи какой команды можно определить абсолютный путь текущего каталога? Приведите пример.

Для определения абсолютного пути текущего каталога в командной строке Linux можно использовать команду “pwd” (print working directory).

Пример: если вы находитесь в домашней директории, то команда “pwd” выведет “/home/username”, где “username” - ваше имя пользователя.

3. При помощи какой команды и каких опций можно определить только тип файлов и их имена в текущем каталоге? Приведите примеры.

Для определения только типа файлов и их имен в текущем каталоге можно использовать команду ls с опцией -p. Эта опция добавляет к имени каждого элемента символ, указывающий на его тип: / для каталогов, * для исполняемых файлов, @ для символических ссылок и т.д.

Пример использования:

```
ls -p  
file.txt  folder/  program*
```

В этом примере вывод команды `ls -p` показывает имена трех элементов в текущем каталоге, а также символы, указывающие на их тип: `file.txt` - обычный файл, `folder/` - каталог, `program*` - исполняемый файл.

4. Какие файлы считаются скрытыми? Как получить информацию о скрытых файлах? Приведите примеры.

В UNIX-подобных системах скрытыми считаются файлы и каталоги, имена которых начинаются с точки (.). Такие файлы скрываются от обычного просмотра с помощью команды `ls` без дополнительных опций.

Чтобы получить информацию о скрытых файлах, можно использовать опцию `-a` (или `-all`), которая позволяет отобразить все файлы, включая скрытые. Например, команда `ls -a` отобразит все файлы и каталоги, включая скрытые.

Пример:

```
ls -a  
.    ..   .bashrc  Documents  Downloads  .ssh
```

5. При помощи каких команд можно удалить файл и каталог? Можно ли это сделать одной и той же командой? Приведите примеры. Для удаления файлов в Linux можно использовать команду `rm`, например:

```
rm file.txt
```

Для удаления каталогов с их содержимым можно использовать команду `rm` с опцией `-r` или `-R` (рекурсивное удаление), например:

```
rm -r dir/
```

Удаление файлов и каталогов можно также осуществлять при помощи команды `rmdir`, однако она удаляет только пустые каталоги. Например:

```
rmdir empty_dir/
```

Нельзя удалить каталог с помощью команды `rm`, если он не пустой. Также не рекомендуется использовать команду `rm` для удаления каталогов, так как она может удалить файлы, которые находятся внутри каталога, и не дает возможности восстановления.

Если нужно удалить каталог с его содержимым, то можно использовать команду `rm` с опцией `-r` или `-R`. Однако, для безопасности, лучше использовать команду `rm` с опцией `-i` или `-interactive`, которая запросит подтверждение удаления каждого файла или каталога. Например:

```
rm -ri dir/
```

6. Как определить, какие команды выполнил пользователь в сеансе работы? Чтобы определить, какие команды были выполнены пользователем в текущем сеансе работы в терминале, можно воспользоваться командой `history`. Она выводит список последних команд, выполненных пользователем, с указанием их номера. Кроме того, команда `history` позволяет повторить выполнение любой из ранее выполненных команд, указав ее номер.

7. Каким образом можно исправить и запустить на выполнение команду, которую пользователь уже использовал в сеансе работы? Приведите примеры.

В Linux есть несколько способов исправить и запустить на выполнение ранее использованную команду:

- Использование клавиши стрелки вверх: при нажатии на эту клавишу отображается последняя использованная команда, которую можно отредактировать и выполнить снова.

- Использование команды `history` и последующее использование команды `!:` с помощью этой команды можно выполнить конкретную команду из истории, указав ее номер после символа `!`. Например, команда `!10` выполнит команду с номером 10 из истории.
- Комбинация клавиш `Ctrl+R` позволяет использовать поиск по истории команд в обратном порядке. Это означает, что вы можете искать в истории команд за текущий сеанс работы, начиная с самых последних введенных команд и двигаясь назад во времени.

Чтобы использовать эту комбинацию клавиш, просто нажмите `Ctrl+R` в терминале, и он перейдет в интерактивный режим поиска. Наберите часть команды, которую вы ищете, и терминал найдет последнюю команду, содержащую эту строку. Если вы хотите найти предыдущее совпадение, нажмите `Ctrl+R` еще раз.

Кроме того, можно использовать стрелки вверх и вниз, чтобы просматривать найденные команды в истории, а затем нажать `Enter`, чтобы выполнить нужную команду. Если вы хотите редактировать найденную команду перед ее выполнением, используйте стрелки влево и вправо, чтобы перемещаться по строке, и `Backspace` для удаления символов. Пример использования `history` и команды `!:`

```
history 1 ls 2 cd Documents/ 3 cat file.txt 4 nano file.txt 5 rm file.txt 6 history $ !3 #
выполнение команды cat file.txt $ !4 # выполнение команды nano file.txt
```

8. Можно ли в одной строке записать несколько команд? Если да, то как? Приведите примеры. Да, в одной строке можно записать несколько команд, используя символ `;`, который разделяет команды. Каждая команда будет выполняться последовательно, независимо от результата выполнения предыдущей. Например:

```
ls -l; pwd; echo "Hello world!"
```

В этом примере сначала будет выполнена команда `ls -l`, затем `pwd` и, наконец, `echo "Hello world!"`. Каждая команда будет выполнена последовательно, вывод команд будет

отображаться в консоли по мере выполнения.

9. Что такое символ экранирования? Приведите примеры использования этого символа.

Символ экранирования - это специальный символ, который используется для указания того, что следующий за ним символ должен быть интерпретирован буквально, а не в качестве управляющей команды. В Unix-подобных системах символом экранирования является обратная косая черта.

Например, если мы хотим вывести на экран строку "Hello, World!", в которой есть кавычки, то мы можем использовать символ экранирования для указания, что кавычки являются частью строки:

```
echo "Hello, \"World\"!"
```

В результате выполнения этой команды на экран будет выведено:

```
Hello, "World"!
```

Также символ экранирования может использоваться для интерпретации специальных символов, таких как символы новой строки или табуляции. Например, чтобы вывести на экран строку с табуляцией, мы можем использовать следующую команду:

```
ls -l; pwd; echo "Hello world!"
```

Результат выполнения этой команды будет выглядеть так:

```
Name:   John
Age:    30
```

Здесь опция "-e" указывает на то, что специальные символы должны быть интерпретированы. Символ "\n" обозначает табуляцию, а символ "\n" - новую строку.

10. Какая информация выводится на экран о файлах и каталогах, если используется опция `l` в команде `ls`?

Опция `-l` команды `ls` в Linux выводит на экран детальную информацию о файлах и каталогах в текущем каталоге в виде списка. Для каждого элемента списка будет выведена следующая информация:

- Права доступа (permissions) - 10 символов, описывающих, какие операции разрешены для пользователя, группы и остальных пользователей;
- Количество ссылок на файл (links);
- Имя владельца файла (owner);
- Имя группы, которой принадлежит файл (group);
- Размер файла в байтах (size);
- Дата и время последней модификации файла (modification time);
- Имя файла или каталога (name).

11. Что такое относительный путь к файлу? Приведите примеры использования относительного и абсолютного пути при выполнении какой-либо команды.

Относительный путь к файлу - это путь к файлу относительно текущего рабочего каталога, а не относительно корневого каталога файловой системы. Например, если текущий рабочий каталог - это `/home/user`, а файл, который вы хотите открыть, находится в папке `documents`, то относительный путь к файлу будет `./documents/filename`.

В отличие от относительного пути, абсолютный путь начинается с корневого каталога. Например, `/home/user/documents/filename`.

Пример использования относительного пути в команде `cd`:

```
cd ./documents
```

Эта команда переместит текущий рабочий каталог в папку `documents`, которая находится в текущем каталоге.

Пример использования абсолютного пути в команде `cat`:

```
cat /home/user/documents/filename
```

Эта команда откроет файл с именем filename, который находится в папке documents, находящейся в домашней папке пользователя user.

12. Как получить информацию об интересующей вас команде?

Есть несколько способов получения информации о команде в терминале. Один из наиболее распространенных способов - использование команды man. man - это утилита командной строки, которая выводит справочную информацию о большинстве команд и утилит в Unix-подобных операционных системах. Чтобы получить информацию о команде, нужно ввести man и название команды.

Если команда не имеет руководства man, то можно попробовать использовать опцию -help. Эта опция выводит краткую информацию о том, как использовать команду и какие опции доступны.

Если опция -help не работает, то можно попробовать использовать -usage. Эта опция обычно выводит краткую информацию о том, как использовать команду.

Также можно использовать поиск в Интернете для получения информации о команде. Существуют многочисленные сайты и форумы, где можно найти ответы на вопросы об использовании команд.

13. Какая клавиша или комбинация клавиш служит для автоматического дополнения вводимых команд?

Для автоматического дополнения вводимых команд используется клавиша Tab. При наборе команды или пути к файлу можно нажать клавишу Tab, и система автоматически дополнит введенную строку до уникального значения. Если есть несколько возможных вариантов, то нажатие Tab дважды выведет список доступных вариантов. Эта функция также работает для дополнения аргументов команды. Например, можно начать вводить имя файла и затем нажать Tab, чтобы система автоматически дополнила имя файла.

Список литературы

<https://www.google.ru>

<https://chat.openai.com/chat>