

# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики

## ОТЧЕТ

### ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

*дисциплина:* Операционные системы

Студент группы НПИбд-01-21

Студенческий билет № 1032205621

Фамилия Имя Отчество Дессие Абди Бедаса

**МОСКВА**

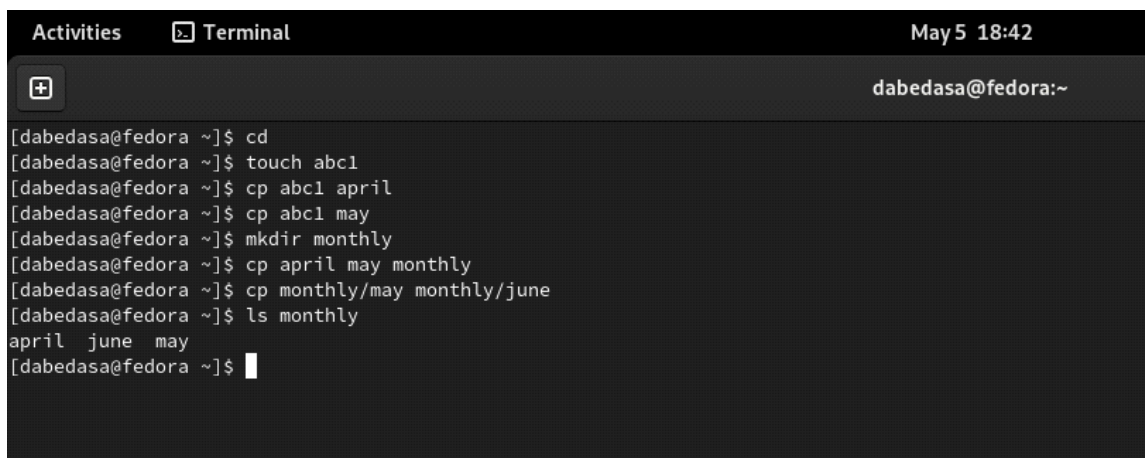
2022 г.

## Цель работы:

Ознакомиться с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобрести практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

## Ход работы:

- Выполнили все примеры, приведённые в первой части описания
- лабораторной работы.



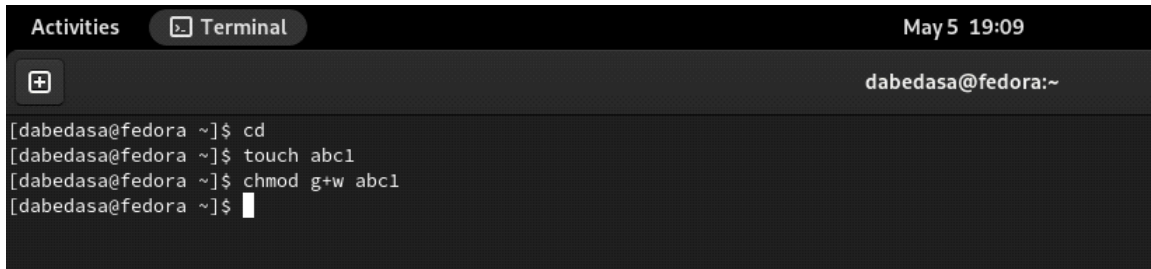
```
Activities  Terminal  May 5 18:42
dabedasa@fedora:~
[dabedasa@fedora ~]$ cd
[dabedasa@fedora ~]$ touch abc1
[dabedasa@fedora ~]$ cp abc1 april
[dabedasa@fedora ~]$ cp abc1 may
[dabedasa@fedora ~]$ mkdir monthly
[dabedasa@fedora ~]$ cp april may monthly
[dabedasa@fedora ~]$ cp monthly/may monthly/june
[dabedasa@fedora ~]$ ls monthly
april  june  may
[dabedasa@fedora ~]$
```

```
Activities Terminal May 5 18:46
dabedasa@fedora:~
[dabedasa@fedora ~]$ cd
[dabedasa@fedora ~]$ mkdir monthly.00
[dabedasa@fedora ~]$ cp -r monthly monthly.00
[dabedasa@fedora ~]$ cp -r monthly.00 /tmp
[dabedasa@fedora ~]$
```

```
Activities Terminal May 5 19:01
dabedasa@fedora:~
[dabedasa@fedora ~]$ cd
[dabedasa@fedora ~]$ mv april july
[dabedasa@fedora ~]$ mv july monthly.00
[dabedasa@fedora ~]$ ls monthly.00
july monthly
[dabedasa@fedora ~]$ ls monthly
april june may
[dabedasa@fedora ~]$ mv monthly.00 monthly.01
[dabedasa@fedora ~]$ mkdir reports
[dabedasa@fedora ~]$ mv monthly.01 reports
[dabedasa@fedora ~]$ mv reports/monthly.01 reports/monthly
[dabedasa@fedora ~]$
```

```
Activities Terminal May 5 19:04
dabedasa@fedora:~
[dabedasa@fedora ~]$ cd
[dabedasa@fedora ~]$ touch may
[dabedasa@fedora ~]$ ls -l may
-rw-rw-r--. 1 dabedasa dabedasa 0 May  5 19:02 may
[dabedasa@fedora ~]$ chmod u+x may
[dabedasa@fedora ~]$ ls -l may
-rwxrw-r--. 1 dabedasa dabedasa 0 May  5 19:02 may
[dabedasa@fedora ~]$ chmod u-x may
[dabedasa@fedora ~]$ ls -l may
-rw-rw-r--. 1 dabedasa dabedasa 0 May  5 19:02 may
[dabedasa@fedora ~]$
```

```
Activities Terminal May 5 19:08
dabedasa@fedora:~
[dabedasa@fedora ~]$ cd
[dabedasa@fedora ~]$ mkdir Monthly
[dabedasa@fedora ~]$ chmod g-r,o-r Monthly
[dabedasa@fedora ~]$
```

A terminal window titled 'Terminal' with a timestamp 'May 5 19:09'. The user is 'dabedasa@fedora' in the home directory '~'. The terminal shows the following commands and their outputs:

```
[dabedasa@fedora ~]$ cd
[dabedasa@fedora ~]$ touch abc1
[dabedasa@fedora ~]$ chmod g+w abc1
[dabedasa@fedora ~]$
```

- Выполнили следующие действия:
  - Скопировали файл `/usr/include/sys/io.h` в домашний каталог и назвали его `equipment`.
  - В домашнем каталоге создали директорию `~/ski.plases`.
  - Переместили файл `equipment` в каталог `~/ski.plases`.
  - Переименовали файл `~/ski.plases/equipment` в `~/ski.plases/equiplist`.
  - Создали в домашнем каталоге файл `abc1` и скопировали его в каталог `~/ski.plases`, назвали его `equiplist2`.
  - Создали каталог с именем `equipment` в каталоге `~/ski.plases`.
  - Переместили файлы `~/ski.plases/equiplist` и `equiplist2` в каталог `~/ski.plases/equipment`.
  - Создали и переместили каталог `~/newdir` в каталог `~/ski.plases` и назвали его `plans`.

```
Activities Terminal May 5 19:22
dabedasa@fedora:~/ski.places

[dabedasa@fedora ~]$ cd
[dabedasa@fedora ~]$ cp /usr/include/sys/io.h equipment
[dabedasa@fedora ~]$ ls
abc1      Documents  info      monthly  Pictures  Templates
ad.txt    Downloads  july.txt  Monthly  Public    Videos
april.txt equipment  junly     Music    refs
Desktop   hooks      may       objects  reports
[dabedasa@fedora ~]$ mkdir ski.places
[dabedasa@fedora ~]$ mv equipment ski.places
[dabedasa@fedora ~]$ mv ski.places/equipment ski.places/equiplist
[dabedasa@fedora ~]$ cp abc1 ski.places
[dabedasa@fedora ~]$ mv ski.places/abc1 ski.places/equiplist2
[dabedasa@fedora ~]$ cd ski.places
[dabedasa@fedora ski.places]$ mkdir equipment
[dabedasa@fedora ski.places]$ mv equiplist equipment
[dabedasa@fedora ski.places]$ mv equiplist2 equipment
[dabedasa@fedora ski.places]$ cd
[dabedasa@fedora ~]$ mkdir newdir
[dabedasa@fedora ~]$ mv newdir ski.places
[dabedasa@fedora ~]$ cd ski.places
[dabedasa@fedora ski.places]$ ls
equipment  newdir
[dabedasa@fedora ski.places]$ mv newdir plans
[dabedasa@fedora ski.places]$ ls
equipment  plans
[dabedasa@fedora ski.places]$
```

- Определили опции команды `chmod`, необходимые для того, чтобы присвоить файлам `australia`, `play`, `my_os` и `feathers` следующие права доступа соответственно, считая, что в начале таких прав нет: `drwxr--r--`, `drwx--x—x`, `-r-xr--r--`, `-rw-rw-r--`.

```
Activities Terminal May 5 19:37
dabedasa@fedora:~

[dabedasa@fedora ~]$ cd
[dabedasa@fedora ~]$ touch australia
[dabedasa@fedora ~]$ touch play
[dabedasa@fedora ~]$ touch my_os
[dabedasa@fedora ~]$ touch feathers
[dabedasa@fedora ~]$ chmod u+r,u+w,u+x australia
[dabedasa@fedora ~]$ chmod g+r,g-w,g-x australia
[dabedasa@fedora ~]$ chmod o+r,o-w,o-x australia
[dabedasa@fedora ~]$ chmod u+r,u+w,u+x play
[dabedasa@fedora ~]$ chmod g-r,g-w,g+x play
[dabedasa@fedora ~]$ chmod o-r,o-w,o+x play
[dabedasa@fedora ~]$ chmod u+r,u+w,u+x my_os
[dabedasa@fedora ~]$ chmod g+r,g-w,g-x my_os
[dabedasa@fedora ~]$ chmod o+r,o-w,o-x my_os
[dabedasa@fedora ~]$ chmod u+r,u+w,u+x feathers
[dabedasa@fedora ~]$ chmod o+r,o-w,o-x feathers
[dabedasa@fedora ~]$ chmod g+r,g-w,g-x feathers
[dabedasa@fedora ~]$
```

- Проделили следующие упражнения:

- Просмотрели содержимое файла /etc/passwd.

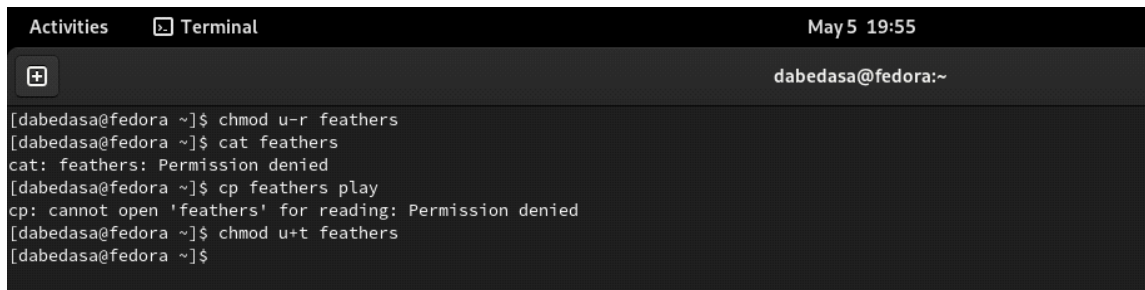
```
Activities Terminal May 5 19:40
dabedasa@fedora:~
[dabedasa@fedora ~]$ cd
[dabedasa@fedora ~]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
dbus:x:81:81:System message bus:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/usr/sbin/nologin
systemd-oom:x:999:999:systemd Userspace OOM Killer:/usr/sbin/nologin
systemd-resolve:x:193:193:systemd Resolver:/usr/sbin/nologin
systemd-timesync:x:998:998:systemd Time Synchronization:/usr/sbin/nologin
systemd-coredump:x:997:997:systemd Core Dumper:/usr/sbin/nologin
tss:x:59:59:Account used for TPM access:/dev/null:/sbin/nologin
qemu:x:107:107:qemu user:/sbin/nologin
polkitd:x:996:996:User for polkitd:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
unbound:x:995:994:Unbound DNS resolver:/etc/unbound:/sbin/nologin
dnsmasq:x:994:993:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/sbin/nologin
nm-openconnect:x:993:991:NetworkManager user for OpenConnect:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/sbin/nologin
gluster:x:992:990:GlusterFS daemons:/run/gluster:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
```

- Скопировали файл ~/feathers в файл ~/file.old.
- Переместили файл ~/file.old в каталог ~/play.
- Скопировали каталог ~/play в каталог ~/fun.
- Переместили каталог ~/fun в каталог ~/play и назвали его games.

```
Activities Terminal May 5 19:53
dabedasa@fedora:~
[dabedasa@fedora ~]$ cp feathers file.old
[dabedasa@fedora ~]$ mv file.old play
[dabedasa@fedora ~]$ cp -r play fun
[dabedasa@fedora ~]$ ls fun
fun
[dabedasa@fedora ~]$ mv fun play
[dabedasa@fedora ~]$ mv play games
[dabedasa@fedora ~]$ mv games play
[dabedasa@fedora ~]$ mv play/fun play/games
mv: cannot stat 'play/fun': No such file or directory
l[dabedasa@fedora ~]$ ls play
play
[dabedasa@fedora ~]$
```

- Лишили владельца файла ~/feathers права на чтение.

- Попытались просмотреть файл ~/feathers командой cat. Из-за лишения права на чтение, сделать этого не получилось.
- Попытались скопировать файл ~/feathers. Из-за лишения права на чтение, сделать этого не получилось.
- Дали владельцу файла ~/feathers право на чтение.

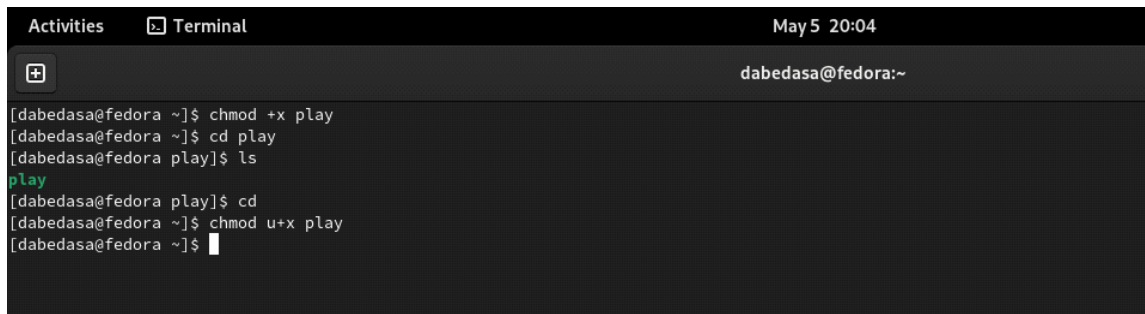


```

Activities Terminal May 5 19:55
dabedasa@fedora:~
[dabedasa@fedora ~]$ chmod u-r feathers
[dabedasa@fedora ~]$ cat feathers
cat: feathers: Permission denied
[dabedasa@fedora ~]$ cp feathers play
cp: cannot open 'feathers' for reading: Permission denied
[dabedasa@fedora ~]$ chmod u+t feathers
[dabedasa@fedora ~]$

```

- Лишили владельца каталога ~/play права на выполнение.
- Перешли в каталог ~/play. Ничего не изменилось, так как мы не запускали каталог, а перешли в него, однако сама иконка каталога изменилась.
- Дали владельцу каталога ~/play право на выполнение.



```

Activities Terminal May 5 20:04
dabedasa@fedora:~
[dabedasa@fedora ~]$ chmod +x play
[dabedasa@fedora ~]$ cd play
[dabedasa@fedora play]$ ls
play
[dabedasa@fedora play]$ cd
[dabedasa@fedora ~]$ chmod u+x play
[dabedasa@fedora ~]$

```

- Прочитали man по командам mount, fsck, mkfs, kill.

```
Activities  Terminal  May 5 20:08
dabedasa@fedora:~
[dabedasa@fedora ~]$ cd
[dabedasa@fedora ~]$ man mount
[dabedasa@fedora ~]$ man fsck
[dabedasa@fedora ~]$ man mkfs
[dabedasa@fedora ~]$ man kill
[dabedasa@fedora ~]$
```

- *mount* - нужна для просмотра смонтированных файловых систем, а также для монтирования любых локальных или удаленных файловых систем. Например, при вызове команды «`mount /dev/cdrom /mnt/cdrom`» устройство `/dev/cdrom` монтируется в каталог `/mnt/cdrom`, если он существует. Начиная от момента монтирования и пока пользователь не отмонтирует файловую систему (или туда не будет смонтировано что-то иное) в каталоге `/mnt/cdrom` будет содержаться дерево каталогов устройства `/dev/cdrom`; те файлы, и подкаталоги, которые раньше находились в `/mnt/cdrom`, сохранятся, но будут недоступны до размонтирования устройства `/dev/cdrom`. Для размонтирования достаточно указать точку монтирования или имя устройства, команда «`umount /dev/cdrom`». При запуске команды `mount` без параметров выводится список смонтированных файловых систем.



```
Activities Terminal May 5 20:05
dabedasa@fedora:~ — man mount
MOUNT(8) System Administration MOUNT(8)
NAME
mount - mount a filesystem
SYNOPSIS
mount [-h|-V]

mount [-l] [-t fstype]

mount -a [-fFnrsvw] [-t fstype] [-O optlist]

mount [-fnrsvw] [-o options] device|mountpoint

mount [-fnrsvw] [-t fstype] [-o options] device mountpoint

mount --bind|--rbind|--move olddir newdir

mount
--make-[shared|slave|private|unbindable|rshared|rslave|rprivate|runbindable]
mountpoint
DESCRIPTION
All files accessible in a Unix system are arranged in one big tree,
the file hierarchy, rooted at /. These files can be spread out over
several devices. The mount command serves to attach the filesystem
found on some device to the big file tree. Conversely, the
umount(8) command will detach it again. The filesystem is used to
control how data is stored on the device or provided in a virtual
way by network or other services.
Manual page mount(8) line 1 (press h for help or q to quit)
```

- *fsck* - `fsck [ -sAVRTNP ] [ -C [ fd ] ] [ -t fstype ] [filesystem ... ] [--] [ fs-specific-options ]` - проверяет и устраняет ошибки в файловой системе. Например, `fsck -fy -t ext4 /dev/sda1`. Опция `-f` (force) используется для принудительного выполнения проверки. Опция `-y` (yes) позволяет программе автоматически отвечать "да" на все вопросы в ходе работы.

```
Activities Terminal May 5 20:07
dabedasa@fedora:~ — man fsck

FSCK(8) System Administration FSCK(8)

NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-lsAVRTMNP] [-r [fd]] [-C [fd]] [-t fstype] [filesystem...]
    [--] [fs-specific-options]

DESCRIPTION
    fsck is used to check and optionally repair one or more Linux
    filesystems. filesystem can be a device name (e.g., /dev/hdc1,
    /dev/sdb2), a mount point (e.g., /, /usr, /home), or an filesystem
    label or UUID specifier (e.g.,
    UUID=8868abf6-88c5-4a83-98b8-bfc24057f7bd or LABEL=root). Normally,
    the fsck program will try to handle filesystems on different
    physical disk drives in parallel to reduce the total amount of time
    needed to check all of them.

    If no filesystems are specified on the command line, and the -A
    option is not specified, fsck will default to checking filesystems
    in /etc/fstab serially. This is equivalent to the -As options.

    The exit status returned by fsck is the sum of the following
    conditions:

    0
        No errors

    1
        Filesystem errors corrected

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

- *mkfs* - действие заключается в создании указанной файловой системы на выбранном диске или разделе. Например, команда «mkfs-text2 /dev/hda1» создает файловую систему ext2 на разделе hda1.

```
dabedasa@fedora:~ — man mkfs

MKFS(8) System Administration MKFS(8)

NAME
    mkfs - build a Linux filesystem

SYNOPSIS
    mkfs [options] [-t type] [fs-options] device [size]

DESCRIPTION
    This mkfs frontend is deprecated in favour of filesystem specific
    mkfs.<type> utils.

    mkfs is used to build a Linux filesystem on a device, usually a
    hard disk partition. The device argument is either the device name
    (e.g., /dev/hda1, /dev/sdb2), or a regular file that shall contain
    the filesystem. The size argument is the number of blocks to be
    used for the filesystem.

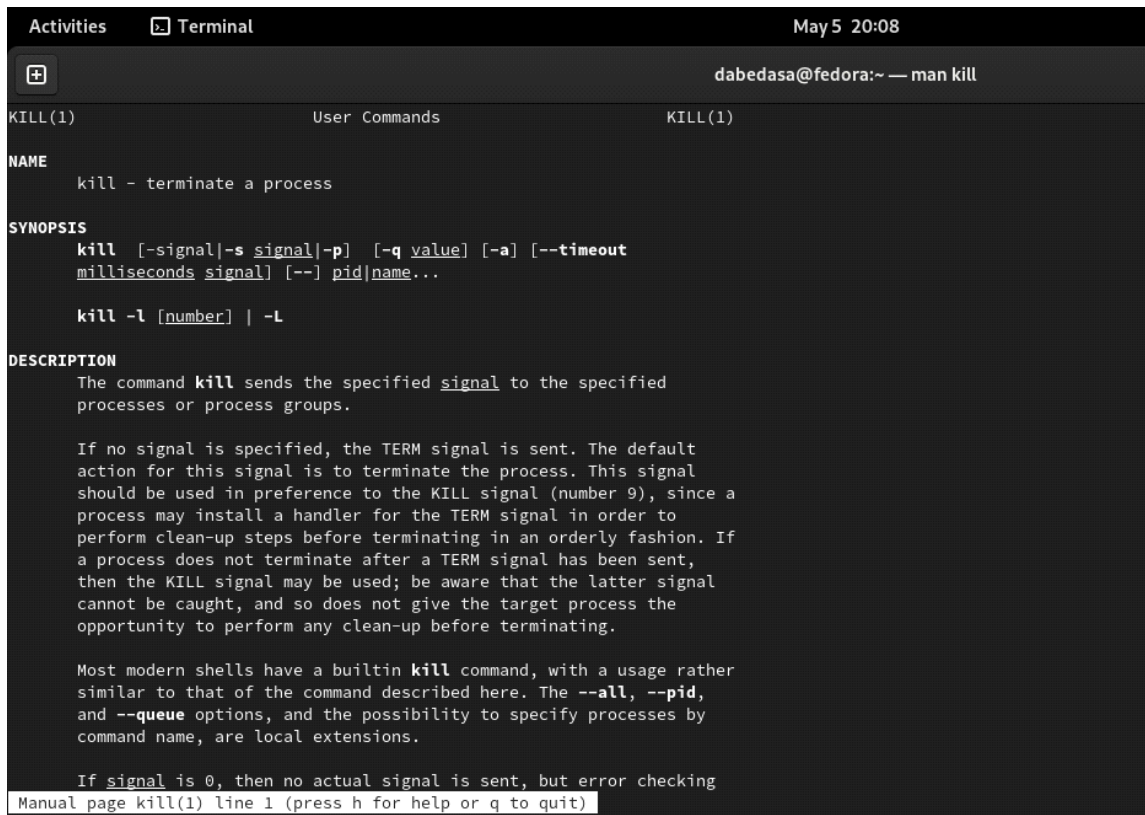
    The exit status returned by mkfs is 0 on success and 1 on failure.

    In actuality, mkfs is simply a front-end for the various filesystem
    builders (mkfs.fstype) available under Linux. The
    filesystem-specific builder is searched for via your PATH
    environment setting only. Please see the filesystem-specific
    builder manual pages for further details.

OPTIONS
    -t, --type type
        Specify the type of filesystem to be built. If not specified,
        the default filesystem type (currently ext2) is used.

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

- *kill* - `kill [ -s сигнал | -p ] [ -a ] pid - kill -l [ сигнал ]` - завершает некорректно работающее приложение. Например, чтобы послать сигнал SIGKILL (он имеет номер 9) процессу 2811, необходимо вызвать команду «`kill -9 2811`».



The screenshot shows a terminal window titled "Terminal" with the date and time "May 5 20:08". The user is logged in as "dabedasa@fedora:~" and has run the command "man kill". The terminal displays the manual page for the "kill" command, which includes the following sections:

```
KILL(1)                                User Commands                                KILL(1)

NAME
    kill - terminate a process

SYNOPSIS
    kill [-signal|-s signal|-p] [-q value] [-a] [--timeout
    milliseconds signal] [--] pid|name...

    kill -l [number] | -L

DESCRIPTION
    The command kill sends the specified signal to the specified
    processes or process groups.

    If no signal is specified, the TERM signal is sent. The default
    action for this signal is to terminate the process. This signal
    should be used in preference to the KILL signal (number 9), since a
    process may install a handler for the TERM signal in order to
    perform clean-up steps before terminating in an orderly fashion. If
    a process does not terminate after a TERM signal has been sent,
    then the KILL signal may be used; be aware that the latter signal
    cannot be caught, and so does not give the target process the
    opportunity to perform any clean-up before terminating.

    Most modern shells have a builtin kill command, with a usage rather
    similar to that of the command described here. The --all, --pid,
    and --queue options, and the possibility to specify processes by
    command name, are local extensions.

    If signal is 0, then no actual signal is sent, but error checking
    Manual page kill(1) line 1 (press h for help or q to quit)
```

## Вывод:

В ходе работы мы ознакомились с файловой системой Linux, а также приобрели практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами, по проверке использования диска и обслуживанию файловой системы.

## Ответы на контрольные вопросы:

- На компьютере в дисплейном классе есть следующие файловые системы:
- TmpFS — это быстрая и эффективная файловая система в ОЗУ, а не на ПЗУ, как прочие ФС. Предназначена для временного хранения файлов с оптимальным расходом памяти и скоростными характеристиками. Обычно используется при монтировании в каталог /tmp, в котором много постоянно меняющихся временных мелких файлов, поэтому может быть целесообразно размещать их в памяти.
- Ext4 - журналируемая файловая система, используемая в ОС на ядре Linux. Основана на файловой системе Ext3, но отличается тем, что в ней представлен механизм записи файлов в непрерывные участки блоков (екстенты), уменьшающий фрагментацию и повышающий производительность.
- Файловая система в дисплейном классе содержит следующие каталоги первого уровня:
  - /bin - Основные программы, необходимые для работы в системе: командные оболочки shell, основные утилиты.
  - /boot - Каталог, который содержит ядро системы— главную программу, загружающую и исполняющую все остальные.
  - /dev - Каталог, в котором содержатся псевдофайлы устройств. С точки зрения Linux все физические устройства, как главные, так и периферийные, представляют собой файлы особого типа, в которые система может записывать данные и из которых она может их считывать. Пользователь не должен работать с этими файлами, поскольку запись неправильных данных в файл устройства может повредить устройство или хранящиеся на нём данные.

- /etc - В этом каталоге содержатся системные конфигурационные файлы — текстовые файлы, которые считываются при загрузке системы и запуске программ и определяют их поведение. Настройка и администрирование Linux в конечном итоге сводится к редактированию этих файлов, даже если оно выполняется при помощи графических средств конфигурирования системы.
- /home - В структуре файловой системы Linux каждый пользователь имеет отдельный личный каталог для своих данных (т.н. домашний каталог), и все пользовательские каталоги выделены в отдельный общий каталог /home.
- /mnt - Каталоги для монтирования файловых систем сменных устройств и внешних файловых систем.
- /proc - Файловая система на виртуальном устройстве, её файлы содержат информацию о текущем состоянии системы.
- /root - Каталог администратора системы.
- /sbin - Системные утилиты.
- /usr - Программы и библиотеки, доступные пользователю.
- /var - Рабочие файлы программ, различные временные данные: очереди (письма на отправку, файлы на печать и др.), системные журналы (файлы, в которые записывается информация о происходящих в системе событиях).
- /tmp - Временные файлы.
- Для того чтобы содержимое некоторой файловой системы было доступно операционной системе, необходимо выполнить команду mount.
- Некорректность файловой системы может возникать не только в результате насильственного прерывания операций ввода-вывода, выполняемых непосредственно с диском, но и в результате нарушения

работы дискового кэша. Кэширование данных с диска предполагает, что в течение некоторого времени результаты операций ввода-вывода никак не сказываются на содержимом диска — все изменения происходят с копиями блоков диска, временно хранящихся в буферах оперативной памяти. В этих буферах оседают данные из пользовательских файлов и служебная информация файловой системы, такая как каталоги, индексные дескрипторы, списки свободных, занятых и поврежденных блоков и т. п.

- Разбитие диска на разделы и создание ФС в Linux делается при помощи специальных утилит — `cfdisk` `fdisk` `sfdisk` `mke2fs` `mkfs` `mkfs.ext2` `mkfs.ext3` `mkfs.ext4` `mkswap` `partimage` `parted` указывая им в качестве аргумента конкретное блочное устройство (`/dev/***`). Блочные устройства HDD вида `/dev/sda` можно использовать целиком для единственной ФС, но это редко применяется на практике. Лучше разделить все пространство на разделы меньшего размера и использовать их под разные задачи.
- Для просмотра небольших файлов удобно пользоваться командой `cat`. Формат команды: `cat имя-файла` Для просмотра больших файлов используйте команду `less` — она позволяет осуществлять постраничный просмотр файлов (длина страницы соответствует размеру экрана). Формат команды: `less имя-файла` Для управления процессом просмотра вы можете использовать следующие управляющие клавиши: Пробел → переход на следующую страницу, ENTER → сдвиг вперед на одну строку, b → возврат на предыдущую страницу, h → обращение за подсказкой, q → выход в режим командной строки. Для просмотра начала файла вы можете воспользоваться командой `head`. По умолчанию она выводит первые 10 строк файла.

- Копирование файлов и каталогов осуществляется при помощи команды `ср`. Формат команды: `ср[-опции] исходный_файл целевой_файл`. Опция `i` в команде `ср` поможет избежать уничтожения информации в случае, если на место целевого файла вы поставите имя уже существующего файла: система попросит подтвердить, что вы хотите перезаписать этот файл. Команда `ср` с опцией `r` (`recursive`) позволяет копировать каталоги вместе с входящими в них файлами и каталогами
- Команды `mv` и `mkdir` предназначены для перемещения и переименования файлов и каталогов. Формат команды: `mv [-опции] старый_файл новый_файл`. Для получения предупреждения перед переписыванием файла стоит использовать опцию `i`.
- Права доступа определяют, кто и что может делать с содержимым файла. Существуют три группы прав доступа: для владельца файла, для членов группы, для всех остальных. Для изменения прав доступа к файлу или каталогу используется команда `chmod`. Права доступа к файлу может поменять только владелец или суперпользователь (администратор). Формат команды: `chmod режим имя_файла`. Режим (в формате команды) имеет следующую структуру и способ записи:
  - `=` установить право;
  - `-` лишить права;
  - `+` дать право;
  - `r` чтение;
  - `w` запись;
  - `x` выполнение;
  - `u` (`user`) владелец файла;
  - `g` (`group`) группа, к которой принадлежит владелец файла;
  - `(others)` все остальные