

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2

Дисциплина: Архитектура компьютеров и операционные системы

Студент: Пестова Е. К.

Группа: НКАбд-04-23

МОСКВА

2023г.

Содержание

1. Цель работы	3
2. Задание	4
3. Теоретическое введение	5
4. Выполнение лабораторной работы	6
5. Выводы	20
6. Ответы на контрольные вопросы для самопроверки.....	27
7. Источники.....	28

1. Цель работы

Цель данной работы- приобретение практических навыков работы с операционной системой на уровне командной строки: организация файловой системы, навигация по файловой системе, создание и удаление файлов и директорий.

2. Задание

1. Перемещение по файловой системе.
2. Создание пустых каталогов и файлов.
3. Перемещение и удаление файлов или каталогов.
4. Команда cat: вывод содержимого файлов.
5. Выполнение заданий для самостоятельной работы.

3. Теоретическое введение

Файловая система определяет способ организации, хранения и именования данных на носителях информации компьютерах представляет собой иерархическую структуру в виде вложенных друг в друга каталогов (директорий), содержащих все файлы. В ОС Linux каталог, который является "вершиной" файловой системы, называется корневым каталогом, обозначается символом « / » и содержит все остальные каталоги и файлы. В большинстве Linux-систем поддерживается стандарт иерархии файловой системы (Filesystem Hierarchy Standard, FHS), унифицирующий местонахождение файлов и каталогов. Это означает, что в корневом каталоге находятся только подкаталоги со стандартными именами и типами данных, которые могут попасть в тот или иной каталог. Так, в любой Linux-системе всегда есть каталоги /etc, /home, /usr, /bin и т.п.

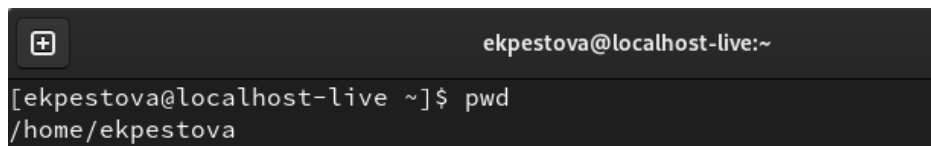
Обратиться к файлу, расположенному в каком-то каталоге, можно указав путь к нему. Полный или абсолютный путь начинается от корня (/), образуется перечислением всех каталогов, разделённых прямым слешем (/), и завершается именем файла, относительный путь - строится перечислением через (/) всех каталогов, но начинается от каталога, в котором "находится" пользователь. Таким образом, в Linux если имя объекта начинается с /, то системой это интерпретируется как полный путь. в любом другом случае - как относительный. В Linux любой пользователь имеет домашний каталог, который, как правило, имеет имя пользователя. В домашних каталогах хранятся документы и настройки пользователя. Для обозначения домашнего каталога используется знак тильды (~). При переходе из домашнего каталога знак тильды будет заменён на имя нового текущего каталога.

В операционной системе GNU Linux взаимодействие пользователя с системой обычно осуществляется с помощью командной строки посредством построения команд.

4. Выполнение лабораторной работы

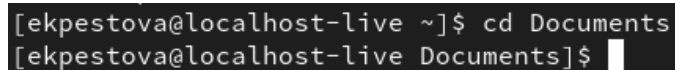
1. Перемещение по файловой системе

Открываю терминал, убеждаюсь, что нахожусь в домашней директории, потому что вижу значок тильда около имени пользователя. Это действительно так, поэтому сразу ввожу в терминале команду `pwd` и узнаю полный путь к домашнему каталогу:



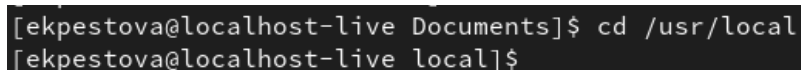
```
ekpestova@localhost-live:~  
[ekpestova@localhost-live ~]$ pwd  
/home/ekpestova
```

С помощью утилиты `cd` указываю относительный путь к каталогу `Documents` и перемещаюсь в указанную директорию, т. к. `Documents` – директория внутри домашнего каталога:



```
[ekpestova@localhost-live ~]$ cd Documents  
[ekpestova@localhost-live Documents]$
```

Перехожу в каталог `local`, который является подкаталогом директории `usr`, находящийся в корневом каталоге, для этого при написании команды указываю после утилиты `cd` абсолютный путь к нужному каталогу, начинающийся с корневого каталога `«/»`:



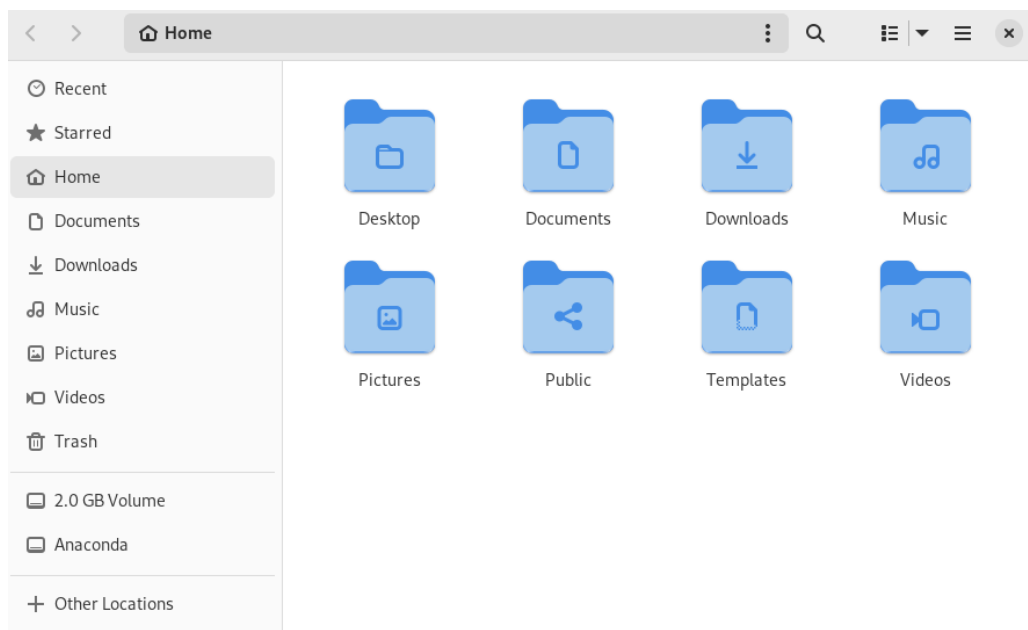
```
[ekpestova@localhost-live Documents]$ cd /usr/local  
[ekpestova@localhost-live local]$
```

Перехожу в последний каталог, в котором я была с помощью команды «cd —», потом перехожу на один каталог выше по иерархии с помощью команды «cd ..». Теперь я нахожусь в домашнем каталоге, потому что около имени пользователя есть значок тильда:

```
[ekpestova@localhost-live local]$ cd -  
/home/ekpestova/Documents  
[ekpestova@localhost-live Documents]$ cd ..  
[ekpestova@localhost-live ~]$
```

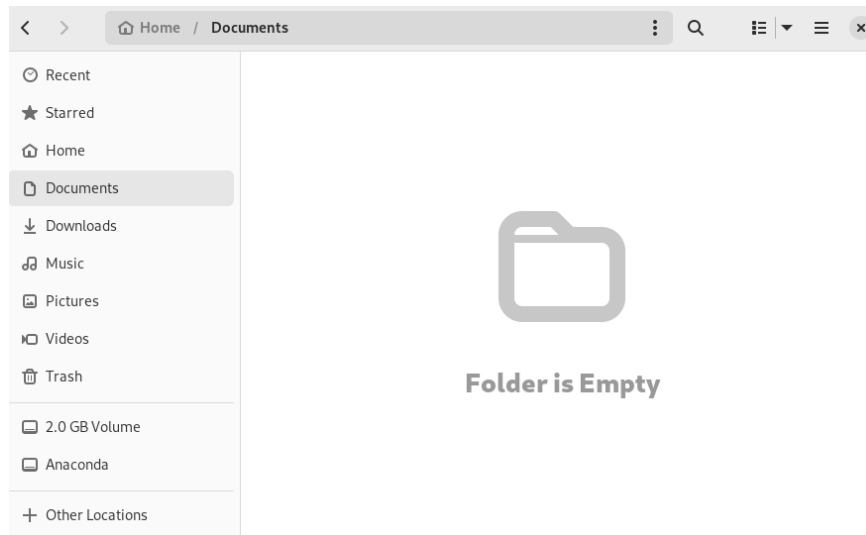
Вывожу директорию домашнего каталога с помощью утилиты ls, которая выдает список файлов текущего каталога. Открываю файловый менеджер графического окружения моей ОС. Выбираю домашнюю директорию пользователя в левой части окна файлового менеджера. Можем заметить, что вывод команды ls совпадает с файлами, отображающимися в графическом файловом менеджере, в домашней директории:

```
[ekpestova@localhost-live ~]$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos  
[ekpestova@localhost-live ~]$
```



Вывожу список файлов каталога Documents, указывая после утилиты ls относительный путь к каталогу, потому что Documents – подкаталог домашней директории. В каталоге Documents нет файлов, поэтому вывод пустой, это мы так же можем проверить через графический файловый менеджер, выбрав в домашнем каталоге директорию Documents:

```
[ekpestova@localhost-live ~]$ ls Documents
[ekpestova@localhost-live ~]$
```



Вывожу список файлов каталога /usr/local, указав абсолютный путь к нему после утилиты ls:

```
[ekpestova@localhost-live ~]$ ls /usr/local
bin  etc  games  include  lib  lib64  libexec  sbin  share  src
[ekpestova@localhost-live ~]$
```

Попробую вывести список файлов каталога /usr/local, используя ключи утилиты. Использую «-la», где -l – выводит дополнительные параметры файлов (права доступа, владельцы и группы, размеры файлов и время последнего доступа), -a – выводит все файлы каталога, включая скрытые файлы, в данном случае добавились директории «.» и «..» как скрытые. Также использую ключ -i, с помощью которого осуществляется вывод уникального номера файла в файловой системе перед каждым файлом:


```
[ekpestova@localhost-live ~]$ ls /usr/local -la
total 48
drwxr-xr-x. 12 root root 4096 Apr 13 17:44 .
drwxr-xr-x. 12 root root 4096 Apr 13 17:44 ..
drwxr-xr-x.  2 root root 4096 Jan 18  2023 bin
drwxr-xr-x.  2 root root 4096 Jan 18  2023 etc
drwxr-xr-x.  2 root root 4096 Jan 18  2023 games
drwxr-xr-x.  2 root root 4096 Jan 18  2023 include
drwxr-xr-x.  2 root root 4096 Jan 18  2023 lib
drwxr-xr-x.  3 root root 4096 Apr 13 17:44 lib64
drwxr-xr-x.  2 root root 4096 Jan 18  2023 libexec
drwxr-xr-x.  2 root root 4096 Jan 18  2023 sbin
drwxr-xr-x.  5 root root 4096 Apr 13 17:44 share
drwxr-xr-x.  2 root root 4096 Jan 18  2023 src
[ekpestova@localhost-live ~]$ ls /usr/local -i
439622 bin 439624 games 439626 lib 439629 libexec 439631 share
439623 etc 439625 include 439627 lib64 439630 sbin 439655 src
[ekpestova@localhost-live ~]$
```

2. Создание пустых каталогов и файлов

Создаю в домашнем каталоге подкаталог с именем parentdir с помощью утилиты mkdir, с помощью следующей команды ls проверяю правильность выполнения задания: да, директория parentdir находится в домашнем каталоге.

Создаю подкаталог dir в только что созданном каталоге parentdir.

Теперь перехожу в директорию parentdir, создаю в ней подкаталоги dir1, dir2, dir3, введя несколько аргументов для утилиты mkdir:

```
[ekpestova@localhost-live ~]$ mkdir parentdir
[ekpestova@localhost-live ~]$ ls
Desktop  Downloads  parentdir  Public  Videos
Documents  Music  Pictures  Templates
[ekpestova@localhost-live ~]$ mkdir parentdir/dir
[ekpestova@localhost-live ~]$ cd parentdir
[ekpestova@localhost-live parentdir]$ mkdir dir1 dir2 dir3
```

Создаю подкаталог в каталоге, отличном от текущего, для этого указываю путь к месту создания подкаталога: mkdir ~/newdir, т. е. сначала домашнюю директорию, в которой буду создавать подкаталог, потом название создаваемого подкаталога. Следующей командой «ls ~» проверяю, получилось ли создать подкаталог в домашнем каталоге:

```
[ekpestova@localhost-live parentdir]$ mkdir ~/newdir
[ekpestova@localhost-live parentdir]$ ls ~
Desktop    Downloads  newdir     Pictures   Templates
Documents  Music      parentdir  Public     Videos
```

Создаю иерархическую цепочку подкаталогов newdir/dir1/dir2, создавая все промежуточные каталоги, выбрав у утилиты mkdir опцию -p, позволяющую создавать последовательность вложенных каталогов:

```
[ekpestova@localhost-live parentdir]$ mkdir -p ~/newdir/dir1/dir2
[ekpestova@localhost-live parentdir]$
```

Создаю файл text.txt в каталоге ~/newdir/dir1/dir2, с помощью утилиты touch, прописывая путь к месту создания файла, в конце которого добавляю имя создаваемого файла ~/newdir/dir1/dir2/text.txt, также проверяю наличие файла с помощью команды ls ~/newdir/dir1/dir2, снова указывая путь от домашней директории:

```
[ekpestova@localhost-live parentdir]$ touch ~/newdir/dir1/dir2/text.txt
[ekpestova@localhost-live parentdir]$ ls ~/newdir/dir1/dir2
text.txt
[ekpestova@localhost-live parentdir]$
```

3. Перемещение и удаление файлов и каталогов

Для удаления пустых каталогов воспользуюсь командой rmdir. Запрашиваю подтверждение на удаление каждого файла в текущем каталоге с помощью ключа -i (в подтверждении отвечаю «Yes», чтобы удалить), удаляю в подкаталоге /newdir/dir1/dir2/ все файлы с именами, заканчивающимися на .txt, прописав в имени файла маску *, обозначающую любой символ или строку символов в имени файла. Удаляю из текущего каталога parentdir без запроса подтверждения на удаление каталог newdir с помощью ключа -R, также удаляю файлы, чьи имена начинаются с dir в каталоге parentdir, указывая ~/parentdir/dir* вторым аргументом для утилиты rm и добавляя маску * после dir:

```
[ekpestova@localhost-live parentdir]$ rm -R ~/newdir ~/parentdir/dir*
[ekpestova@localhost-live parentdir]$
```

С помощью `ls` и `ls ~` проверяю правильность выполнения команды:

```
[ekpestova@localhost-live parentdir]$ ls
[ekpestova@localhost-live parentdir]$ ls ~
Desktop    Downloads  parentdir  Public     Videos
Documents  Music      Pictures   Templates
```

Перемещаюсь в домашний каталог, создаю последовательности вложенных каталогов `parentdir1/dir1` `parentdir2/dir2` с помощью ключа `-p` утилиты `mkdir` и каталог `parentdir3`, передаю утилите три аргумента:

```
[ekpestova@localhost-live ~]$ mkdir -p parentdir1/dir1 parentdir2/dir2 parentdir3
```

Создаю файлы `text1.txt` и `text2.txt` в директории `parentdir1/dir1/` и `parentdir2/dir2/` с помощью утилиты `touch`. Сразу же делаю проверку на наличие созданных файлов в директории:

```
[ekpestova@localhost-live ~]$ touch parentdir1/dir1/text1.txt
[ekpestova@localhost-live ~]$ ls parentdir1/dir1
text1.txt
[ekpestova@localhost-live ~]$ touch parentdir2/dir2/text2.txt
[ekpestova@localhost-live ~]$ ls parentdir2/dir2
text2.txt
```

Использую команду `mv`, перемещаю файл `text1.txt`, указывая путь к нему, в директорию `parentdir3`. Использую команду `cp`, копирую файл `text2.txt` в каталог `parentdir3`, также указывая путь к файлу, который нужно скопировать:

```
[ekpestova@localhost-live ~]$ mv parentdir1/dir1/text1.txt parentdir3
[ekpestova@localhost-live ~]$ cp parentdir2/dir2/text2.txt parentdir3
[ekpestova@localhost-live ~]$
```

Проверяю, что в каталоге `parentdir3` действительно два файла, файла `text1.txt` теперь нет в каталоге `parentdir1/dir1`, `text2.txt` все еще находится в `parentdir2/dir2`:

```
[ekpestova@localhost-live ~]$ ls parentdir3
text1.txt  text2.txt
[ekpestova@localhost-live ~]$ ls parentdir1/dir1
[ekpestova@localhost-live ~]$ ls parentdir2/dir2
text2.txt
```

Еще раз посмотрим файлы в директории parentdir3 с помощью ls. Создаю копию text2.txt с новым именем subtest2.txt благодаря утилите cp. Переименовываю файл text1.txt из каталога parentdir3 в newtest.txt с помощью утилиты mv, а с помощью ее ключа -i запрашиваю подтверждение перед перезаписью. Проверяю правильность выполнения работы с помощью ls:

```
[ekpestova@localhost-live ~]$ ls parentdir3
text1.txt  text2.txt
[ekpestova@localhost-live ~]$ cp parentdir3/text2.txt parentdir3/subtest2.txt
[ekpestova@localhost-live ~]$ mv -i parentdir3/text1.txt parentdir3/newtest.txt
[ekpestova@localhost-live ~]$ ls parentdir3
newtest.txt  subtest2.txt  text2.txt
```

Перехожу в директорию parentdir1 с помощью утилиты cd и переименовываю каталог dir1 в каталоге parentdir1 в newdir с помощью mv. Я нахожусь в директории, где находится подкаталог dir1, поэтому прописывать путь до подкаталога мне не нужно:

```
[ekpestova@localhost-live ~]$ cd parentdir1
[ekpestova@localhost-live parentdir1]$ ls
dir1
[ekpestova@localhost-live parentdir1]$ mv dir1 newdir
[ekpestova@localhost-live parentdir1]$ ls
newdir
```

4. Команда cat: вывод содержимого файлов

Возвращаюсь в домашнюю директорию с помощью утилиты cd. Команда cat объединяет файлы и выводит их на стандартный вывод: использую команду cat чтобы прочитать файл hosts в подкаталоге etc корневого каталога, для этого в аргументе к команде указываю абсолютный путь к файлу:

```
[ekpestova@localhost-live ~]$ cat /etc/hosts
# Loopback entries; do not change.
# For historical reasons, localhost precedes localhost.localdomain:
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
# See hosts(5) for proper format and other examples:
# 192.168.1.10 foo.mydomain.org foo
# 192.168.1.13 bar.mydomain.org bar
```

5. Выполнение заданий для самостоятельной работы

1. Воспользовавшись командой `pwd` узнаю путь к своему домашнему каталогу:

```
[ekpestova@localhost-live ~]$ pwd
/home/ekpestova
```

2. Ввожу последовательность команд:

Сначала я возвращаюсь в домашнюю директорию (уже в ней), создаю в ней директорию `tmp`, перехожу в подкаталог домашнего каталога `tmp` с помощью `cd`. Если после этих действий я использую команду `pwd`, то получаю путь к директории `tmp`, начинающийся от корневого каталога, домашнего каталога пользователя, потому что именно в домашнем каталоге я сама создала директорию. Если я использую команду «`cd /tmp`», где `/` - корневой каталог, `tmp` – подкаталог корневого каталога, в котором содержатся временные файлы, эта директория есть в системе по умолчанию и путь к ней отличен он созданной мной директории `tmp`, поэтому при последующем использовании утилиты `pwd`, я получаю вывод `/tmp` (перехожу в разные каталоги `tmp`).

```
[ekpestova@localhost-live ~]$ cd
[ekpestova@localhost-live ~]$ mkdir tmp
[ekpestova@localhost-live ~]$ cd tmp
[ekpestova@localhost-live tmp]$ pwd
/home/ekpestova/tmp
[ekpestova@localhost-live tmp]$ cd /tmp
[ekpestova@localhost-live tmp]$ pwd
/tmp
```

3. Перехожу в корневой каталог с помощью `cd /`, просматриваю его содержимое с помощью `ls`, добавляю к утилите ключ `-a`, чтобы увидеть скрытые файлы «.» и «..» в директории:

```
[ekpestova@localhost-live tmp]$ cd /
[ekpestova@localhost-live /]$ ls
afs  boot  etc  lib  lost+found  mnt  proc  run  srv  tmp  var
bin  dev  home  lib64  media  opt  root  sbin  sys  usr
[ekpestova@localhost-live /]$ ls -a
.  bin  etc  lib64  lost+found  opt  run  sys  var
.. boot  home  .liveimg-configured  media  proc  sbin  tmp
afs  dev  lib  .liveimg-late-configured  mnt  root  srv  usr
```

Возвращаюсь в домашнюю директорию с помощью `cd`, указываю к директории абсолютный путь. Просматриваю с помощью `ls` содержимое домашнего каталога. Чтобы просмотреть содержимое со скрытыми файлами снова использую `ls -a`:

```
[ekpestova@localhost-live /]$ cd /home/ekpestova
[ekpestova@localhost-live ~]$ ls
Desktop  Downloads  parentdir  parentdir2  Pictures  Templates  Videos
Documents  Music  parentdir1  parentdir3  Public  tmp
[ekpestova@localhost-live ~]$ ls -a
.  .config  parentdir  tmp
.. Desktop  parentdir1  .vboxclient-clipboard.pid
.bash_history Documents  parentdir2  .vboxclient-draganddrop.pid
.bash_logout Downloads  parentdir3  .vboxclient-seamless.pid
.bash_profile .local  Pictures  Videos
.bashrc  .mozilla  Public
.cache  Music  Templates
```

Из домашней директории просматриваю содержимое каталога `etc` с помощью утилиты `ls`, указав абсолютный путь к искомому каталогу:

```
[ekpestova@localhost-live ~]$ ls /etc
abrt                                httpd                                printcap
adjtime                            idmapd.conf                         profile
aliases                           ImageMagick-7                       profile.d
alsa                               inittab                             protocols
alternatives                       inputrc                              pulse
anaconda                           ipp-usb                             qemu
anthy-unicode.conf                 iproute2                            qemu-ga
appstream.conf                     iscsi                                rc.d
asound.conf                         issue                               reader.conf.d
audit                              issue.d                             redhat-release
authselect                         issue.net                           request-key.conf
avahi                              java                                request-key.d
bash_completion.d                 jvm                                 resolv.conf
bashrc                             jvm-common                          rpc
bindresvport.blacklist             kdump                               rpm
binfmt.d                           kdump.conf                         rsyncd.conf
bluetooth                          kernel                              rwtab.d
brlapi.key                         krb5.conf                           rygel.conf
brlatty                            krb5.conf.d                        samba
brltty.conf                        ld.so.cache                        sane.d
ceph                                ld.so.conf                         sasl2
chromium                           ld.so.conf.d                       security
chrony.conf                        libaudit.conf                      selinux
cifs-utils                         libblockdev                        services
containers                         libibverbs.d                       sestatus.conf
credstore                          libnl                               sgml
credstore.encrypted               libreport                           shadow
crypto-policies                   libssh                              shadow-
csh.cshrc                         libuser.conf                       shells
csh.login                         libvirt                             skel
cups                               locale.conf                        sos
cupshelpers                       localtime                          speech-dispatcher
```

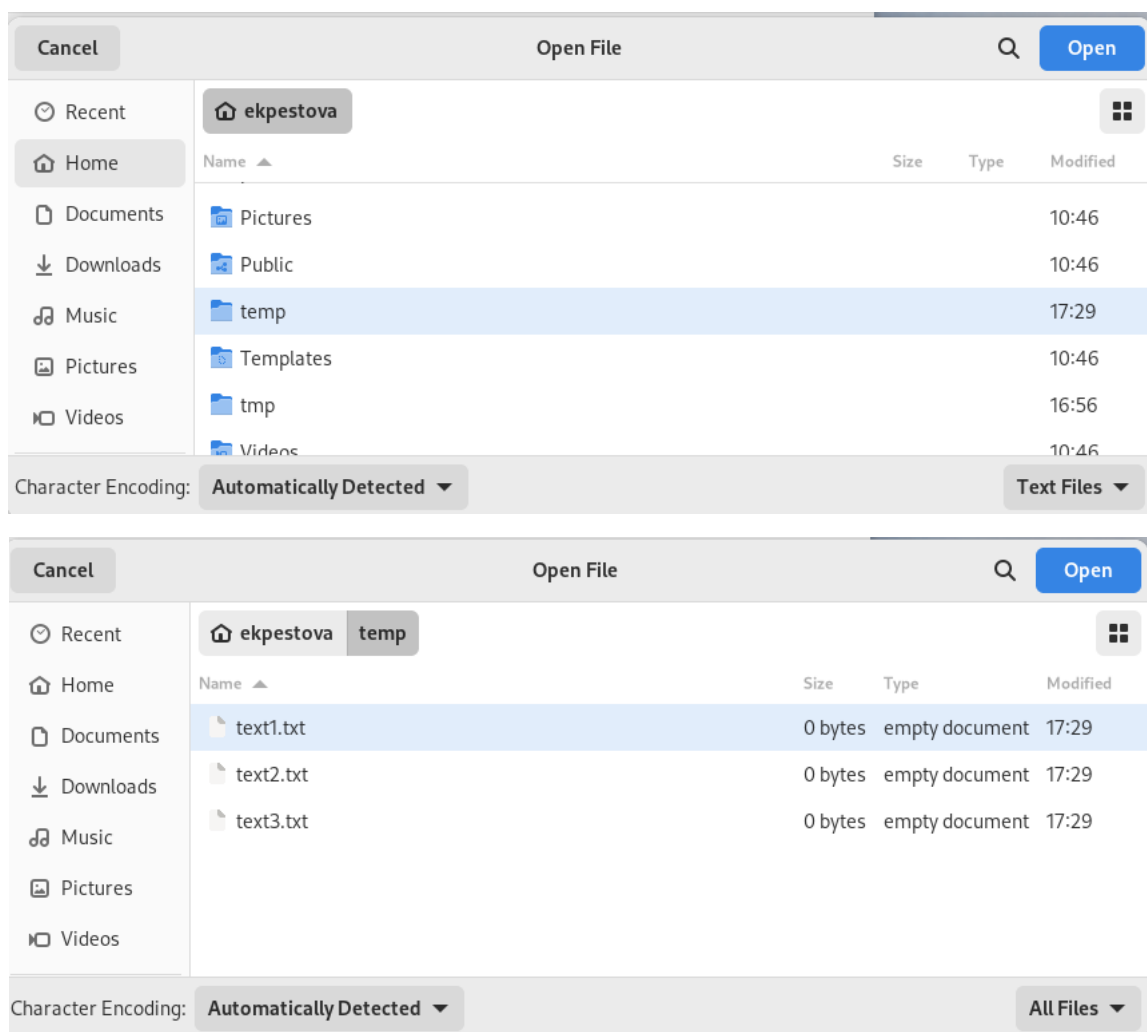
Перемещаюсь с помощью `cd` в каталог `/usr/local`. С помощью `ls` смотрю содержание этого каталога. Добавляю к утилите ключ `-a` и просматриваю всё содержимое каталога, включая скрытые файлы:

```
[ekpestova@localhost-live ~]$ cd /usr/local
[ekpestova@localhost-live local]$ ls
bin  etc  games  include  lib  lib64  libexec  sbin  share  src
[ekpestova@localhost-live local]$ ls -a
.  ..  bin  etc  games  include  lib  lib64  libexec  sbin  share  src
```

4. Возвращаюсь в домашний каталог. В нем с помощью утилиты `mkdir` создаю несколько каталогов (`temp` и `labs`) при этом `labs` нужно создавать с подкаталогами, для этого использую ключ `-p`. В каталоге `temp` создаю файлы `text1.txt`, `text2.txt`, `text3.txt` с помощью утилиты `touch`, все еще находясь в домашней директории. С помощью команды `ls temp` проверяю правильность создания файлов. В каталоге `temp` действительно есть три созданных файла. Аналогично, с помощью `ls labs` проверяю правильность создания подкаталогов в каталоге `labs`:

```
[ekpestova@localhost-live local]$ cd
[ekpestova@localhost-live ~]$ mkdir -p temp labs/lab1 labs/lab2 labs/lab3
[ekpestova@localhost-live ~]$ touch temp/text1.txt temp/text2.txt temp/text3.txt
[ekpestova@localhost-live ~]$ ls temp
text1.txt  text2.txt  text3.txt
[ekpestova@localhost-live ~]$ ls labs
lab1  lab2  lab3
```

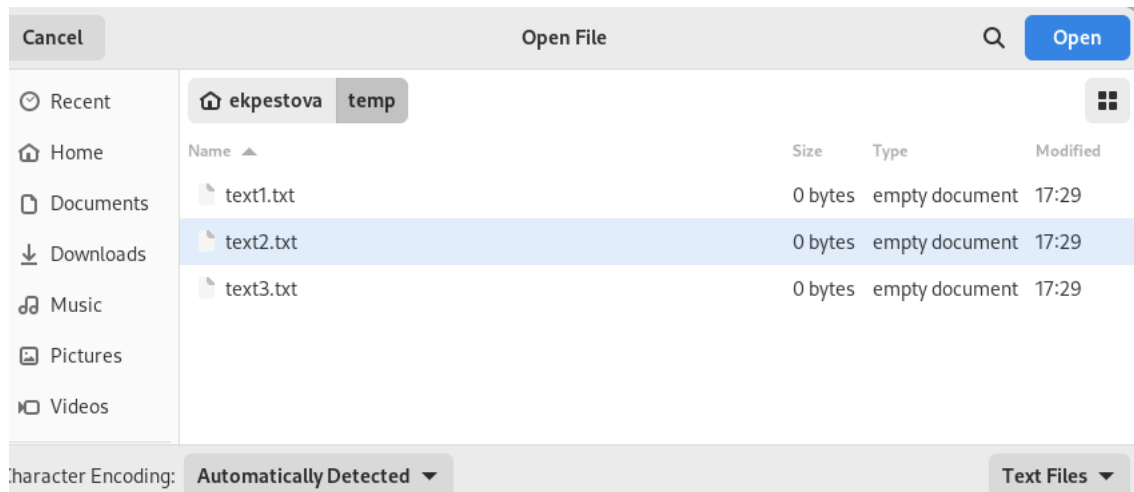
5. Открываю через меню приложений текстовый редактор, открываю выбираю во вкладке «файл» пункт «открыть». text.1. Открывается графический файловый менеджер, в нем выбираю путь к нужному файлу: домашний каталог, подкаталог temp. Выбираю нужный файл text1.txt:



Записываю в файл свое имя:



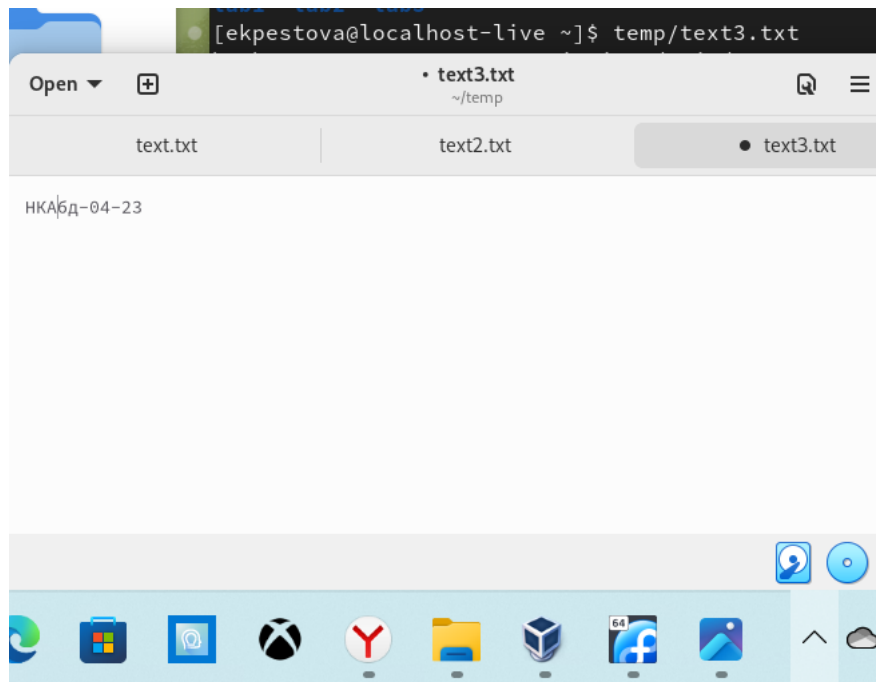
Открываю в текстовом редакторе файл text2.txt аналогично тому, как открывала text1.txt:



Записываю в файл text2.txt свою фамилию:



Записываю в файл text3.txt номер своей группы:



Проверяю правильность выполнения команд. Перехожу в каталог `temp` с помощью `cd`, использую утилиту `cat`, чтобы прочесть содержимое файлов `text.txt`, `text.2.txt`, `text3.txt`:

```
[ekpestova@localhost-live ~]$ cd temp
[ekpestova@localhost-live temp]$ cat text1.txt text2.txt text3.txt
Eva
Pestova
НКАбд-04-23
```

6. Копирую файлы, чьи имена заканчиваются на `.txt`, из каталога `~/temp` в каталог `labs`. Выбираю все файлы с помощью маски `«*»`, 23 обозначающей любое количество любых символов, копирую их с помощью утилиты `cp`. После этого переименовываю файлы каталога `labs` с помощью утилиты `mv`: `text1.txt` переименовываю в `firstname.txt` и перемещаю в подкаталог `lab1`, `text2.txt` переименовываю в `lastname.txt` и перемещаю в подкаталог `lab2`, `text3.txt` переименовываю в `id-group.txt` и перемещаю в подкаталог `lab3`. Воспользовавшись командой `ls`, я проверила содержание каталога `lab`.

С помощью `ls` проверяю содержание каждого подкаталога каталога `labs` и тут же читаю с помощью утилиты `cat` содержимое файла в каталоге, которое выводилось при применении прошлой команды: проверяю, какие файлы есть в директории `lab1`, читаю содержимое этого файла в командной строке. Аналогично для `lab2` и `lab3`:

```

[ekpestova@localhost-live ~]$ cp ~/temp/*.txt labs
[ekpestova@localhost-live ~]$ mv ~/labs/text1.txt ~/labs/lab1/firstname.txt
[ekpestova@localhost-live ~]$ mv ~/labs/text2.txt ~/labs/lab2/lastname.txt
[ekpestova@localhost-live ~]$ mv ~/labs/text3.txt ~/labs/lab3/id-group.txt
[ekpestova@localhost-live ~]$ ls labs
lab1 lab2 lab3
[ekpestova@localhost-live ~]$ ls labs/lab1
firstname.txt
[ekpestova@localhost-live ~]$ cat labs/lab1/firstname.txt
Eva
[ekpestova@localhost-live ~]$ ls labs/lab2
lastname.txt
[ekpestova@localhost-live ~]$ cat labs/lab2/lastname.txt
Pestova
[ekpestova@localhost-live ~]$ cat labs/lab3/id-group.txt
НКА6д-04-23

```

7. Я создавала новые директории только в домашнем каталоге, если рекурсивно удалить созданные в ходе лабораторной работы каталоги в домашнем каталоге, то все их подкаталоги и файлы в них тоже будут удалены. Использую ls, чтобы проверить содержимое домашнего каталога, ищу созданные в ходе лабораторной работы каталоги.

С помощью утилиты rm и ее ключа -R удаляю каталоги labs, temp, tmp, parentdir, parentdir1, parentdir2, parentdir3 вместе с их содержимым. Проверяю с помощью ls, удалились ли директории:

```

[ekpestova@localhost-live ~]$ ls
Desktop  labs      parentdir1  Pictures  Templates
Documents Music     parentdir2  Public    tmp
Downloads parentdir parentdir3  temp      Videos
[ekpestova@localhost-live ~]$ rm -R labs temp tmp parentdir parentdir1 parentdir2 parentdir3
[ekpestova@localhost-live ~]$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos

```

5. Вывод

При выполнении данной лабораторной работы я изучила организацию файловой системы, научилась создавать и удалять файлы и директории, приобрела практические навыки работы с операционной системой на уровне командной строки.

6. Ответы на контрольные вопросы для самопроверки.

1. Командная строка – это текстовый интерфейс между человеком и компьютером, в котором инструкции компьютеру даются путём ввода с клавиатуры текстовых строк.
2. Для получения достаточно подробной информации по каждой из команд можно использовать команду `man`: “`man ls`”
3. Абсолютный путь — начинается от корневого каталога (`/`), образуется перечислением всех каталогов, разделённых прямым слешем (`/`), и завершается именем файла. Относительный путь тоже строится перечислением через (`/`) всех каталогов, но начинается от каталога, в котором “находится” пользователь.
4. Определить абсолютный путь к текущей директории можно с помощью утилиты `pwd`.
5. При помощи команд `rmdir` и `rm` можно удалить файл и каталог? Командой `rmdir` нельзя удалить файлы, а командой `rm` можно удалить файлы и директории (с помощью опции `-r`). Утилита `rmdir` удаляет только пустые каталоги.
6. Запустить несколько команд в одной строке можно, перечисляя их через точку с запятой. Например: `cd /my_folder; rm *.txt`. Также можно использовать логические И и ИЛИ как `&` и `|` соответственно.
7. `-l` – выводит дополнительные параметры файлов (права доступа, владельцы и группы, размеры файлов и время последнего доступа)
8. Информацию о скрытых файлах можно просмотреть утилитой `ls` с ключами `-la`. `-l` – выведет дополнительную информацию о файлах, `-a` – выведет скрытые файлы. Можно использовать только ключ `-a`, если дополнительная информация о файле не нужна.
9. Для автоматического дополнения вводимых команд может служить клавиша `Tab`.

7. Источники

<https://esystem.rudn.ru/course/view.php?id=112>