МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное

образовательное учреждение высшего образования

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Отчёт по лабораторной работе №9

по дисциплине «Операционные системы»

Выполнила студентка ПИЖ-б-о-21-1

Кувшин Ирина Анатольевна

Работа защищена «  » 2022 г.

Проверил старший преподаватель

Говорова С.В.\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_

                       (оценка)       (подпись)

 Ставрополь 2022

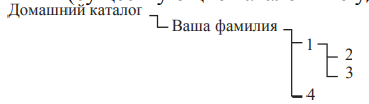
**ЛАБОРАТОРНАЯ РАБОТА №9.**

**ФАЙЛОВЫЕ СИСТЕМЫ ОС LINUX**

**Цель:**– практическое знакомство с организацией данных основной файловой системы ОС Linux и используемыми утилитами.

**ХОД ВЫПОЛНЕНИЯ**

**1. Создать в домашнем каталоге следующую структуру подкаталогов (существующие каталоги не удалять!):**

****

Каждая выполняемая программа "работает" в строго определенном каталоге файловой системы. Такой каталог называется текущим каталогом.

Команда **pwd** (print working directory) возвращает полный путь текущего каталога командной оболочки

В Linux у каждого пользователя обязательно есть собственный каталог, который и становится текущим сразу после регистрации в cистеме – домашний каталог.

Домашний каталог (home directory) - это каталог, предназначенный для хранения собственных данных пользователя Linux.

Полный путь к домашнему каталогу хранится в переменной окружения HOME.

**printenv** - Команда печатает все или указанные переменные среды.

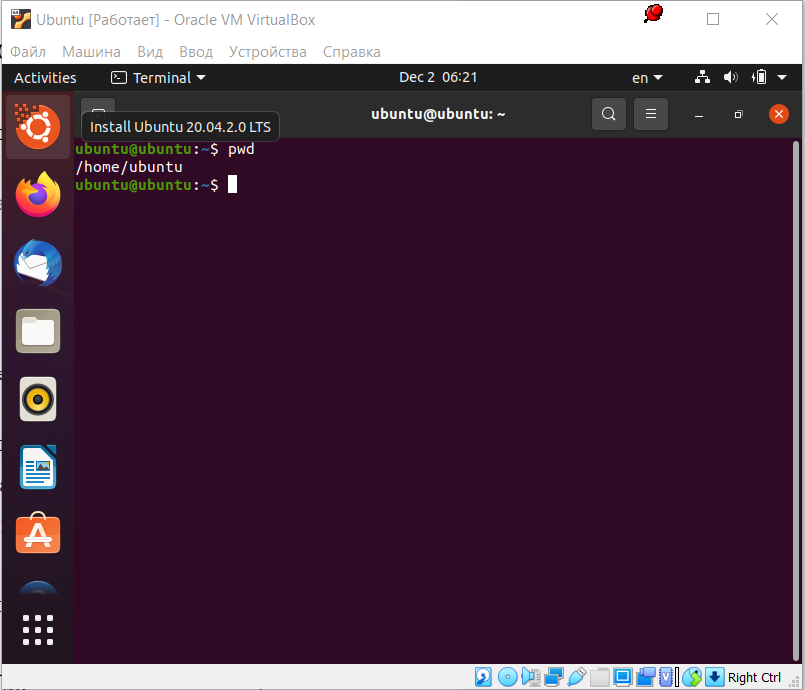
****

Рисунок 9.1 – Окно «Terminal», результат выполнения команды pwd

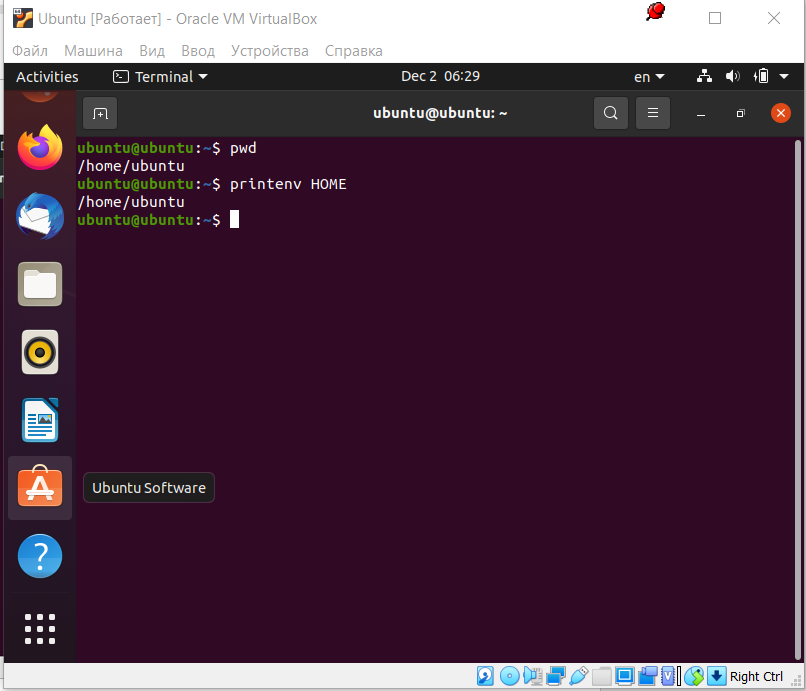


Рисунок 9.2 – Окно «Terminal», результат выполнения команды printenv HOME

В домашнем каталоге, как и в любом другом, можно создавать сколько угодно подкаталогов, в них - свои подкаталоги и т. д.

**mkdir** – утилита для создания каталогов.

Для перемещения по дереву каталогов используют команду cd и путь к каталогу в который хотят попасть. Различают два вида пути:

относительный – относительно текущего каталога;

абсолютный – всегда начинающийся с корня (/).

Еще при переходе по каталогам можно использовать служебные символы:

. — текущий каталог;

.. — родительский каталог;

– — предыдущий каталог;

~ — домашний каталог;

Создание каталога Kuvshin (рисунок 9.3):

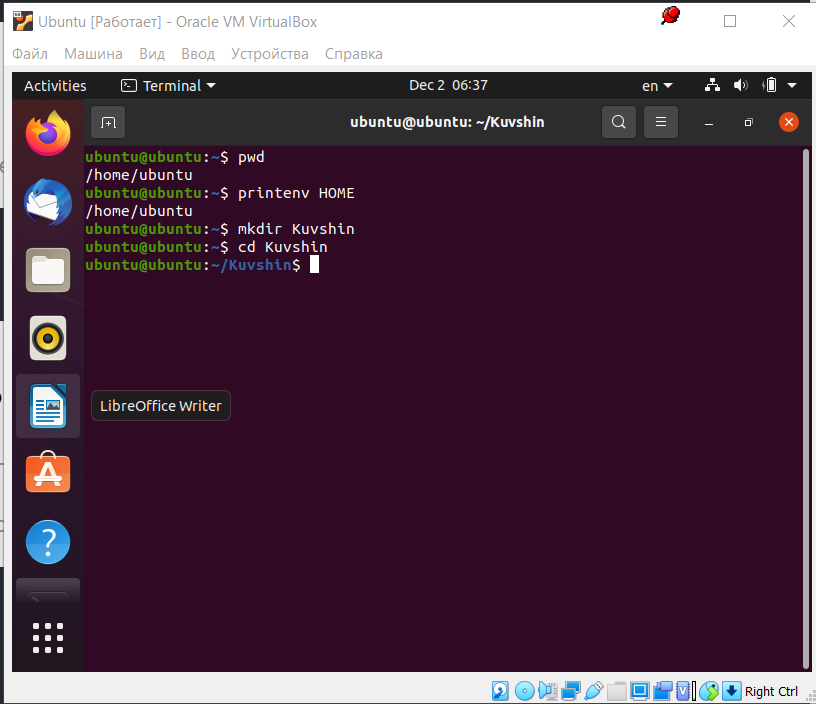


Рисунок 9.3 – Окно «Terminal», результат выполнения команд mkdir и cd

Создание структуры подкаталогов (рисунок 9.4):

**ls** – утилита для просмотра содержимого каталога

Параметр **-R** указывает команде **ls** рекурсивно отображать содержимое подкаталогов.

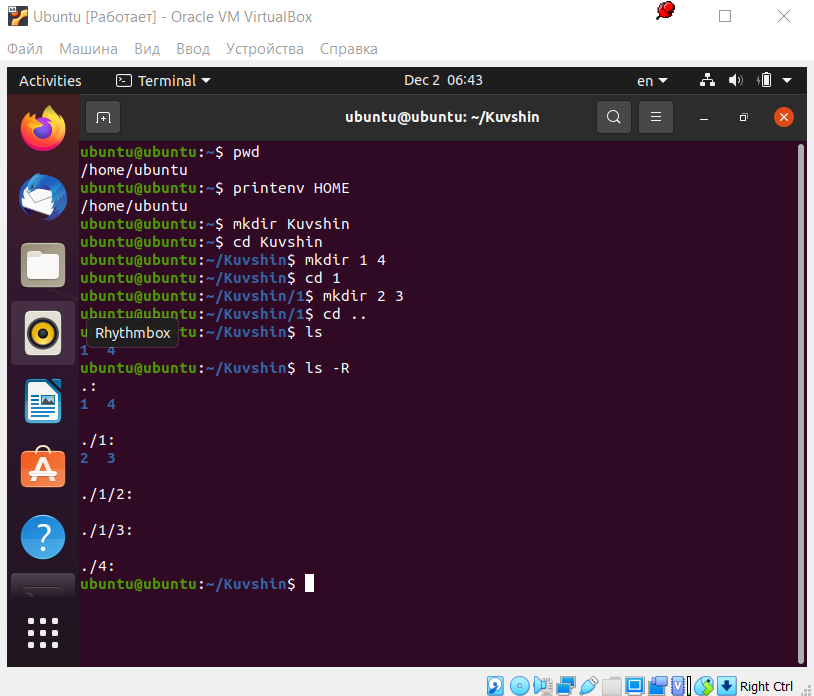


Рисунок 9.3 – Окно «Terminal», результат создания структуры подкаталогов

**2. Скопировать файл /etc/group в каталоги 1, 2, 3 и 4 используя абсолютные имена копируемого файла и каталога назначения.**

файл **/etc/group** содержит группы, в которые входит пользователь.

**mv** - используется для перемещения одного или нескольких файлов (или директорий) в другую директорию, а также для переименования файлов и директорий.

**cp** - представляет собой утилиту командной строки для копирования файлов и каталогов.

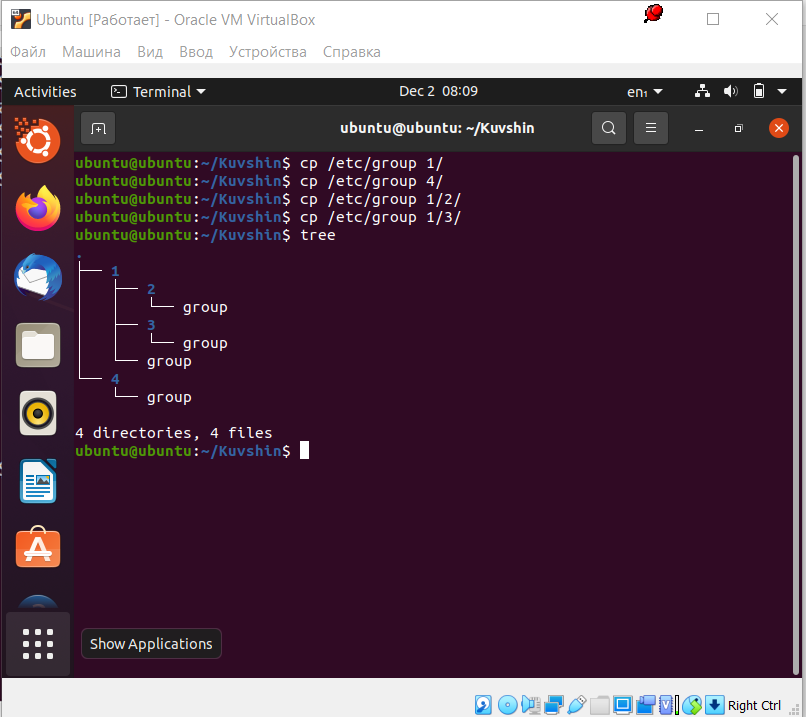


Рисунок 9.4 – Окно «Terminal», результат копирования файла

**3. С помощью утилиты file вывести на экран сведения о 3 - 4 различных файлах (в том числе из каталогов /bin и /dev).**

**file** - утилита, которая предназначена для определения типа содержащихся в файле данных.

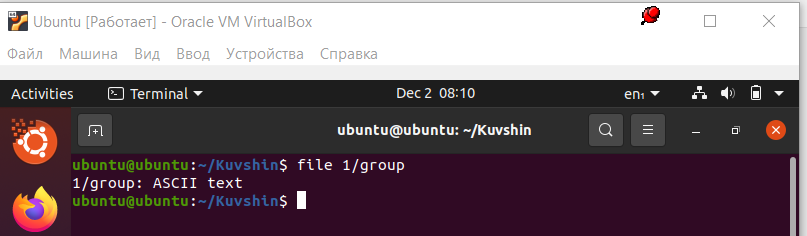


Рисунок 9.5 – Окно «Terminal», сведения о файле 1/group

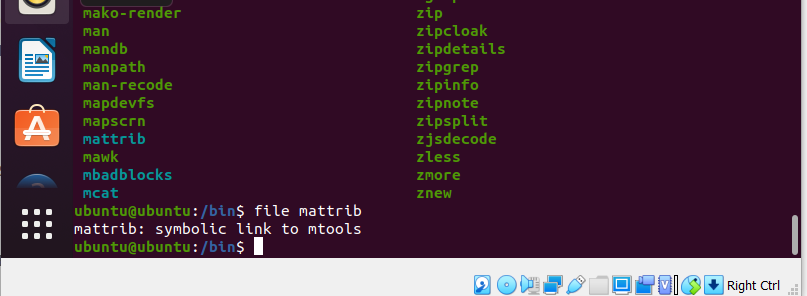


Рисунок 9.6 – Окно «Terminal», сведения о файле /bin/mattrib

Cимвольная ссылка (symbolic link, файл-ссылка) - это файл особого типа ("l"), в котором содержится путь к другому файлу.

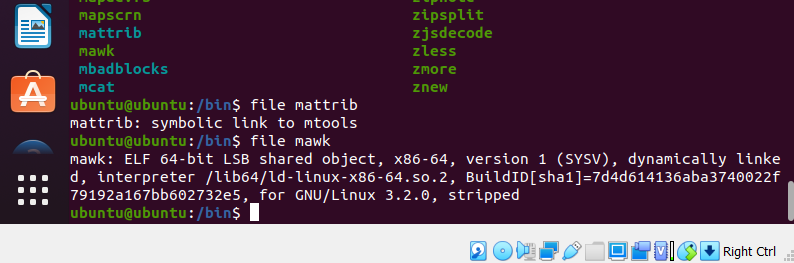


Рисунок 9.7 – Окно «Terminal», сведения о файле /bin/mawk

ELF — это сокращение от Executable and Linkable Format (формат исполняемых и связываемых файлов) и определяет структуру бинарных файлов, библиотек, и файлов ядра (core files). Спецификация формата позволяет операционной системе корректно интерпретировать содержащиеся в файле машинные команды.

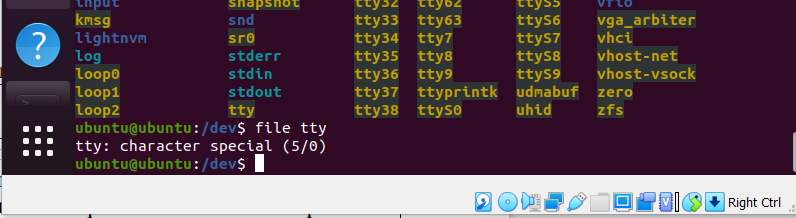


Рисунок 9.8 – Окно «Terminal», сведения о файле /dev/tty

character special file – символьный специальный файл. Специальный символьный файл - это файл, который обеспечивает доступ к устройству ввода / вывода.

**4. Выполнить команду ls –l /dev используя таблицу 2 обозначений типов файлов**

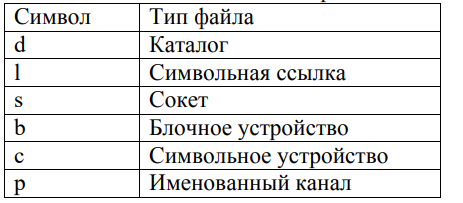


Таблица 2. Обозначения типов файлов.

перечислить типы файлов, хранящихся в каталоге /dev

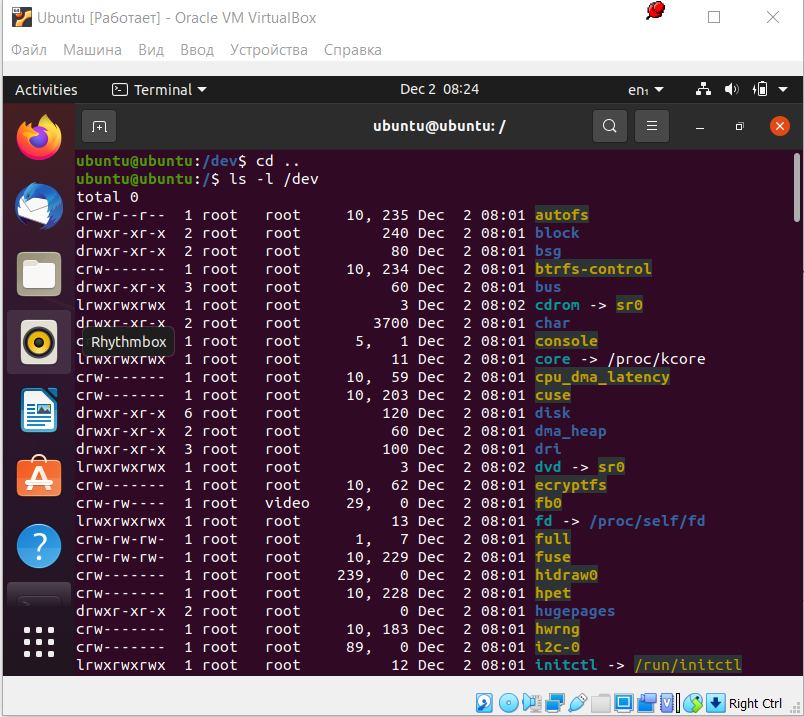


Рисунок 9.9 – Окно «Terminal», результат выполнения команды ls –l /dev

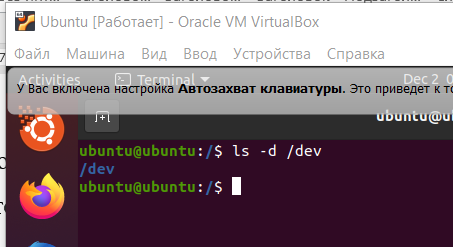


Рисунок 9.10 – Окно «Terminal», результат выполнения команды ls –d /dev

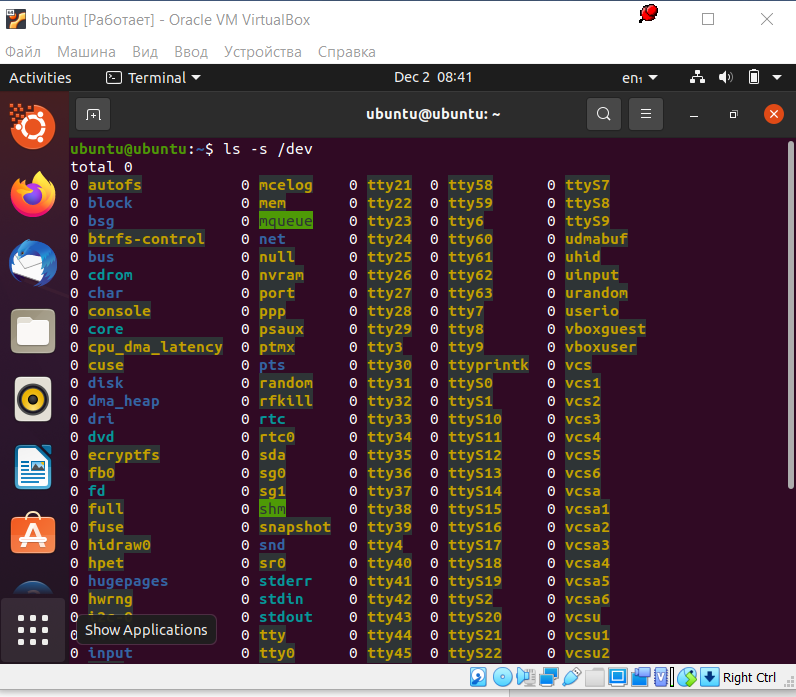


Рисунок 9.11 – Окно «Terminal», результат выполнения команды ls –s /dev

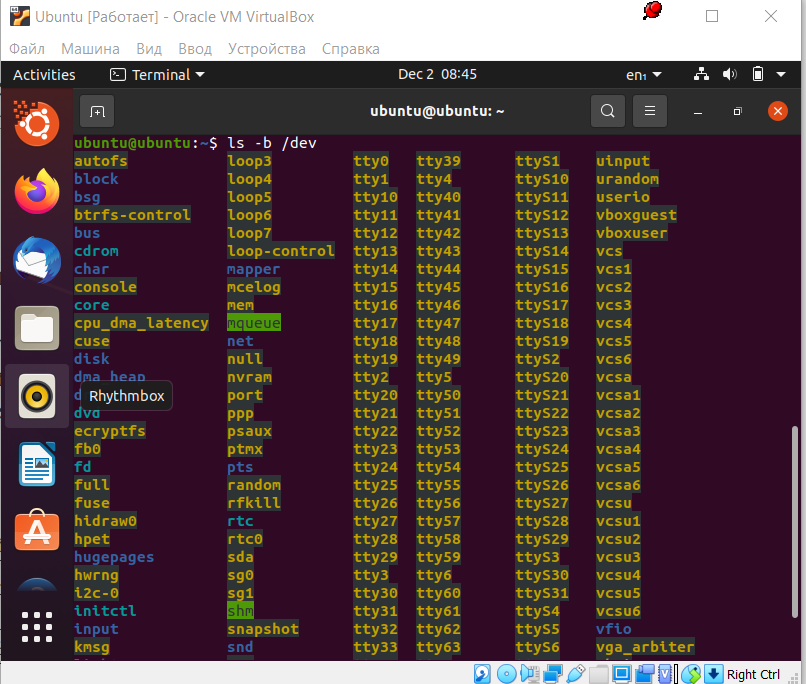


Рисунок 9.12 – Окно «Terminal», результат выполнения команды ls –b /dev

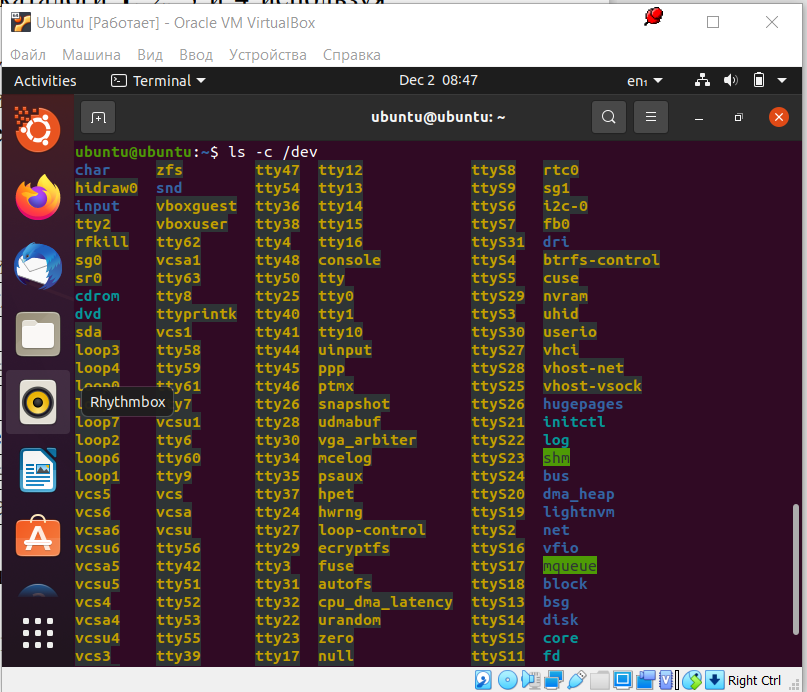


Рисунок 9.13 – Окно «Terminal», результат выполнения команды ls –c /dev

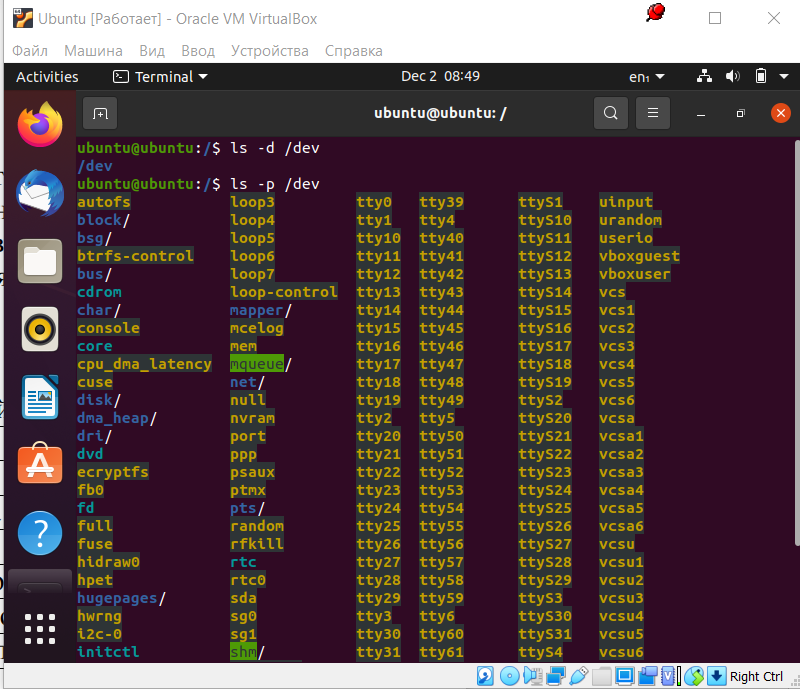


Рисунок 9.14 – Окно «Terminal», результат выполнения команды ls –p /dev

/dev

В этом каталоге находятся все имеющиеся в системе файлы особого типа, предназначенные для обращения к различным системным ресурсам и устройствам.

**5. Используя справочную систему, ознакомиться с ключами утилиты ls -R, -1 (единица), -m, --color, ключи, определяющие порядок вывода на экран**

**-R** - рекурсивно отображать содержимое поддиректорий;

**-1** - отображать один файл на одну строку;

**--color** - включить цветной режим вывода, автоматически активирована во многих дистрибутивах; (По умолчанию каждый из цветов, которые он выделяет при выполнении команды ls, имеет значение, поскольку это способ классификации файлов по их характеристикам.

зеленый: Исполняемые файлы

черный: Обычный файл

синий: Каталоги или папки

Небесный: Символическая ссылка

красный: Сжатые файлы (.tar, .gz, .zip, .rpm)

фуксин: Файлы изображений (.jpg, gif, bmp, png, tif

желтый – FIFO (FIFO) - это средство передачи данных между процессами))

-**m** - разделять элементы списка запятой;

**-S** - сортировать по размеру, сначала большие;

**-t** - сортировать по времени последней модификации;

**-u** - сортировать по времени последнего доступа;

**-U** - не сортировать;

**-X** - сортировать по алфавиту;

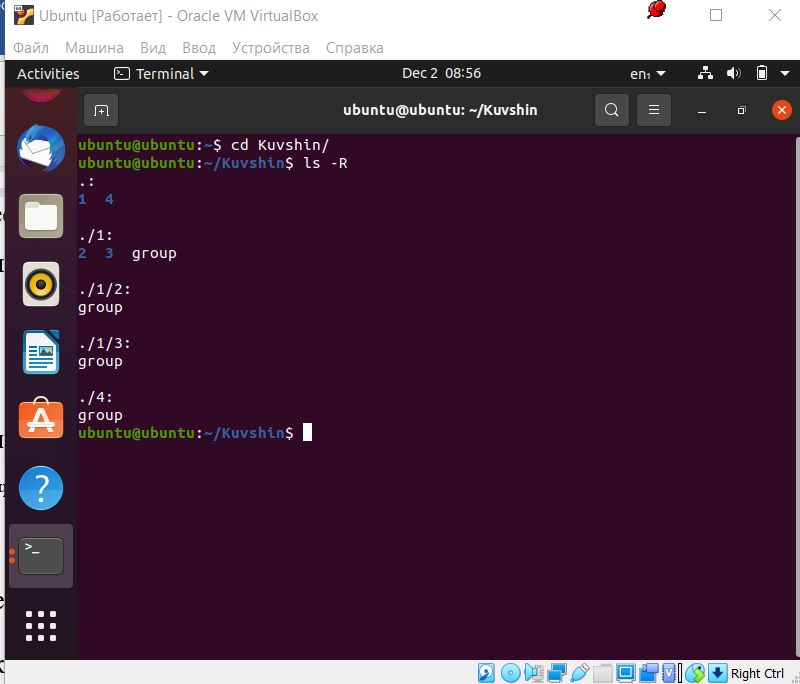


Рисунок 9.15 – Окно «Terminal», результат выполнения команды ls –R

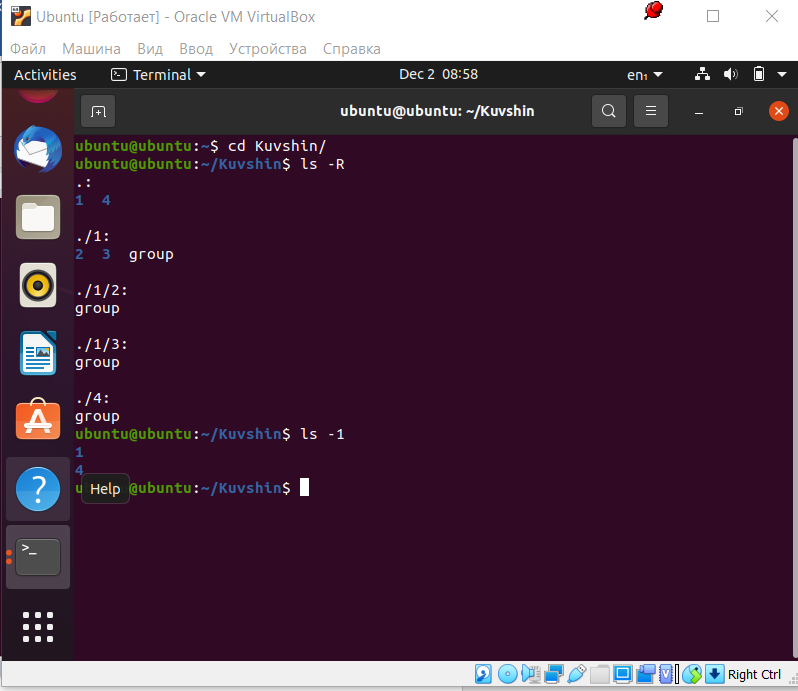


Рисунок 9.16 – Окно «Terminal», результат выполнения команды ls –1

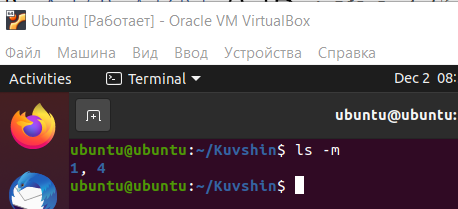


Рисунок 9.17 – Окно «Terminal», результат выполнения команды ls –m

**6. Создать жесткую и символическую ссылки для одного из созданных в п.2 файлов**.

Каждый файл представляет собой область данных на жестком диске компьютера или на другом носителе информации, которую можно найти по имени. В файловой системе Linux содержимое файла связывается с его именем при помощи жестких ссылок. Создание файла с помощью любой программы означает, что будет создана жесткая ссылка - имя файла, и открыта новая область данных на диске.

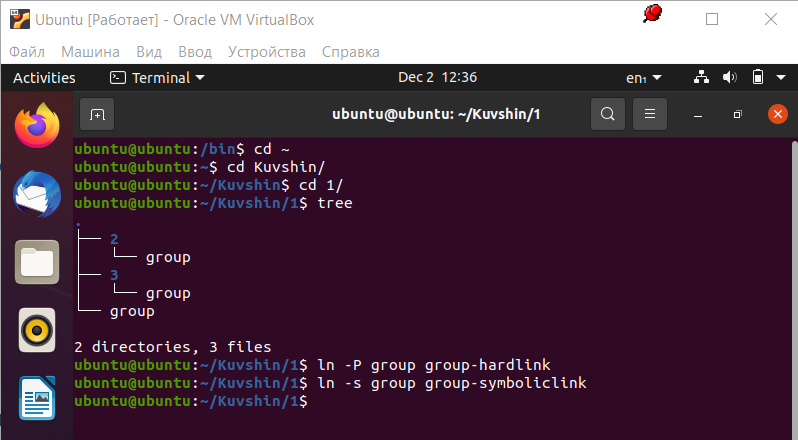
Символические ссылки более всего похожи на обычные ярлыки. Они содержат адрес нужного файла в вашей файловой системе.

**$ ln опции файл\_источник файл\_ссылки**

-**P** - создать жесткую ссылку;

-**s** - создать символическую ссылку.

Команда **cat** читает данные из файла или стандартного ввода и выводит их на экран.

Рисунок 9.18 – Окно «Terminal», результат создание ссылок

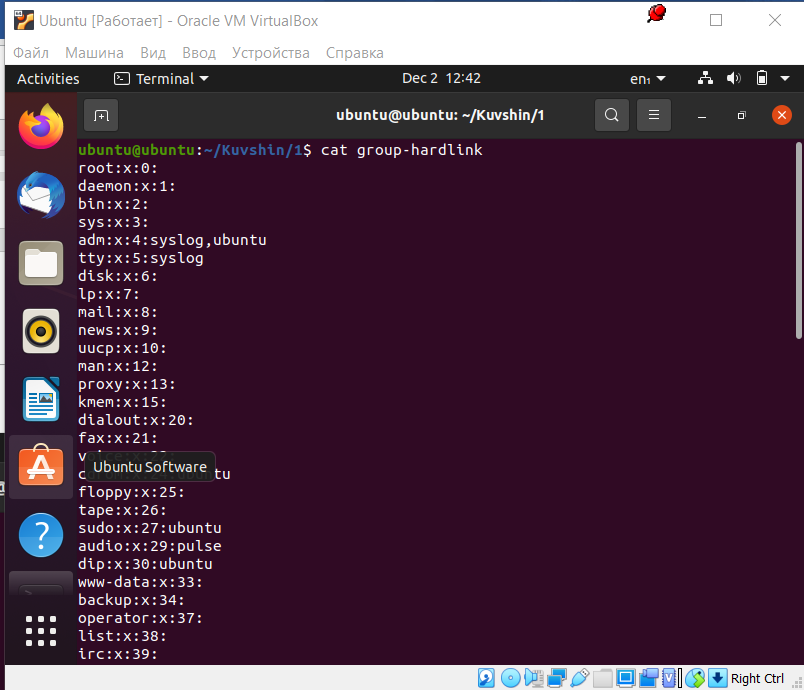


Рисунок 9.19 – Окно «Terminal», результат открытия жесткой ссылки

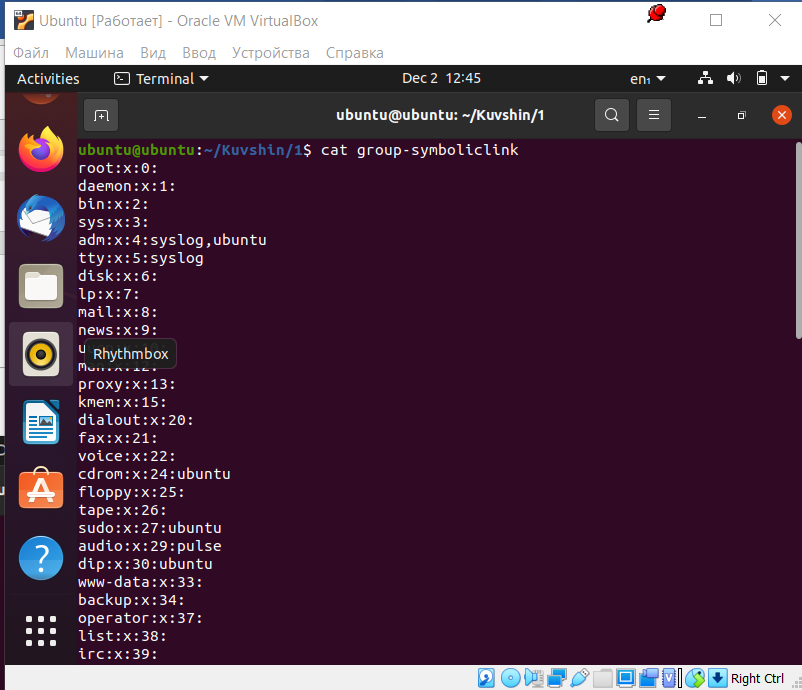
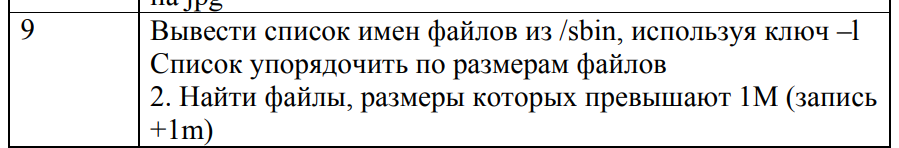


Рисунок 9.20 – Окно «Terminal», результат открытия символической ссылки

**Индивидуальное задание:**

****

**-S** - сортировать по размеру, сначала большие;

**find** - производит поиск файлов и папок на основе заданных вами критериев и позволяет выполнять действия с результатами поиска.

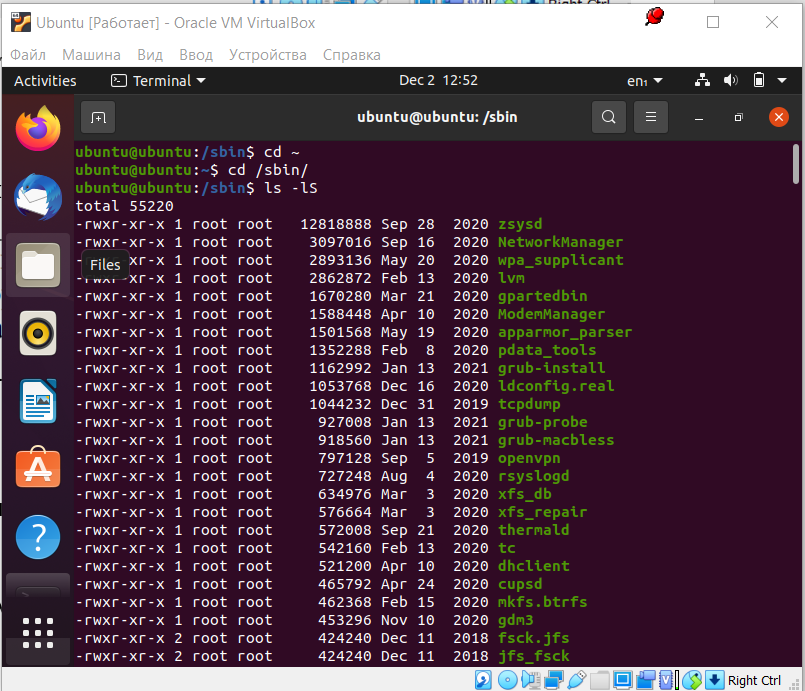
****

Рисунок 9.21 – Окно «Terminal», результат выполнения 1го задания

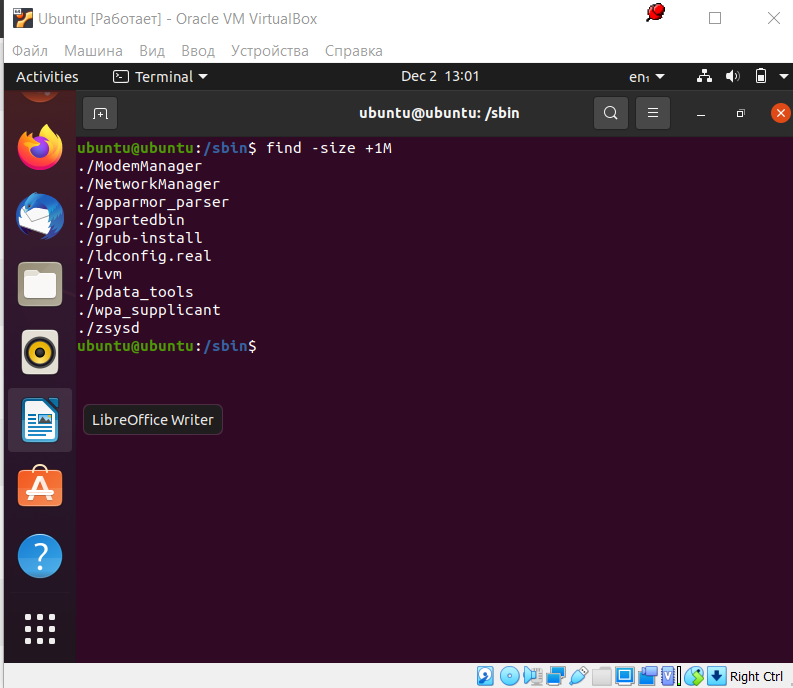
****

Рисунок 9.22 – Окно «Terminal», результат выполнения 2го задания

**Вывод:** В результате выполнения данной работы было проведен практическое знакомство с организацией данных основной файловой системы ОС Linux и используемыми утилитами.

**Контрольные вопросы**

1. Типы файлов ОС Linux.

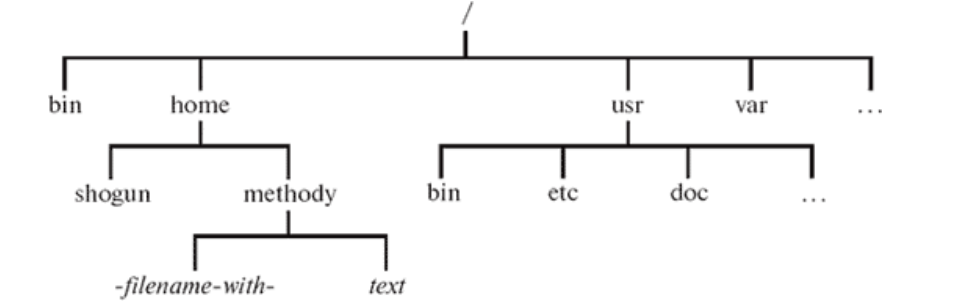
В Linux определено несколько различных типов файлов. В основном пользователь имеет дело с файлами трех типов: обычными файлами, предназначенными для хранения данных, каталогами и файлами-ссылками.

2. Назначение утилиты file.

В Linux есть утилита file, которая предназначена для определения типа содержащихся в файле данных.

3. Структура дерева каталогов ОС Linux.

Структуру файловой системы можно представить наглядно в виде дерева, "корнем" которого является корневой каталог, а в вершинах расположены все остальные каталоги.



4. Отличия структуры файловых систем ОС Windows и Linux.

В любой файловой системе Linux всегда есть только один корневой каталог, который называется "/". Пользователь Linux всегда работает с единым деревом каталогов, даже если разные данные расположены на разных носителях: жестких или сетевых дисках, съемных дисках, CD-ROM и т. п. Такое представление отличается от технологии, применяемой в Windows, где для каждого устройства, на котором есть файловая система, используется свой корневой каталог, обозначенный литерой, например "a", "c", "d" и т. д. Для того чтобы отключать и подключать файловые системы на разных устройствах в состав одного общего дерева, используются процедуры монтирования и размонтирования. После того, как файловые системы на разных носителях подключены к общему дереву, содержащиеся на них данные доступны так, как если бы все они составляли единую файловую систему: пользователь может даже не знать, на каком устройстве какие файлы хранятся.

5. В чем отличие каталогов /var и /tmp.

/var Название этого каталога - сокращение от "variable" ("переменные" данные). Здесь размещаются те данные, которые создаются в процессе работы разными программами и предназначены для передачи другим программам и системам (очереди печати, электронной почты и др.) или для сведения системного администратора (системные журналы, содержащие протоколы работы системы). В отличие от каталога /tmp сюда попадают те данные, которые могут понадобиться после того, как создавшая их программа завершила работу.

6. Назначение утилиты pwd.

Команда pwd (print working directory) возвращает полный путь текущего каталога командной оболочки - естественно, именно той командной оболочки, при помощи которой была выполнена команда pwd.

7. Назначение утилиты cat.

Например, утилита cat (concatenation – конкатенация) - выводит на экран содержимое файла "text":

[student@localhost student]$ cat text

8. Назначение утилиты ls. Использование ключей –F, -a.

Команда ls без параметров выводит список файлов и каталогов, содержащихся в текущем каталоге. Утилита ls принимает один параметр - имя каталога, содержимое которого нужно вывести.

Ключ –F используется, чтобы отличать файлы от каталогов. При наличии этого ключа ls в конце имени каждого каталога ставит символ "/", чтобы показать, что в нем может содержаться что-то еще.

Ссылки на текущий и на родительский каталог обязательно присутствуют в каждом каталоге в Linux. Даже если каталог пуст, т. е. не содержит ни одного файла или подкаталога, команда "ls -a" выведет список из двух имен: "." и "..".

9. Утилита mkdir.

**mkdir** – утилита для создания каталогов.

10.Утилиты копирования и перемещения файлов.

Для перемещения файлов и каталогов предназначена утилита mv (от англ. "move" - "перемещать").

Иногда требуется создать копию файла: для большей сохранности данных, для того, чтобы создать модифицированную версию файла и т. п. В Linux для этого предназначена утилита cp (от англ. "copy" - "копировать").

11.Жесткие ссылки: назначение и создание.

Каждый файл представляет собой область данных на жестком диске компьютера или на другом носителе информации, которую можно найти по имени. В файловой системе Linux содержимое файла связывается с его именем при помощи жестких ссылок. Создание файла с помощью любой программы означает, что будет создана жесткая ссылка - имя файла, и открыта новая область данных на диске. Причем количество ссылок на одну и ту же область данных (файл) не ограничено, то есть у файла может быть несколько имен.

12.Cоздание файлов.

Для создания пустого файла с текущим временем создания служит команда touch имя\_нового\_файла. Для указания даты создания в формате ГГГГММДДhhmm используется ключ –t.

Например

touch –t 0904080000 tst файл создан 8 апреля 2015 г

13.Символьные ссылки.

Cимвольная ссылка (symbolic link, файл-ссылка) - это файл особого типа ("l"), в котором содержится путь к другому файлу. Если на пути к файлу встречается cимвольная ссылка, система выполняет подстановку: исходный путь заменяется тем, что содержится в ссылке. Cимвольную ссылку можно создать при помощи команды ln с ключом "-s" (сокращение от "symbolic").

14.Удаление файлов и каталогов. Как восстановить ошибочно удаленный файл?

В ОС Linux для удаления файлов предназначена утилита rm (сокращение от англ. "remove" - "удалять")

15.Назначение утилиты id.

Утилита id выводит входное имя пользователя и соответствующий ему UID, а также группу по умолчанию и полный список групп, членом которых он является

16.Ярлыки объектов файловой системы.

При создании объектов файловой системы – файлов, каталогов и т. п. – каждому приписывается ярлык. Ярлык включает в себя UID – идентификатор пользователя-хозяина файла, GID – идентификатор группы, которой принадлежит файл, тип объекта и набор так называемых атрибутов (код доступа), а также некоторую дополнительную информацию.

17.Права доступа к файлу.

В Linux определено несколько системных групп, задача которых – обеспечивать доступ членов этих групп к разнообразным ресурсам системы. Часто такие группы носят говорящие названия: "disk", "audio", "cdwriter" и т. п. Если обычным пользователям доступ к некоторому файлу, каталогу или специальному файлу Linux закрыт, он открыт членам группы, которой этот объект принадлежит.

18.Суперпользователь и его права.

Суперпользователь - единственный пользователь в Linux, на которого не распространяются ограничения прав доступа. Имеет нулевой идентификатор пользователя.

19.Назначение утилиты sudo.

Второй способ —воспользоваться специальной утилитой sudo, которая позволяет выполнить одну или несколько команд от лица другого пользователя. По умолчанию эта утилита выполняет команду от лица пользователя root, то есть запускает командный интерпретатор с нулевым UID. Отличие от предыдущего способа в том, что всегда известно, кто именно запускал sudo, а значит, ясно, с кого спрашивать за последствия.

20.Утилиты поиска файлов locate и find, их достоинства и недостатки.

Для поиска файла по имени или его части используется утилита locate. Параметр задает имя файла. Для поиска без учета регистра служит ключ –i. Для ограничения объема выводимой информации используется ключ –n число. Построчный вывод получается, если результаты поиска направить по конвейеру в программу less, например locate mp3 | less Утилита locate ведет поиск в базе данных, которая должна периодически обновляться утилитой updatedb, выполняемой с правами администратора. Другой способ найти файл предоставляет утилита find.

Достоинствами утилиты find являются независимость от базы данных и широкие функциональные возможности, недостаток – меньшая скорость поиска по сравнению с locate.