## Факультет информационных технологий и робототехники

Кафедра программного обеспечения информационных систем и технологий

**Отчет по лабораторной работе № 2**

по дисциплине: ”Системное программирование”

на тему: *”Основы работы с терминалом и командной строкй UNIX”*

Вариант -

Выполнили:

студенты группы 10701222

Медведский Е.В.

Зухта К.М.

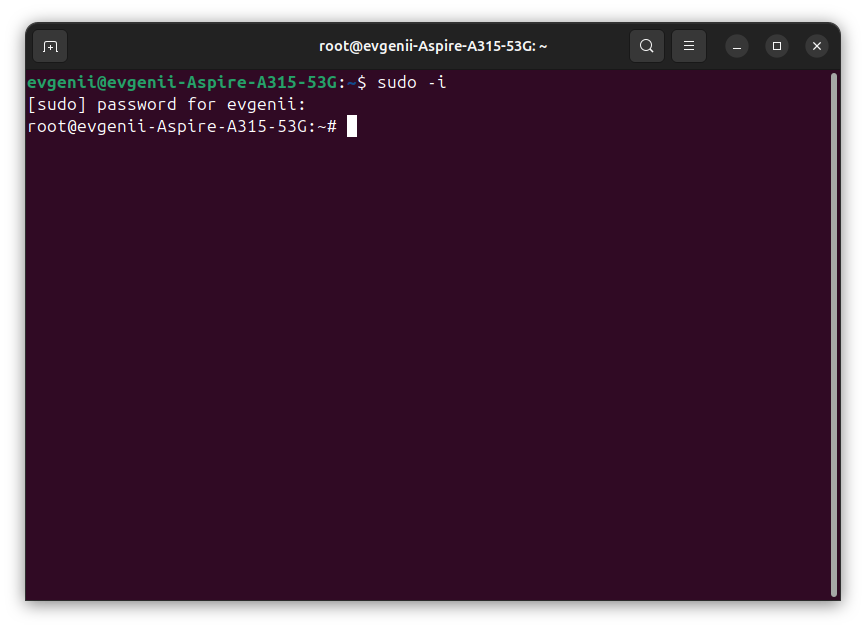
Принял**:** пр. Давыденко Н.В.

Минск 2024

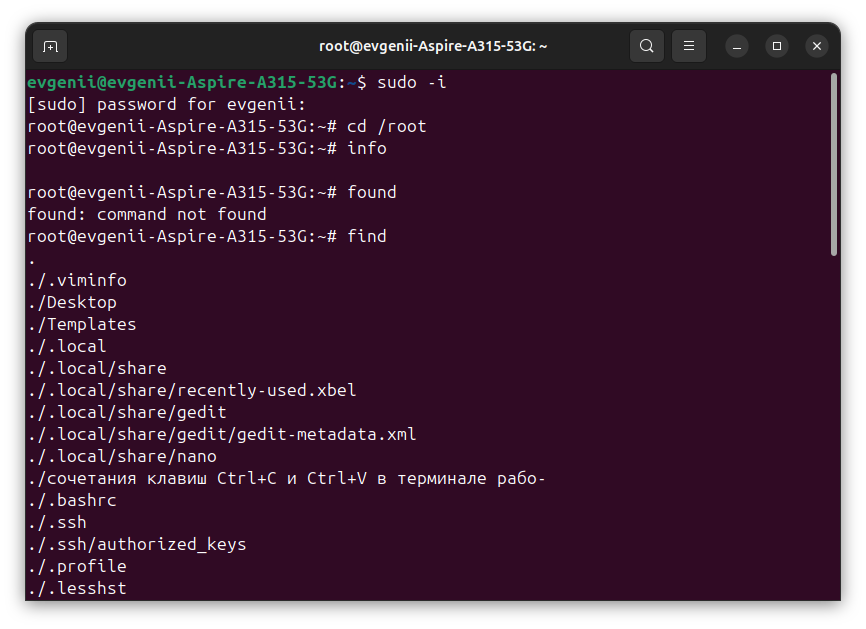
**Цель работы:** Изучить командную строку UNIX и получить основы работы с терминалом в дистрибутивах Linux (UNIX).

**Задание1**

Зайдите в корневую директорию root и получите все доступные каталоги. Выведите все файлы и директории в данном каталоге (root)

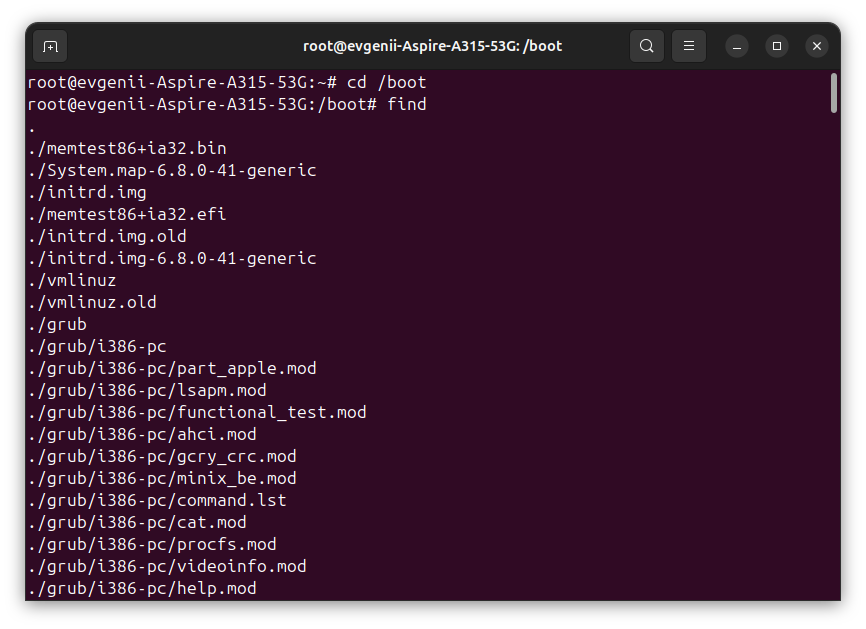


переходим в корневую папку root



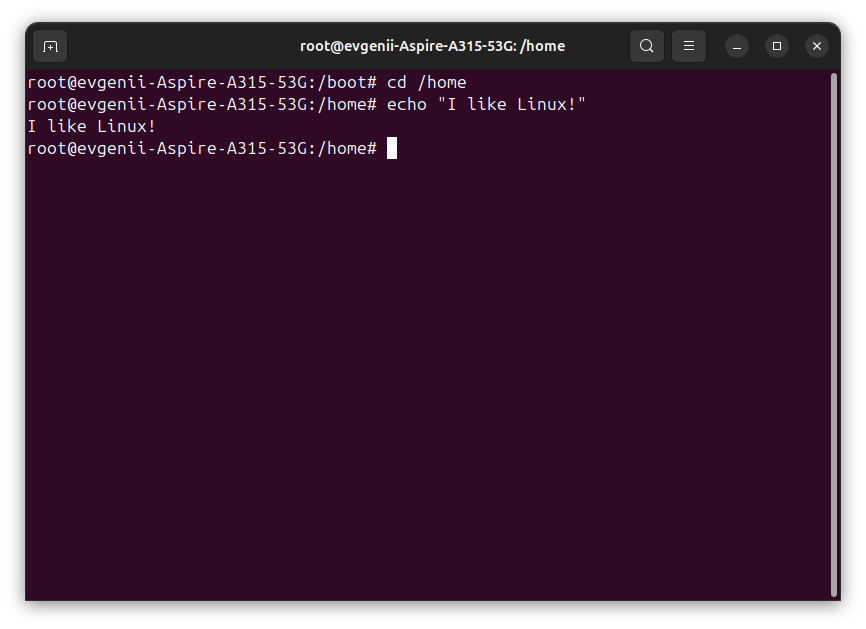
все каталоги внутри root

Получите данные о вашей системе. Найдите исполняемый файл ядра Linux. В какой директории он находится?



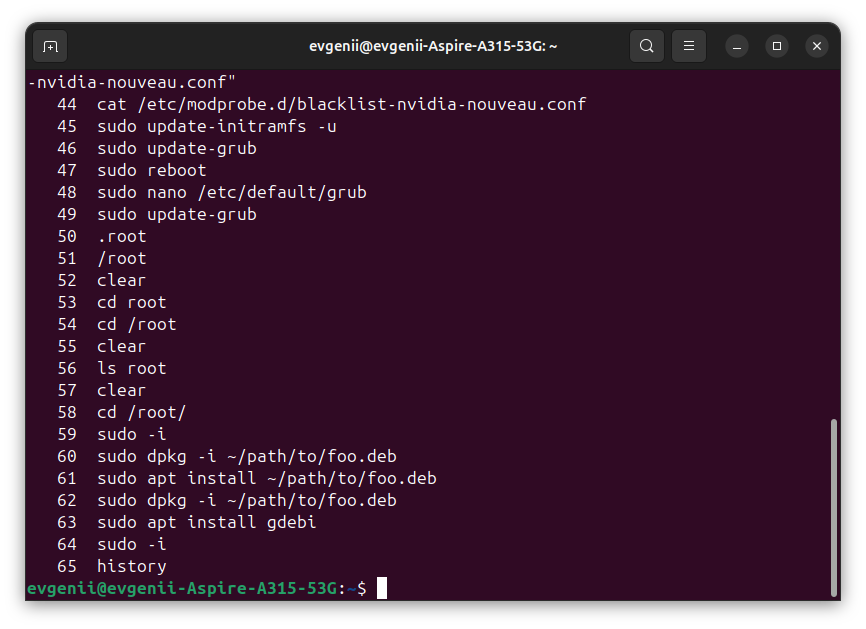
находим файлы ядра в каталоге /boot

Вернитесь в домашний каталог пользователя (home). Выведите сообщение «I‘m like Linux!».

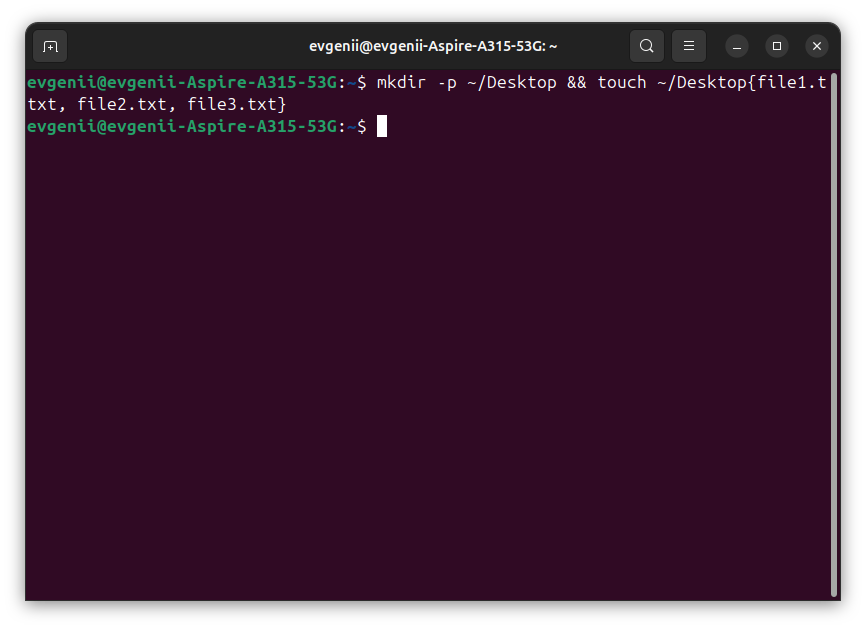


также для перехода в домашний каталог можно использовать команду cd~

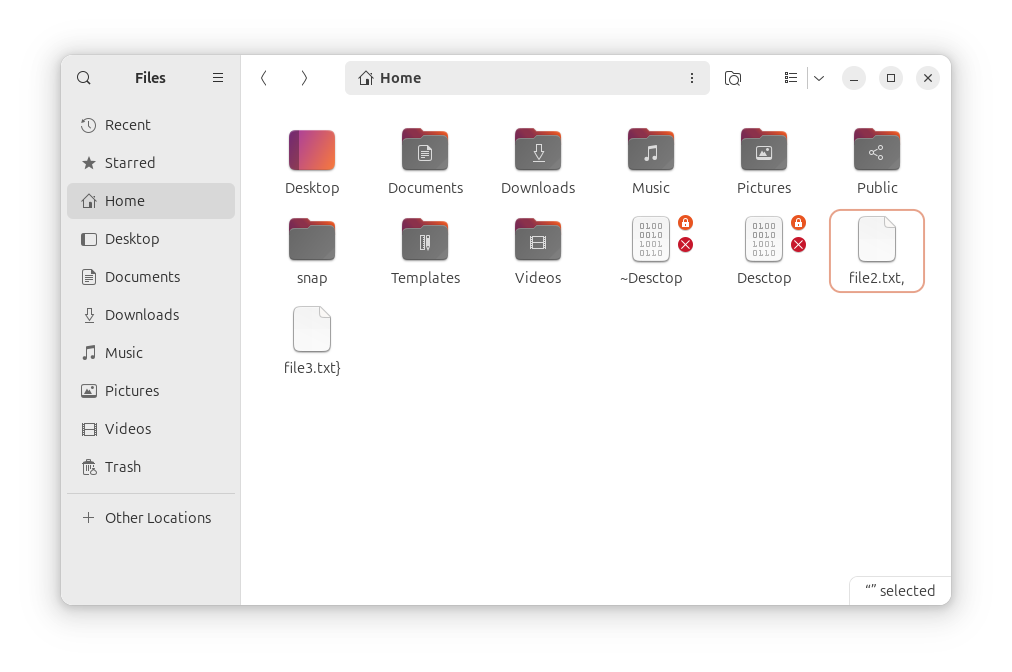
Получите историю введенных команд.



Создайте директорию на рабочем столе. Внутри П этой директории создайте 3 текстовых файла одним действием.



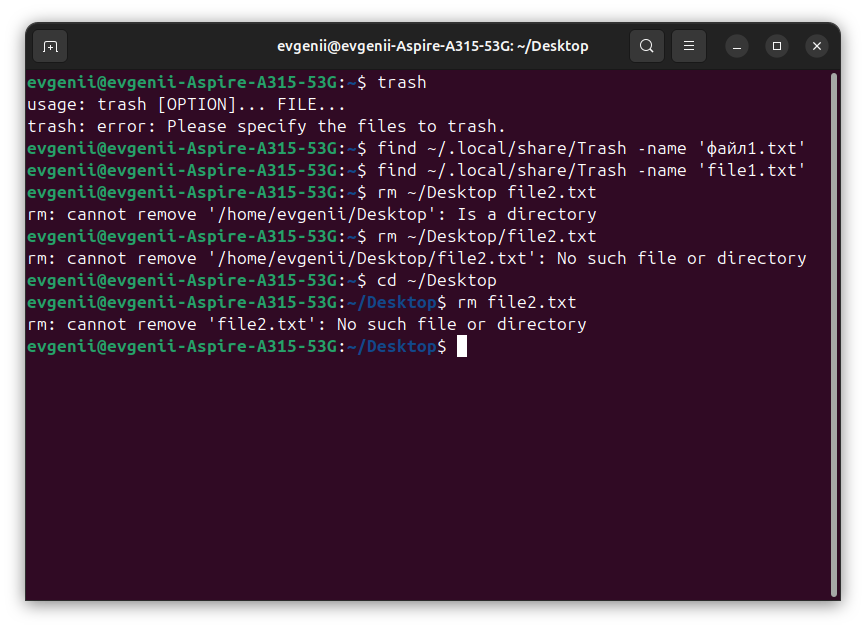
Удалите один из созданных файлов с помощью мыши (в графическом интерфейсе), а другой файл с помощью консольной команды. Далее с помощью консольной команды попробуйте найти удаленные файлы. Посмотрите атрибуты найденного файла. Объясните в чем разница этих способов удаления файлов.



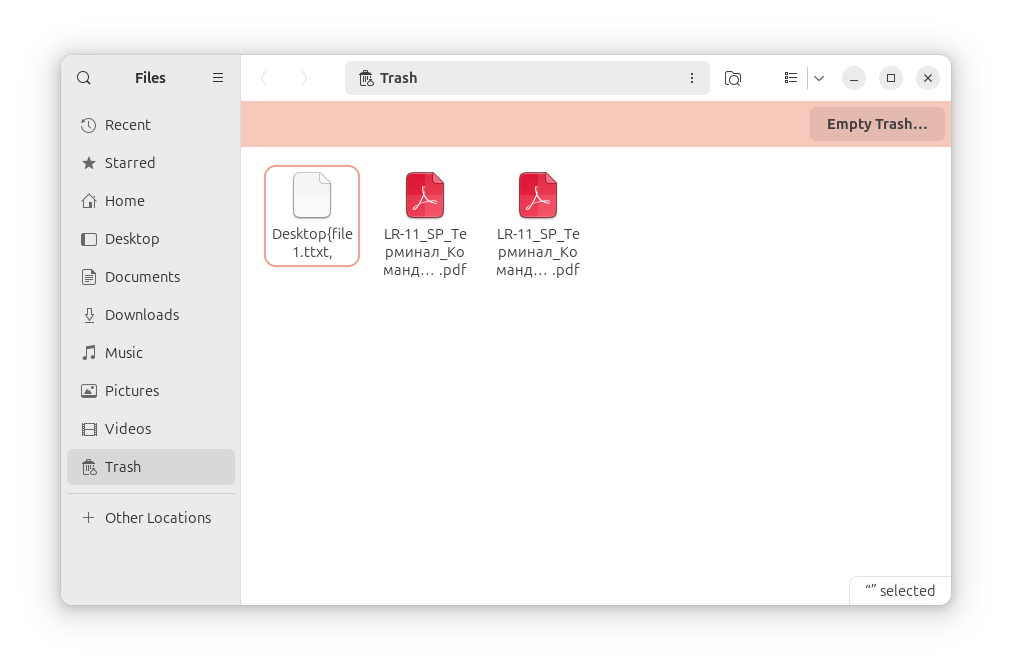
удалил файл file1.txt с помощью графического интерфейса

Когда вы удаляете файл с помощью графического интерфейса (например, через файловый менеджер), файл фактически не удаляется окончательно. Он перемещается в "Корзину" (Trash). Это дает возможность восстановить файл, если удаление было ошибочным.

Удаление файла через консоль командой rm приводит к немедленному удалению файла, и он **не перемещается в "Корзину"**. Файл считается удалённым, и его восстановление становится сложным (возможно, с помощью специальных инструментов для восстановления данных, но это уже другой процесс).



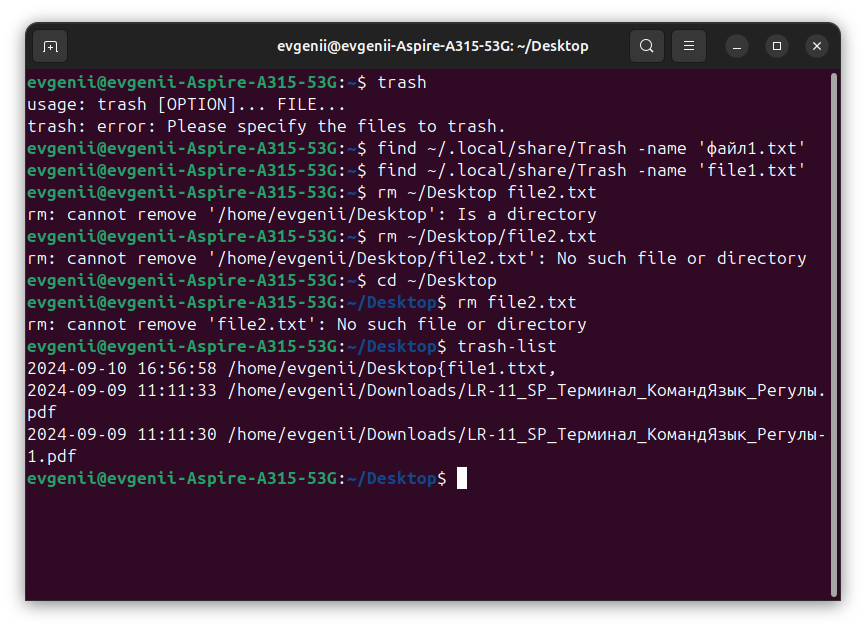
удаление файла file2.txt при помощи терминала



удаленный файл при помощи графического интерфейса в корзине

В случае если при удалении файла с помощью консольной команды файл пропал безвозвратно, попробуйте удалить подобный файл с помощью команды (утилиты) trash-cli, при необходимости утилиту скачайте и установите. Объясните, как работает утилита.

Чтобы увидеть файлы, которые находятся в корзине, используйте команду trash-list



Чтобы восстановить файл из корзины, используйте команду trash-restore

Если вы хотите окончательно удалить файл из корзины, используйте команду:trash-empty

### Как работает утилита trash-cli

trash-cli работает следующим образом:

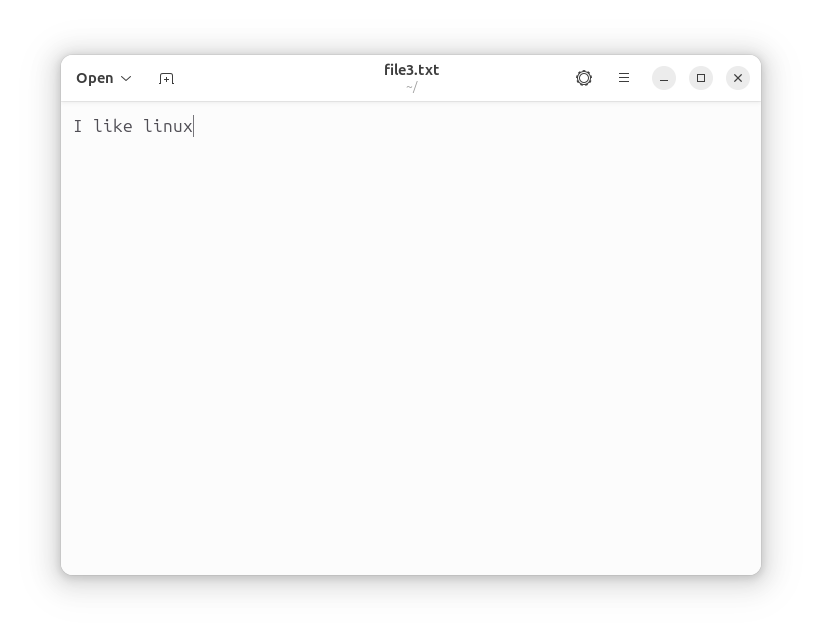
**Перемещение в корзину**: Когда вы удаляете файл с помощью trash-put, он перемещается в специальную папку корзины (~/.local/share/Trash/files в Ubuntu). Этот процесс аналогичен тому, как удаление происходит через графический интерфейс.

**Восстановление и очистка**: Файлы, помещённые в корзину, можно восстанавливать, просматривая содержимое корзины и выбирая нужные файлы для восстановления. Если вы решите, что файл больше не нужен, вы можете очистить корзину, удалив все файлы окончательно.

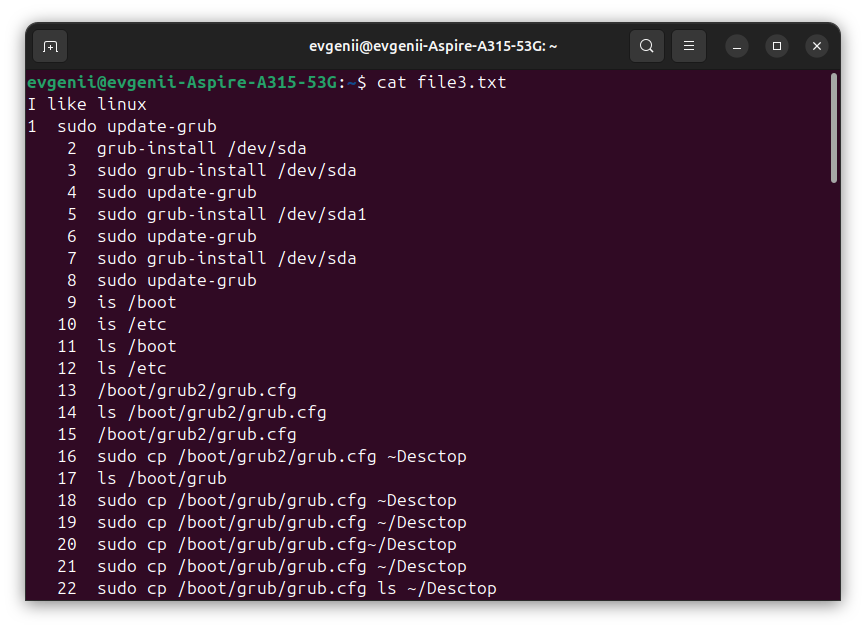
**Безопасность**: trash-cli предоставляет более безопасный способ удаления файлов, так как позволяет предотвратить случайное удаление важных данных и даёт возможность их восстановления.

Использование trash-cli делает процесс удаления файлов более контролируемым и безопасным, так как вы можете восстановить файлы из корзины, если удалили их случайно.

Записать текст «I like Linux!» в оставшийся файл.

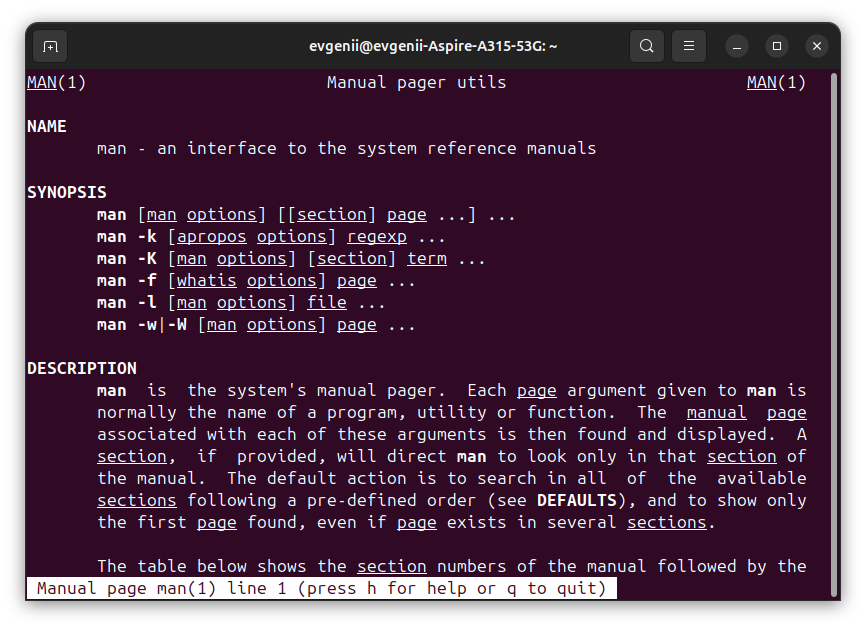


Допишите в этот файл историю команд. Вывести содержимое файла на консоль.



**Задание2**

Получите справку о справке. Укажите все разделы руководства.

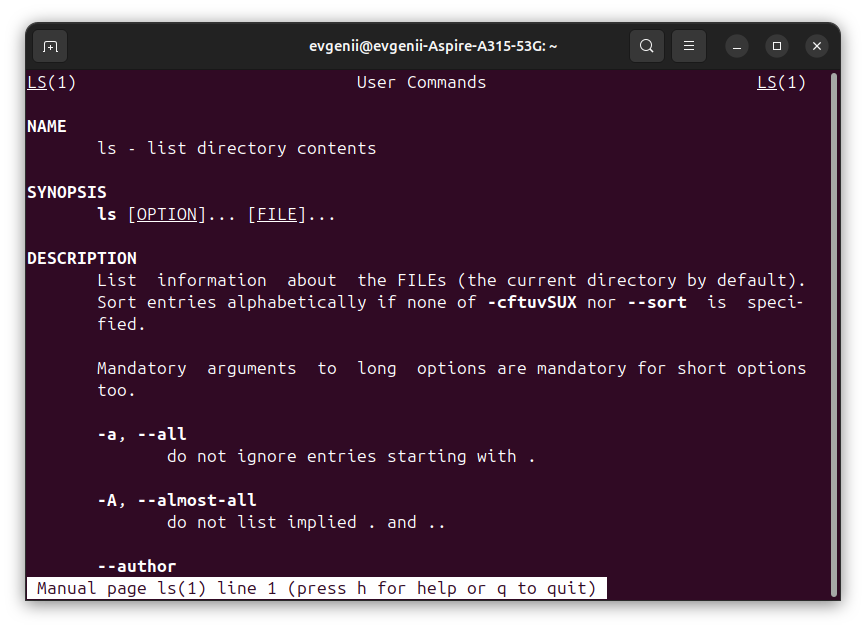


Справка о справке

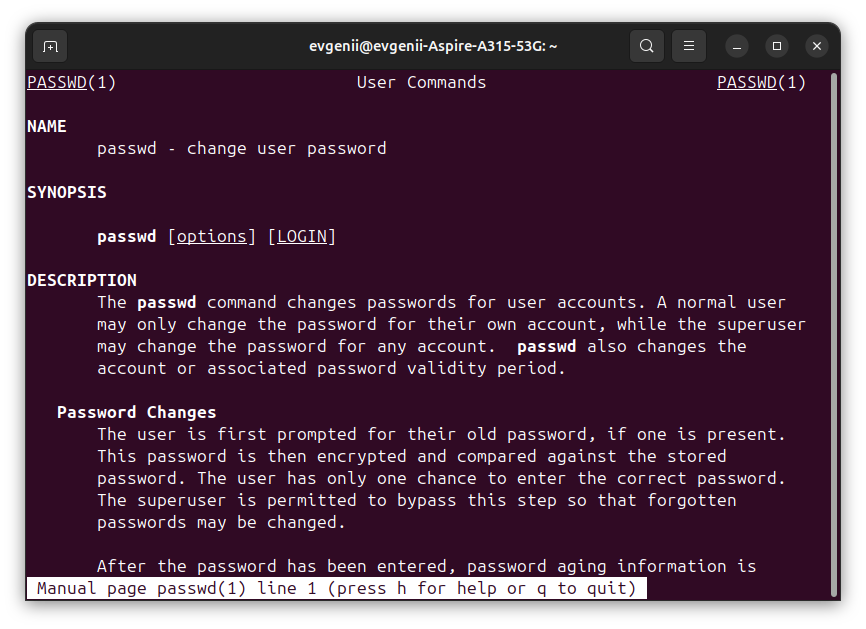
### Основные разделы команды man:

1. **NAME** – имя команды и краткое описание её назначения.
2. **SYNOPSIS** – общий синтаксис команды.
3. **DESCRIPTION** – более подробное описание команды и её опций.
4. **OPTIONS** – опции и аргументы, которые можно использовать с командой.
5. **FILES** – файлы, которые используются или создаются командой.
6. **SEE ALSO** – ссылки на другие связанные команды или документацию.
7. **BUGS** – информация о известных ошибках или проблемах с командой.
8. **AUTHORS** – информация о разработчиках команды или программы.

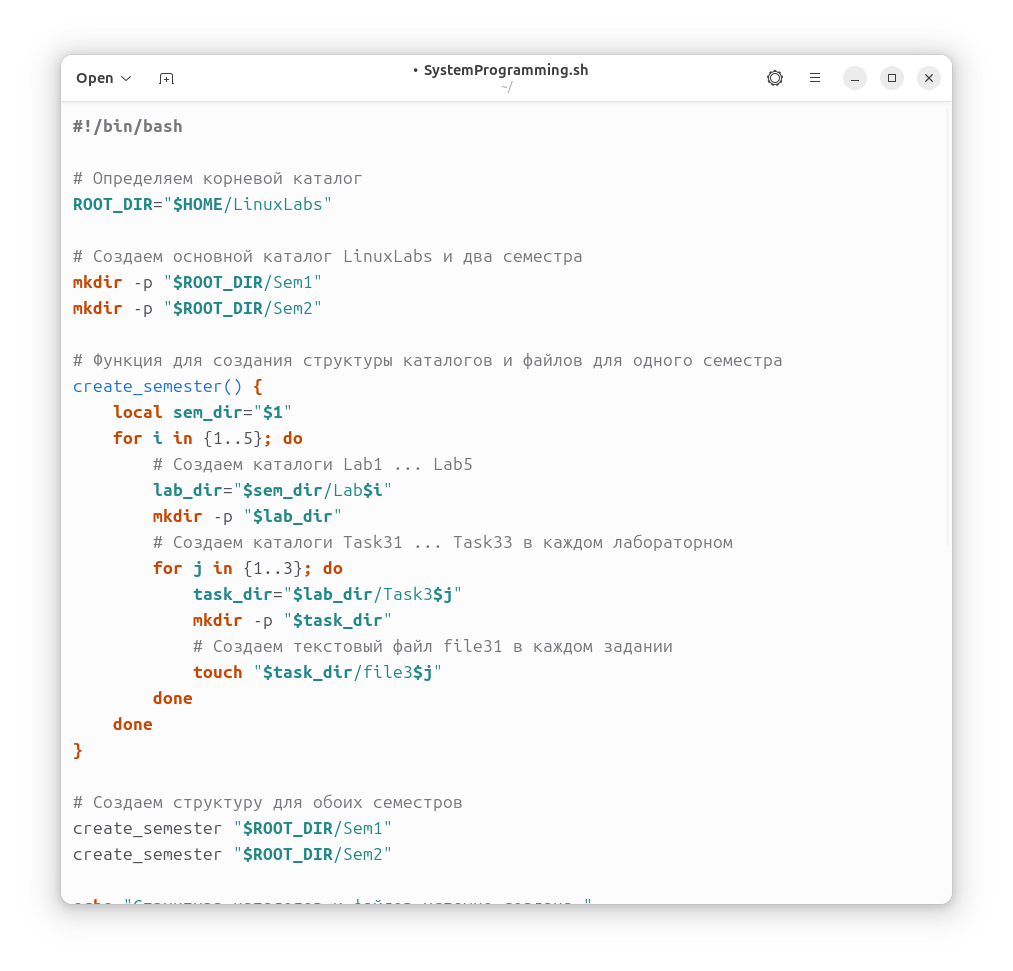
Получите краткую справку о любой команде, ранее использованной вами.



Справка об команде ls



Справка об команде passwd



#!/bin/bash

# Определяем корневой каталог

ROOT\_DIR="$HOME/LinuxLabs"

# Создаем основной каталог LinuxLabs и два семестра

mkdir -p "$ROOT\_DIR/Sem1"

mkdir -p "$ROOT\_DIR/Sem2"

# Функция для создания структуры каталогов и файлов для одного семестра

create\_semester() {

local sem\_dir="$1"

for i in {1..5}; do

# Создаем каталоги Lab1 ... Lab5

lab\_dir="$sem\_dir/Lab$i"

mkdir -p "$lab\_dir"

# Создаем каталоги Task31 ... Task33 в каждом лабораторном

for j in {1..3}; do

task\_dir="$lab\_dir/Task3$j"

mkdir -p "$task\_dir"

# Создаем текстовый файл file31 в каждом задании

touch "$task\_dir/file3$j"

done

done

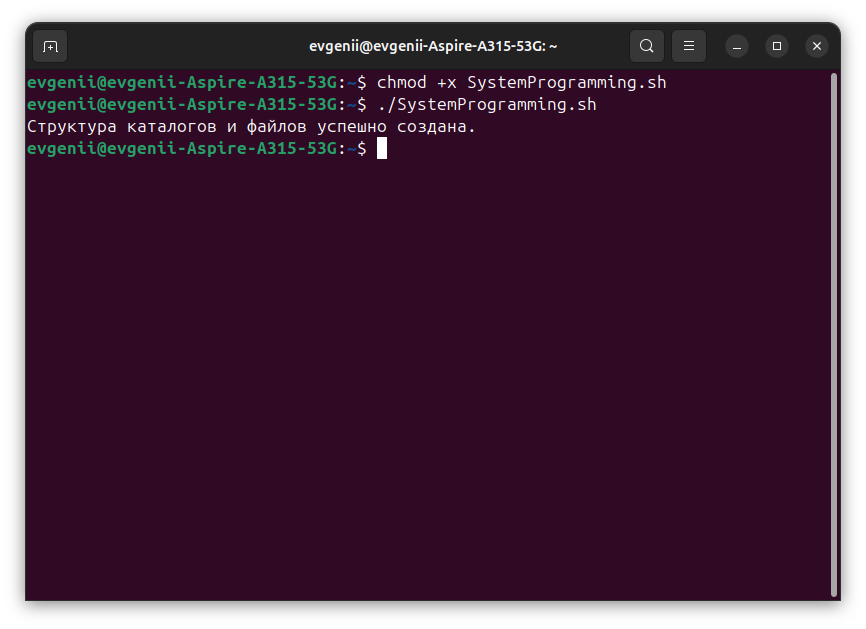
}

# Создаем структуру для обоих семестров

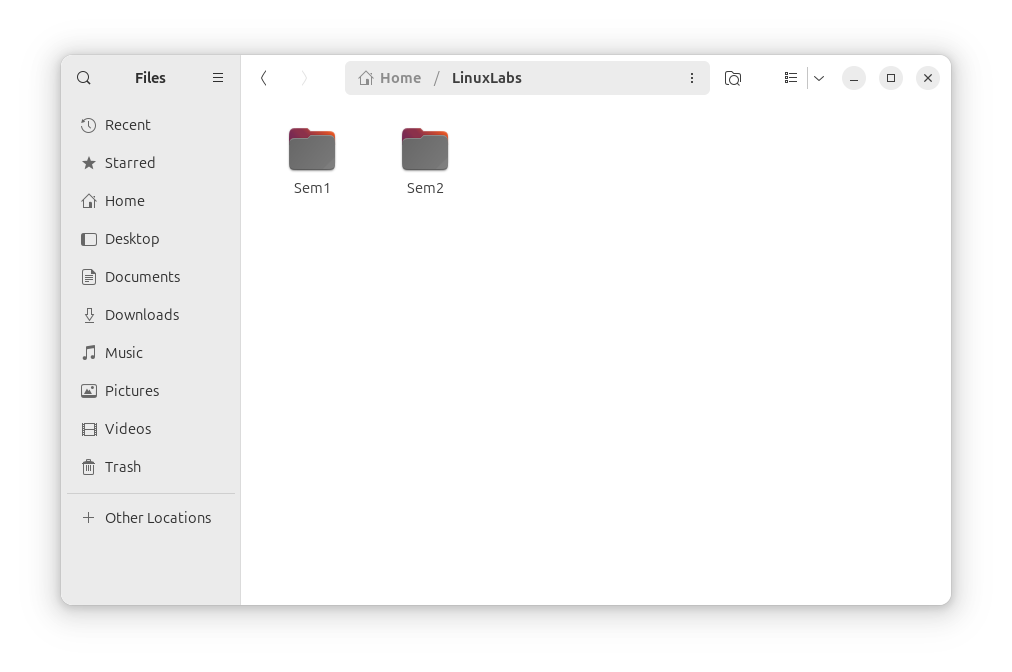
create\_semester "$ROOT\_DIR/Sem1"

create\_semester "$ROOT\_DIR/Sem2"

echo "Структура каталогов и файлов успешно создана."



Первая строчка делает файл со скриптом исполняемым, а вторая его запускает.



Видим, что все нужные дериктории были созданы корректно

**Ответы на вопросы(Медведский Е.В.)**

1)Как запустить консоль и затем вернуться в терминал (GUI) Ubuntu?

### 1. Открытие текстовой консоли (TTY):

В Ubuntu можно открыть консоль с помощью сочетания клавиш:

* **Ctrl + Alt + F3** до **Ctrl + Alt + F6** — это переключит вас на одну из виртуальных консолей (TTY), которая работает без графического интерфейса.

Каждая консоль работает независимо, и вы можете войти в систему, введя ваше имя пользователя и пароль.

### 2. Возвращение в графический интерфейс (GUI):

Чтобы вернуться обратно в GUI:

* **Ctrl + Alt + F2** — обычно графический интерфейс (где работает GNOME или другой рабочий стол) находится на TTY2 в последних версиях Ubuntu.

Если это не сработает, попробуйте **Ctrl + Alt + F1**.

### Итог:

* **Ctrl + Alt + F3-F6** — открыть текстовую консоль.
* **Ctrl + Alt + F2** — вернуться обратно в GUI на Ubuntu.

1. Почему в именах файлов нельзя использовать небуквенные символыи пробелы в Ubuntu?

На самом деле, в Ubuntu (и в большинстве UNIX-подобных систем) можно использовать небуквенные символы и пробелы в именах файлов. Однако есть несколько вещей, которые нужно учитывать при работе с такими именами:

* **Примеры разрешённых символов**: !, @, #, $, %, ^, &, \*, (, ), -, \_, +, =, {, }, [, ], |, ;, ', ", :, ,, ., /, ?, и ~.
* **Примеры запрещённых символов**: / (потому что он используется как разделитель директорий), и некоторые другие специфические символы в зависимости от файловой системы.

Пробелы в именах файлов допустимы, но при работе с ними в командной строке необходимо использовать специальные символы или экранирование, чтобы избежать путаницы:

3)Что такое встроенная справка утилиты, как ее получить в Ubuntu?

Встроенная справка утилиты — это краткое описание и список команд и опций, доступных для конкретной утилиты или команды в системе. Она предоставляет информацию о том, как использовать утилиту, какие параметры и опции поддерживаются, и часто включает примеры использования.

В Ubuntu, как и в других UNIX-подобных системах, есть несколько способов получить встроенную справку для утилит:

### 1. Опция --help

Многие утилиты поддерживают опцию --help, которая выводит справочную информацию:

command --help

Например, для получения справки по команде ls:

ls --help

1. **Опция -h**

Некоторые утилиты поддерживают опцию -h (или -?), которая также выводит справочную информацию:

command -h

**3. Команда man**

Команда man (от "manual") позволяет просматривать более подробную документацию для утилит. Вызовите её с именем команды, чтобы открыть её справочную страницу:

man command

Например, чтобы открыть справочную страницу для команды ls:

man ls

### 4. Команда info

Для некоторых утилит доступна информация в формате GNU Info. Вызовите команду info с именем утилиты:

**Получение справки по команде** grep:

grep --help

man grep

info grep

Ответы на вопросы (Зухта К.М.)

* 1. В какой директории находится командный интерпритатор вашей эко-системы?

Ответ: Командный интерпритатор (bash) в Linux Ubuntu находится в директории /bin. Файл называется `bash`.

* 1. Как запустить терминал с помощью мыши из текущей папки?

Ответ: Открыть проводник и перейти в нужную папку. Щелкнуть правой кнопкой мыши на пустом месте. В контекстном меню выбрать “Открыть в терминале”

* 1. Что такое экранирование символов и имен файлов, и как это делается?

Ответ: Экранирование символов и имен файлов — это процесс защиты определённых символов от интерпретации оболочкой (shell), чтобы они воспринимались именно как текст, а не как специальные команды или символы. Это особенно важно в Linux и других Unix-подобных системах.

Когда нужно экранирование?

1. Специальные символы: Некоторые символы, такие как \*, ?, $, &, ;, и пробелы имеют особое значение в оболочке. Например, \* используется для подстановки всех файлов в директории.

2. Пробелы: Пробелы в именах файлов могут привести к ошибкам, так как оболочка воспринимает пробелы как разделители.

Как это сделать?

1. С помощью обратной косой черты (`\`):

- Добавьте \ перед специальным символом или пробелом.

- Пример: my\ file.txt для файла с пробелом в имени.

2. В одинарных кавычках (`'`):

- Используйте одинарные кавычки, чтобы защитить всю строку от интерпретации.

- Пример: 'my file.txt'.

* 1. Назовите основные разделы справочника man.

### Ответ: Основные разделы команды man:

1. **NAME** – имя команды и краткое описание её назначения.
2. **SYNOPSIS** – общий синтаксис команды.
3. **DESCRIPTION** – более подробное описание команды и её опций.
4. **OPTIONS** – опции и аргументы, которые можно использовать с командой.
5. **FILES** – файлы, которые используются или создаются командой.
6. **SEE ALSO** – ссылки на другие связанные команды или документацию.
7. **BUGS** – информация о известных ошибках или проблемах с командой.
8. **AUTHORS** – информация о разработчиках команды или программы.