МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение   
высшего образования

«КРЫМСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ им. В. И. ВЕРНАДСКОГО»

ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ

Кафедра компьютерной инженерии и моделирования

**ОТЧЕТ ПО ПРАКТИЧЕСКОМУ ЗАДАНИЮ №2**

# «Изучение базовых возможностей OC Linux»

Практическая работа

по дисциплине «Современные технологии программирования»

студента 1 курса группы ПИ-б-о-231(2)

Аметов Кемран Ленверович

направления подготовки 09.03.04 «Программная инженерия»

|  |  |  |
| --- | --- | --- |
|  |  |  |

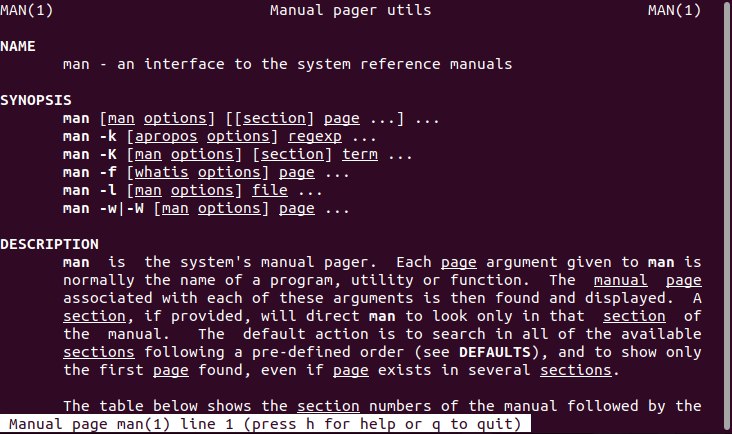
Симферополь, 2024

**Цель:** Освоить основные возможности консоли OC семейства Linux. Изучить базовый набор команд для работы с OC Ubuntu. Познакомиться с фундаментальными концепциями используемыми в ОС семейства Linux.

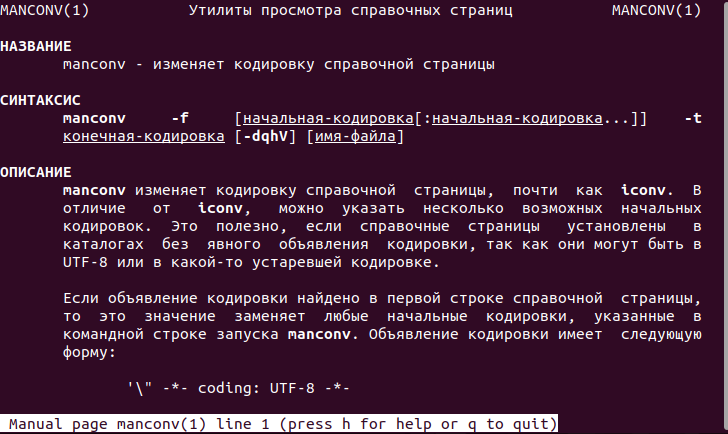
**Ход выполнения задания.**

# Справочная система Linux

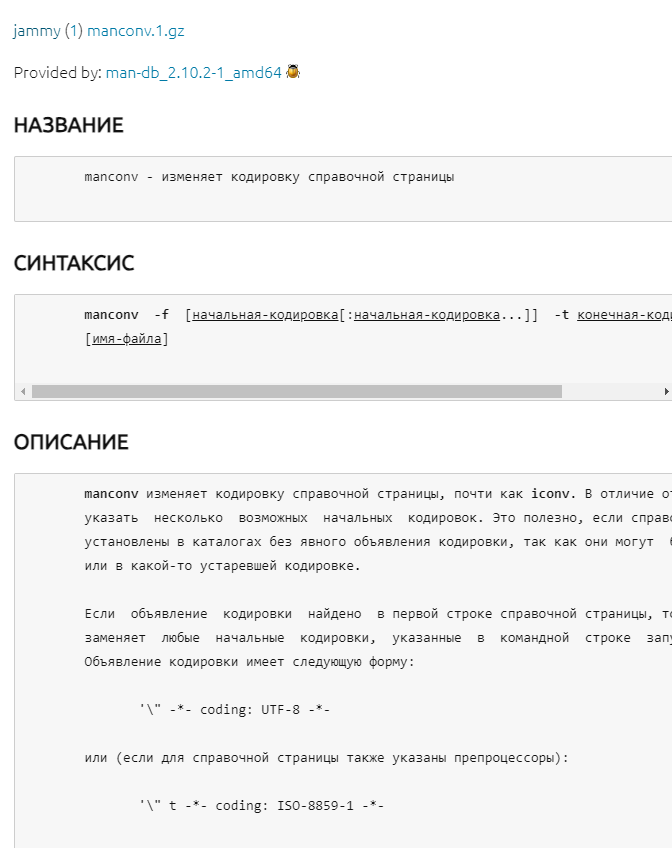
* 1. При помощи man изучите справочную информацию по команде man;



* 1. Найдите команду для которой доступен перевод на русский язык (любую кроме команды man) и запустите man для этой команды с явным указание языка;



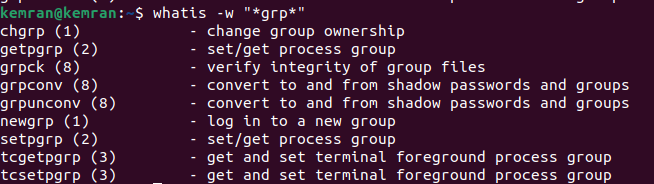
* 1. Информацию для этой же команды найдите на сайте <https://manpages.ubuntu.com/>.



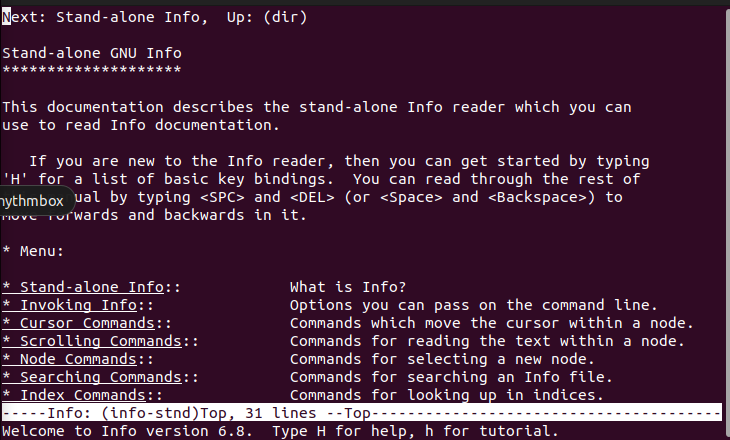
* 1. Воспользуйтесь командой apropos и найдите команду для добавления пользователя в группу. В процессе поиска вводить название команды нельзя, только описание;



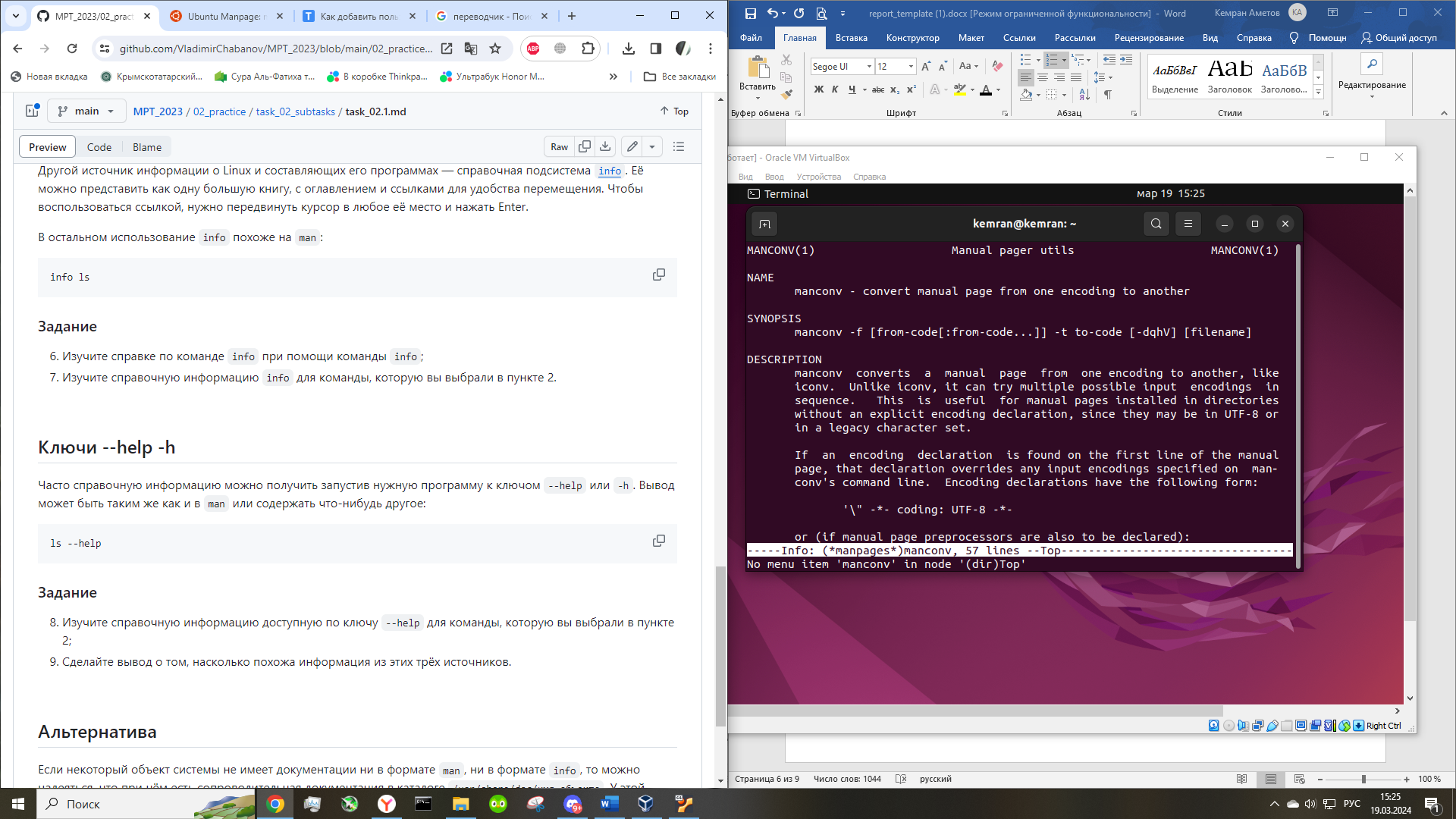
* 1. Воспользуйтесь командой whatis и найдите команды, название которых содержит grp в любом месте (в начале, середине, конце).



* 1. Изучите справке по команде info при помощи команды info;



* 1. Изучите справочную информацию info для команды, которую вы выбрали в пункте 2.



* 1. Изучите справочную информацию доступную по ключу --help для команды, которую вы выбрали в пункте 2;

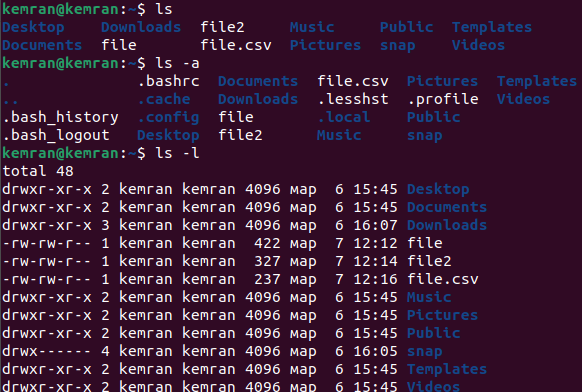


* 1. Сделайте вывод о том, насколько похожа информация из этих трёх источников.

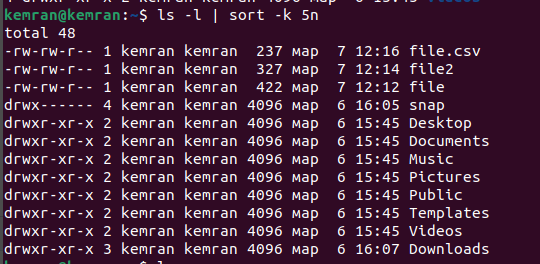
1. Файлы и утилиты для работы с ними
   1. Определите путь к вашему текущему каталогу при помощи команды pwd;



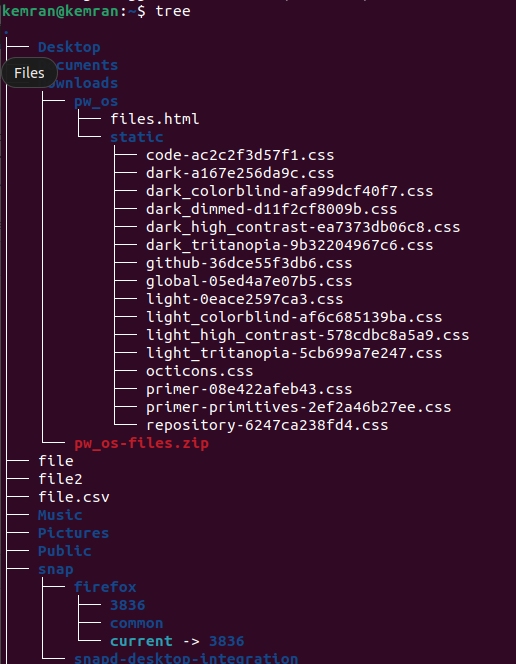
* 1. Посмотрите, какие файлы есть в этом каталоге. Сначала просто командой ls, затем с ключами -a и -l;



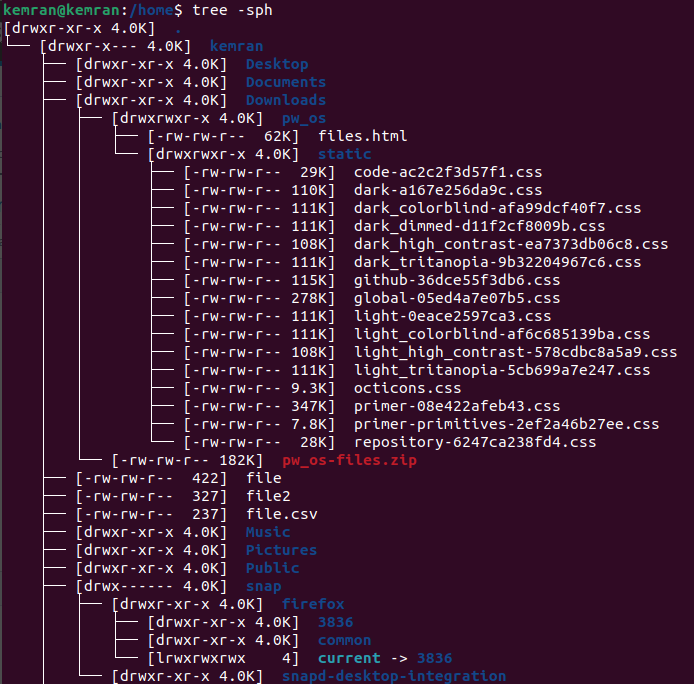
* 1. Выведете содержимое каталога, отсортированное по размеру файлов при помощи ls. Порядок сортировки: по возрастанию размера.



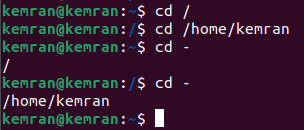
* 1. Посмотрите содержимое текущего каталога утилитой tree;



* 1. Посмотрите содержимое родительского каталога и добавьте ключи для просмотра всех файлов, их прав и размеров.



* 1. Перейдите в корневой каталог;
  2. Перейдите по пути /home/<имя вашего пользователя>;
  3. Вернитесь в предыдущий каталог при помощи -, а затем, таким же образом, обратно в домашний каталог пользователя.



* 1. В домашнем каталоге создайте каталог project. Не заходите в него;



* 1. Создайте иерархию каталогов project/app/items и project/app/site за один вызов mkdir



* 1. Используете команду touch, чтобы создать в каталоге ~/project/app/site файл empty.html прямо из домашнего каталога;
  2. Теперь перейдите в ~/project/app/site;



* 1. Находясь в ~/project/app/site используйте команду echo (4 раза) чтобы в каталоге ~/project/app создать файл app.py содержащий:

import socketserver

from http.server import SimpleHTTPRequestHandler

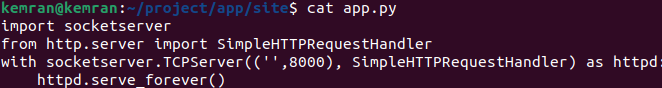
with socketserver.TCPServer(('', 8000), SimpleHTTPRequestHandler) as httpd:

httpd.serve\_forever()





* 1. Используйте команду cat, чтобы посмотреть содержимое только что созданного файла ~/project/app/app.py;



* 1. Используйте команду cat, чтобы создать файл index.html в каталоге ~/project/app/site с текстом:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Animals</title>

</head>

<body>

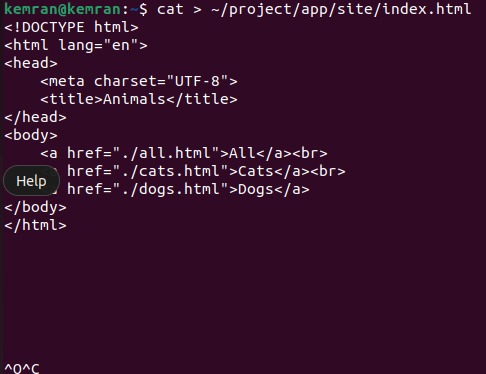
<a href="./all.html">All</a><br>

<a href="./cats.html">Cats</a><br>

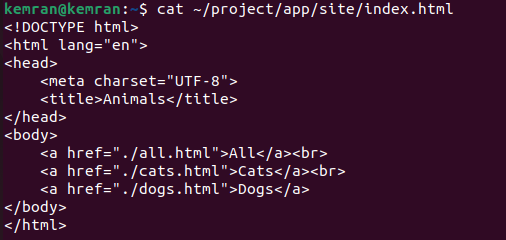
<a href="./dogs.html">Dogs</a>

</body>

</html>



* 1. Проверьте, что index.html содержит нужный текст при помощи команды cat;



* 1. Проверьте структуру каталога project командой tree. Если всё правильно, структура должна быть такой

project/

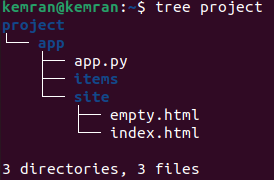
└── app

├── app.py

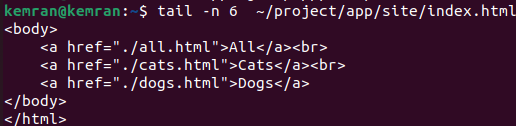
├── items

└── site

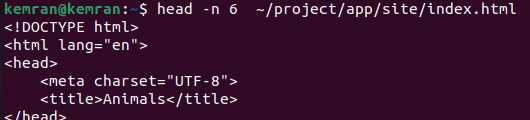
├── empty.html

 └── index.html

* 1. Используя head посмотрите первые 6 строк файла index.html;



* 1. Используя tail посмотрите последние 6 строк файла index.html;



* 1. В каталоге ~/project создайте каталог templates;



* 1. В нём создайте файл header cодержащий:

<!DOCTYPE html>

<html lang="en">

<head>

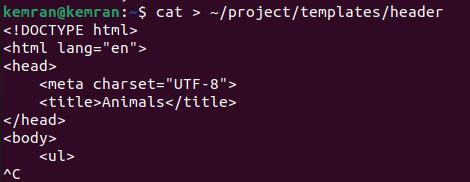
<meta charset="UTF-8">

<title>Animals</title>

</head>

<body>

<ul>

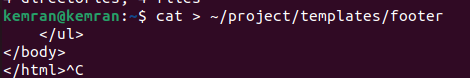


* 1. Там же создайте файл footer cодержащий:

</ul>

</body>

</html>



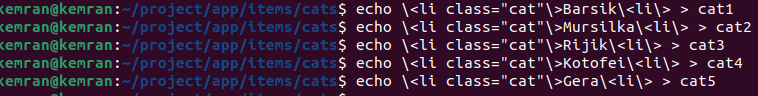
* 1. В каталоге ~/project/app/items создайте два новых каталога, с именами: cats и dogs;



* 1. В каталоге cats создайте 5 файлов с названием cat{Номер} содержащих текст соответствующий шаблону:

<li class="cat">{Кличка}</li>

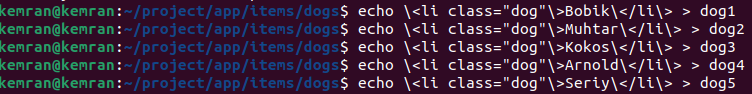
Вместо {Номер} укажите номер файла; вместо {Кличка} придумайте любую кличку для кота;



* 1. В каталоге dogs создайте 5 файлов с названием dog{Номер} содержащих текст соответствующий шаблону:

<li class="dog">{Кличка}</li>

Вместо {Номер} укажите номер файла; вместо {Кличка} придумайте любую кличку для собаки;



* 1. Перейдите в каталог ~/project/app/site и напишите команду find таким образом, чтобы она:

в каталоге items нашла все файлы (но не папки) с именами начинающимися на cat;

для каждого найденного файла выполнила команду cat имя\_файла >> cats;



* 1. Затем напишите команду find таким образом, чтобы она:

в каталоге items нашла все файлы (но не папки) с именами начинающимися на dog;

для каждого найденного файла выполнила команду cat имя\_файла >> dogs;



* 1. Затем напишите команду find таким образом, чтобы она:

в каталоге items нашла все файлы (но не папки);

для каждого найденного файла выполнила команду cat имя\_файла >> all.



* 1. Используйте команду cat, чтобы склеить файлы header, cats и footer в указанном порядке и сохранить результат в cats.html;



* 1. Повторите предыдущий пункт для файлов dogs и all.



* 1. Перейдите в каталог ~/project/app/items и создайте в нем каталог с именем tigers;



* 1. Скопируете все файлы из каталога cats в tigers;



* 1. Переименуйте файлы в каталоге tigers, таким образом, чтобы они начинались не на cat, а на tiger.
  2. Скопируйте каталог dogs вместе с файлами таким образом, чтобы копия называлась lupus.
  3. Переименуйте файлы в каталоге lupus, таким образом, чтобы они начинались не на dog, а на lupus.
  4. Перейдите в каталог ~/project;
  5. Используйте grep, чтобы найти все файлы, в тексте которых присутствует слово class. Добавьте ключ -l, чтоб в вывод покали только имена файлов. Вывод направьте в файл animals;
  6. Используйте файл animals и связку cat с grep, чтобы вывести имена всех файлов, которые не заканчиваются на .html. Вывод направьте в файл trash;
  7. Добавьте в файл trash каталоги app/items и templates, а также файлы animals и trash;
  8. Перед выполнением следующего пункта рекомендуется сделать копию каталога project, чтобы восстановить его, в случае ошибки;
  9. Воспользуйтесь командой rm, чтобы удалить все файлы и папки из списка. Для этого добавьте ключ -r (чтобы удалять папки) и в качестве аргумента команды напишите cat trash окружив его апострофами (это приведёт к тому, что вывод команды cat будет подставлен в команду rm как список аргументов).

Если вы все сделали правильно, то структура каталога project будет такая:

project/

└── app

├── app.py

└── site

├── all.html

├── cats.html

├── dogs.html

├── empty.html

└── index.html

* 1. Находясь в каталоге project/app/site запустите скрипт app.py командой:

python3 ../app.py

* 1. Перейдите в браузер на хостовой машине и введите в строку адреса IP виртуальной машины и, через двоеточие, порт 8000. В результате должна открыться страница с тремя ссылками: All, Cats, Dogs, переходя по которым должны отображаться списки кличек животных, которые вы добавляли ранее;
  2. Остановите приложение Ctrl + C.
  3. Используя find найдите все файлы в директории /usr/share/doc и к каждому найденному файлу примените утилиту file.
  4. Результат передайте утилите grep которая должна найти в выводе текст HTML document. То что получится сохраните в файл docs;
  5. Используйте утилиту nl (number lines), чтобы определить сколько строк в файле.
  6. Используйте связку cat и less чтобы посмотреть содержимое файла docs;
  7. Используйте функцию поиска в less чтобы найти в файле слово examples.
  8. Изучите справочную систему nano и vimtutor.

**Ответы на вопросы.**

1. Чем отличается вывод команд ls -F и ls -la?

Команда ls -F отображает содержимое текущего каталога, добавляя к именам файлов и каталогов специальные символы, указывающие на их тип (например, "/" для каталогов, "\*" для исполняемых файлов и т. д.). Это позволяет быстро определить тип каждого элемента.

Команда ls -la также выводит содержимое текущего каталога, но включает в вывод дополнительную информацию о каждом элементе, такую как права доступа, владелец, группа, размер, дата последнего изменения и имя файла. Также эта команда включает скрытые файлы и каталоги, которые начинаются с точки.

1. С помощью какой команды и как можно переместить файл в другой каталог?

Для перемещения файла в другой каталог можно использовать команду mv (move). Синтаксис команды выглядит следующим образом:

mv [путь к файлу] [путь к новому каталогу]

Например, чтобы переместить файл с именем "file.txt" из текущего каталога в каталог "new\_directory", нужно выполнить следующую команду:

mv file.txt new\_directory/

После выполнения этой команды файл "file.txt" будет перемещен в каталог "new\_directory".

1. Какие действия вы совершаете, нажимая на кнопки "стрелка вверх" и "стрелка вниз" в терминале?

Нажатие на кнопку "стрелка вверх" в терминале позволяет просматривать предыдущие команды, которые были введены ранее. Это удобно, если нужно быстро повторить или изменить команду, которую вы вводили ранее.

Нажатие на кнопку "стрелка вниз" позволяет просматривать последующие команды в истории команд терминала. Также это позволяет вернуться к более ранним командам после просмотра предыдущих команд с помощью стрелки вверх.

Таким образом, использование стрелок вверх и вниз в терминале упрощает работу с историей команд и повторное использование предыдущих команд.

1. Куда вы переходите, выполнив команду cd без параметров?

Выполнив команду cd без параметров, вы перейдете в ваш домашний каталог (home directory). Домашний каталог - это каталог, который является стартовым для каждого пользователя при входе в систему. Когда вы вводите команду cd без указания конкретного пути, терминал автоматически переключается в ваш домашний каталог.

1. Как осуществить вывод содержимого каталога с запятыми в качестве разделителя?

Для того чтобы осуществить вывод содержимого каталога с запятыми в качестве разделителя, можно воспользоваться командой ls с опцией -m. Эта опция позволяет выводить имена файлов и каталогов с запятыми в качестве разделителя. Например, команда ls -m выведет содержимое текущего каталога с именами файлов и каталогов, разделенными запятыми.

Пример использования:

ls -m

Таким образом, вы сможете увидеть содержимое каталога с запятыми в качестве разделителя.

1. Как осуществить создание нового каталога и необходимых подкаталогов рекурсивно?

Для создания нового каталога и необходимых подкаталогов рекурсивно в Linux можно воспользоваться командой mkdir с опцией -p. Эта опция позволяет создавать все указанные каталоги и подкаталоги, включая промежуточные каталоги, если они отсутствуют.

Пример использования:

mkdir -p новый\_каталог/подкаталог1/подкаталог2

В данном примере будет создан новый каталог "новый\_каталог" со всеми указанными подкаталогами. Опция -p гарантирует, что будут созданы все необходимые промежуточные каталоги.

Таким образом, вы сможете создать новый каталог и все необходимые подкаталоги рекурсивно с помощью команды mkdir и опции -p.

1. Как осуществить рекурсивное копирование всех файлов из одного каталога в другой?

Для рекурсивного копирования всех файлов из одного каталога в другой можно воспользоваться командой cp с опцией -r:

cp -r исходный\_каталог/ целевой\_каталог/

1. Как осуществить рекурсивное копирование всех файлов и подкаталогов из одного каталога в другой?

Для рекурсивного копирования всех файлов и подкаталогов из одного каталога в другой, также используйте команду cp с опцией -r:

cp -r исходный\_каталог/ целевой\_каталог/

1. Должен ли пользователь root вводить старый пароль при его смене с помощью команды passwd?

Пользователь root не должен вводить старый пароль при его смене с помощью команды passwd.

1. Что будет, если выполнить команду (от имени root): usermod -G "" user2?

Если выполнить команду (от имени root): usermod -G "" user2, то пользователь user2 будет удален из всех групп, кроме группы, к которой он принадлежал по умолчанию.

1. Можно ли назначить своей учётной записи права пользователя root. А затем удалить пользователя root из системы?

Нельзя назначить своей учетной записи права пользователя root и затем удалить пользователя root из системы, так как пользователь root имеет особые привилегии, которые нельзя передать другому пользователю.

1. Сколько времени по умолчанию команда sudo помнит введённый пароль?

По умолчанию команда sudo помнит введенный пароль в течение 15 минут.

1. Достаточно ли запретить доступ на запись в файл, чтобы предотвратить возможность удаления файла? Если нет, тогда как это сделать?

Запретить доступ на запись в файл не предотвратит возможность его удаления. Чтобы предотвратить удаление файла, необходимо установить соответствующие права на каталог, в котором находится файл. Например, установить sticky бит на каталог.

1. Как можно узнать права доступа к файлу/каталогу?

Чтобы узнать права доступа к файлу или каталогу, можно использовать команду ls с опцией -l:

ls -l файл\_или\_каталог

1. Какие права доступа нужно установить каталогу, чтобы все вновь созданные файлы и каталоги в нём получали группу этого каталога?

Чтобы все вновь созданные файлы и каталоги в каталоге получали группу этого каталога, необходимо установить setgid бит на каталог:

chmod g+s каталог/

**Вывод:**

Я освоил основные возможности консоли OC семейства Linux. Изучил базовый набор команд для работы с OC Ubuntu. Познакомился с фундаментальными концепциями используемыми в ОС семейства Linux.