

Навигация по файловой системе:

<code>pwd</code>	показать текущую директорию
<code>ls</code>	отобразить список файлов и папок
<code>cd</code>	сменить текущую папку

специальные символы для навигации по файловой системе

<code>~</code>	домашняя папка текущего пользователя (e.g., <code>/usr1/tutorial/tuta1</code>)
<code>.</code>	текущая директория
<code>..</code>	родительская директория
<code>*</code>	маска для любого имени файла или папки
<code>?</code>	маска для любого символа в имени айла или папки
<code>TAB</code>	попытка автодополнения

Примеры навигации по файловой системе

`cd /usr/local/lib` сменить директорию на `/usr/local/lib`

`cd ~` сменить директорию на домашнюю директорию текущего пользователя
(could also just type 'cd')

`pwd` показать рабочую (текущую) директорию

`cd ..` перейти в директорию на уровень вверх

`cd /` перейти в корневую директорию

`ls -d pro*` отобразить директории, имя которых начинается с `pro`

История команд

`history` - вся история

стрелки `↑` и `↓` клавиатуры для перехода к предыдущим командам

помощь по командам:

`date --help`

`man date`

`info date`

перенаправления через pipe

w #Показать всех пользователей,которые в данный момент вошли в систему

w | less # показать всех пользователей, которые зашли в формате страниц

w | grep 'тема' # выбрать всех пользователей, которые зашли, и показать только тех, в имени которых содержится «тема»

обработка текста

cat Отобразить содержимое файла

grep поиск по шаблону

tail вывести последние 10 строк файла

awk язык поиска по шаблонам

cut обрезать конкретные поля в каждой строке файла

diff сравнить два файла

head показать первую часть файла

less показать файл постранично

od сделать дампы файлов в различных форматах

sed потоковый редактор

sort отсортировать текстовые файлы

split разделить файлы

tail показать последнюю часть файла

tr транслитерировать символы

uniq отбросить повторяющиеся строки в файле

wc посчитать количество строк или слов или символов

отображение содержимого в файловой системе

ls -a показать все файлы, в том числе скрытые “.”

ls -ld * показать информацию о папке, а не о содержимом

ls -F поместить индикатор в конец кадого имени файла
ls -l простой длинный вывод
ls -lR рекурсивный длинный вывод(с учетом вывода содержимого подпапок)
ls -lh отображение размера файлов в читаемом виде
ls -lS отсортировать файлы по размеру
ls -lt отсортировать файлы по времени создания

Набор полезных команд

cp [file1] [file2] скопировать файлы
mkdir [name] создать папку
rmdir [name] удалить пустую папку(если не пустая, то не удалится)
mv [file] [destination] переместить(переименовать файл)
rm [file] удалить (-r рекурсивно)
file [file] идентифицировать тип файлы
less [file] вывести файл в постраничном виде
head -n [file] показать первые n строк файла
tail -n [file] показать последние n строк файла
ln -s [file] [new] создать символическую ссылку на файл
tac [file] вывести содержимое файла в обратном порядке, в сравнении с cat
touch [file] создать файл(если создан - изменить время создания)

Примеры работы с файлами и папками

cd (так же переместит в домашнюю директорию, как и cd ~)
mkdir test - создаст папку test
cd test - перейти в папку test
echo 'Hello everyone' > myfile.txt - вписать содержимое строки 'Hello everyone' в файл myfile.txt, заменив в нём всё содержимое
echo 'Goodbye all' >> myfile.txt - вписать содержимое строки Goodbye all, дописав его после последней строки
less myfile.txt - вывести содержимое файла myfile.txt постранично

`mkdir subdir1/subdir2` - (создать подпапку `subdir2` в подпапке `subdir1`. В результате получится ошибка, так как `subdir1` ещё не существует

`mkdir -p subdir1/subdir2` - создать подпапку `subdir2` в подпапке `subdir1` рекурсивно. Сначала будет создана подпапка `subdir1`, затем `subdir2`

`mv myfile.txt test/subdir1/subdir2` - переместить файл `myfile.txt` в новую папку `test/subdir1/subdir2`, с сохранением имени файла (так как не указано новое имя)

`cd ..` - перейти в папку на уровень выше

`rmdir test (FAILS)` - удалить папку `test`. Если в папке `test` есть какой то файл, то в результате будет возвращена ошибка

`rm -Rv test (Succeeds)` - рекурсивно удалить папку `test`. удаление придёт успешно

Работа с редактором vi

Есть два основных режима - Normal и Insert. для перехода в режим insert нужно нажать `i` на клавиатуре, для возврата в режим Normal - нажать `ESC`

Normal – режим навигации по тексту и управление им (удаление строк, символов и проч). в этом режиме:

кнопки стрелок, или символы `h, j, k, l` для навигации по тексту

`p` - вставка текста

`x` - удаление символа над курсором

`dd` - удаление текущей строки

`:` - вход в режим ввода команд

Insert – режим для добавления текста (вход из режима normal нажатием кнопки `i`, выход - `ESC`)

Visual – режим выделения текста. (вход из режима normal нажатием кнопки `v`, выход - `ESC`)

`vimtutor` - встроенная обучалка по работе с vi