

File Manager™

Добро пожаловать в File Manager™! Эта утилита позволит вам совершать большое количество различных действий с файлами.

Общие принципы работы с консолью

Командная строка

В начале работы вас встречает командная строка:

```
[FileManager]$
```

В квадратных скобках указано название текущей директории, а цвет знака \$ отображает статус выполнения предыдущей команды: зелёный — успешно, красный — ошибка.

Команды, опции и аргументы

Для выполнения какого-либо действия требуется ввести команду и нажать клавишу Enter. Каждая команда состоит из собственно названия команды, опций и аргументов. Например, в команде

```
find -r . *.pdf
```

Есть опция `r`, а также аргументы `.` и `*.pdf`. Можно задавать несколько опций следующим образом:

```
copy -rf Test1/ Test2/
```

В этом примере опции две: `r` и `f`. При этом стоит упомянуть, что опции должны идти слитно непосредственно после названия команды. То есть следующие команды не сработают:

```
copy -r -f Test1/ Test2/  
copy -r Test1/ -f Test2/  
copy Test1/ Test2/ -rf
```

Если аргумент содержит пробелы, то его необходимо взять в кавычки `"` или `'`, чтобы избежать его восприятия как двух отдельных документов:

```
dir "New Folder"  
dir 'New Folder'
```

При этом каждая кавычка, которая является частью аргумента, должна находиться внутри кавычек другого типа:

```
show 'Operation "Super Secret"'
```

Таким образом можно даже выражать аргументы, содержащие одновременно оба типа кавычек:

```
touch text.txt "There are 'quotes1' and '"quotes2'"
```

В данном примере второй аргумент будет распознан как `There are 'quotes1' and "quotes2"`.

Ввод команд

Предусмотрено несколько действий, упрощающих ввод команды. Следующие клавиши имеют специальные действия:

Клавиша	Действие
Tab	Дополнение названия команды или аргумента (чаще всего пути), при повторном нажатии — вывод всех возможных вариантов команды или аргумента.
Стрелки влево или вправо	Перемещение по введённой команде влево или вправо.
Стрелки вверх или вниз	Перемещение по истории команд: вверх — более старые команды, вниз — более новые.
Home и End	Перемещение в начало (Home) или конец (End) введённой команды.
Backspace и Delete	Удаление символа перед (Backspace) или на (Delete) курсоре.
Enter	Выполнение команды.

Помимо этого, при введении команды серым цветом отображается подсказка, которая при нажатии Tab будет введена.

Список и описание команд

В этом разделе будут подробно описан синтаксис и механизм работы каждой из следующих команд:

Команда	Действие
help, man или ?	Справка
exit	Выход
drive	Список или выбор диска
cd	Смена текущей директории

Команда	Действие
<code>pwd</code>	Вывод пути текущей директории
<code>dir</code> или <code>ls</code>	Вывод содержимого директории
<code>find</code>	Поиск файлов и директорий по маске или выражению
<code>show</code> , <code>cat</code> , <code>more</code> или <code>less</code>	Вывод содержимого файла в выбранной кодировке
<code>copy</code> или <code>cp</code>	Копирование
<code>move</code> или <code>mv</code>	Перемещение
<code>delete</code> , <code>del</code> или <code>rm</code>	Удаление
<code>create</code> или <code>touch</code>	Создание текстового файла
<code>concat</code>	Конкатенация файлов
<code>diff</code>	Вывод различий двух файлов

help (man, ?)

Данная команда предназначена для просмотра справки.

Использование:

Синтаксис	Описание
<code>help</code>	Вывод общей справки
<code>help console</code>	Вывод справки по общей работе консоли
<code>help commands</code>	Вывод списка команд с их кратким описанием
<code>help <i>command</i></code>	Вывод справки по команде <i>command</i>

Также можно вместо `help` использовать его псевдонимы `man` и `?`.

exit

Данная команда предназначена для выхода из программы.

Использование:

Синтаксис	Описание
<code>exit</code>	Выход из программы

drive

Данная команда предназначена для просмотра списка дисков и выбора определённого диска.

Использование:

Синтаксис	Описание
drive	Вывод списка дисков на компьютере
drive <i>drive_name</i>	Установка <i>drive_name</i> как текущего диска (текущей директорией становится корень этого диска)

cd

Данная команда предназначена для смены текущей директории.

Использование:

Синтаксис	Описание
cd <i>directory</i>	Смена текущей директории на <i>directory</i>

pwd

Данная команда предназначена для просмотра пути к текущей директории.

Использование:

Синтаксис	Описание
pwd	Вывод пути к текущей директории

dir (ls)

Данная команда предназначена для просмотра содержимого директории.

Использование:

Синтаксис	Описание
dir	Вывод общей справки
dir <i>directory</i>	Вывод справки по команде <i>command</i>

Также можно вместо `dir` использовать его псевдоним `ls`.

find

Данная команда предназначена для поиска файлов или директорий по **маске** или **выражению**.

Примечание по терминологии

«**Маска**» — строка, предназначенная для поиска по названию **файла**.

Маска	Комментарий
*.txt	Будут найдены файлы с расширением .txt
*.doc?	Будут найдены все файлы, расширение которого начинается с .doc и вместо ? содержит какой-то символ (а так как ? находится в конце, его может и не быть). Например, .doc, .docx, .docs и так далее
sec??? .txt	Например, найдутся файлы secret.txt, second.txt и так далее

«**Выражение**» — строка, предназначенная для поиска по **пути**.

Выражение	Комментарий
/*/*	Найдёт все файлы или директории, находящиеся в каждой директории в корне.
C:\Windows\Users\username*\.pdf (для Windows) или /home/username/*/*.pdf (для Linux)	Найдёт все файлы с расширением .pdf, находящиеся в любой директории в домашней директории пользователя.

ВАЖНО! Ни маска, ни выражение **не является** регулярным выражением!

Использование:

Синтаксис	Описание
find <i>expression</i>	Поиск по выражению <i>expression</i> (в данном случае маска тоже работает как поиск в текущей директории)
find -r <i>pattern</i>	Поиск по маске <i>pattern</i> рекурсивно (то есть в директории и всех её поддиректориях) в текущей директории.
find <i>directory</i> <i>pattern</i>	Поиск по маске <i>pattern</i> в выбранной директории <i>directory</i>
find -r <i>directory</i> <i>pattern</i>	Поиск по маске <i>pattern</i> рекурсивно в выбранной директории <i>directory</i>

show (cat, more, less)

Данная команда предназначена для просмотра содержимого файла в выбранной кодировке.

Примечание

Кодировка по умолчанию, а также выбор кодировок зависит от системы. Выбор кодировок при этом может быть просмотрен с помощью двойного нажатия Tab.

Например, для Linux доступны кодировки utf-16, utf-16BE, utf-32, utf-32BE, us-ascii, iso-8859-1 и utf-8.

Использование:

Синтаксис	Описание
show <i>file</i>	Вывод содержимого файла <i>file</i> в кодировке по умолчанию (зависит от системы)
show <i>encoding</i> <i>file</i>	Вывод содержимого файла <i>file</i> в выбранной кодировке <i>encoding</i> (выбор кодировок зависит от системы)

Также можно вместо show использовать его псевдонимы cat, more и less.

copy (cp)

Данная команда предназначена для копирования файлов и директорий.

Использование:

Синтаксис	Описание
copy <i>src dest</i>	Копирование файла <i>src</i> в <i>dest</i>
copy -r <i>src dest</i>	Рекурсивное копирование директории <i>src</i> в <i>dest</i>
copy -r <i>src dest/</i>	Рекурсивное копирование директории <i>src</i> внутрь директории <i>dest</i>
copy -r <i>src/ dest/</i>	Рекурсивное копирование содержимого директории <i>src</i> внутрь директории <i>dest</i>
copy -rp <i>pattern src</i> <i>dest</i>	Рекурсивное копирование содержимого директории <i>src</i> по маске <i>pattern</i> внутрь директории <i>dest</i>
copy -f <i>src dest</i> , copy -rf <i>src dest</i> или copy -rfp <i>pattern src</i> <i>dest</i>	То же самое, что и без -f, но без подтверждения перезаписи файлов

Также можно вместо copy использовать его псевдоним cp.

move (mv)

Данная команда предназначена для перемещения файлов.

Использование:

Синтаксис	Описание
-----------	----------

Синтаксис	Описание
<code>move src dest</code>	Перемещение файла из <i>src</i> в <i>dest</i>
<code>move -f src dest</code>	Перемещение файла из <i>src</i> в <i>dest</i> без подтверждения перезаписи файлов

Также можно вместо `move` использовать его псевдоним `mv`.

delete (del, rm)

Данная команда предназначена для удаления файлов.

Использование:

Синтаксис	Описание
<code>delete file</code>	Удаление файла <i>file</i>

Также можно вместо `delete` использовать его псевдонимы `del` и `rm`.

create (touch)

Данная команда предназначена для создания простого текстового файла в выбранной кодировке.

Примечание

Кодировка по умолчанию, а также выбор кодировок зависит от системы. Выбор кодировок при этом может быть просмотрен с помощью двойного нажатия Tab.

Например, для Linux доступны кодировки `utf-16`, `utf-16BE`, `utf-32`, `utf-32BE`, `us-ascii`, `iso-8859-1` и `utf-8`.

Использование:

Синтаксис	Описание
<code>create file</code>	Создать пустой файл <i>file</i> в кодировке по умолчанию (зависит от системы)
<code>create file text</code>	Создать файл <i>file</i> с содержимым <i>text</i> в кодировке по умолчанию (зависит от системы)
<code>create file encoding text</code>	Создать файл <i>file</i> с содержимым <i>text</i> в кодировке <i>encoding</i> (выбор кодировок зависит от системы)

Также можно вместо `create` использовать его псевдоним `touch`.

concat

Данная команда предназначена для конкатенации двух и более файлов.

Использование:

Синтаксис	Описание
<code>concat file1 file2 [... fileN]</code>	Конкатенация файлов <i>file1</i> , <i>file2</i> и так далее

diff

Данная команда предназначена для просмотра различий в двух файлах.

Примечание

В `diff` один из файлов принимается за оригинал (*original*), а другой за изменённый (*new*). Строки, которые встречаются в оригинальном файле, но отсутствуют в изменённом, отображаются красным цветом и знаком -, а строки, которые появились в изменённом файле, отображаются зелёным цветом и знаком +.

Использование:

Синтаксис	Описание
<code>diff original new</code>	Вывод различий в двух файлах