

Операционные системы

Огородников Юрий Юрьевич
yogorodnikov@gmail.com

UNIX::find

- *find*: универсальная команда поиска;

UNIX::find

- *find*: универсальная команда поиска;
- Общий шаблон: *find* место_поиска ключ-свойство значение_свойства действие_над_ найденным

UNIX::find

- *find*: универсальная команда поиска;
- Общий шаблон: *find* место_поиска ключ-свойство значение_свойства действие_над_найденным
- Найти в папке folder файлы, которые были изменены за последние 10 дней:
`find folder/ -mtime -10`

UNIX::find

- *find*: универсальная команда поиска;
- Общий шаблон: *find* место_поиска ключ-свойство значение_свойства действие_над_найденным
- Найти в папке folder файлы, которые были изменены за последние 10 дней:
`find folder/ -mtime -10`
- Найти в текущем каталоге файлы, соответствующие шаблону *2*
`find . -name *2*`

UNIX::find

- *find*: универсальная команда поиска;
- Общий шаблон: *find* место_поиска ключ-свойство значение_свойства действие_над_ найденным
- Найти в папке *folder* файлы, которые были изменены за последние 10 дней:
`find folder/ -mtime -10`
- Найти в текущем каталоге файлы, соответствующие шаблону **2**
`find . -name *2*`
- Найти в текущем каталоге все файлы, оканчивающиеся на *txt*. Каждый найденный файл передается команде *cp* (файл подставляется вместо фигурных скобок). Команда *cp* копирует файл в каталог *Downloads*. Конец аргумента *exes* обозначается обратным слэшем и точкой с запятой.
`find . -name '*.txt' -exec cp "{}" ../Downloads/ \;`

UNIX::find

- *find*: универсальная команда поиска;
- Общий шаблон: *find* место_поиска ключ-свойство значение_свойства действие_над_ найденным
- Найти в папке folder файлы, которые были изменены за последние 10 дней:
`find folder/ -mtime -10`
- Найти в текущем каталоге файлы, соответствующие шаблону *2*
`find . -name *2*`
- Найти в текущем каталоге все файлы, оканчивающиеся на txt. Каждый найденный файл передается команде cp (файл подставляется вместо фигурных скобок). Команда cp копирует файл в каталог Downloads. Конец аргумента exes обозначается обратным слэшем и точкой с запятой.
`find . -name '*.txt' -exec cp "{}" ../Downloads/ \;`
- найти файлы с заданным именем, игнорируя регистр
`find /home -iname name.txt`

UNIX::find

- *find*: универсальная команда поиска;
- Общий шаблон: *find* место_поиска ключ-свойство значение_свойства действие_над_найденным
- Найти в папке folder файлы, которые были изменены за последние 10 дней:
`find folder/ -mtime -10`
- Найти в текущем каталоге файлы, соответствующие шаблону *2*
`find . -name *2*`
- Найти в текущем каталоге все файлы, оканчивающиеся на txt. Каждый найденный файл передается команде cp (файл подставляется вместо фигурных скобок). Команда cp копирует файл в каталог Downloads. Конец аргумента exes обозначается обратным слэшем и точкой с запятой.
`find . -name '*.txt' -exec cp "{}" ../Downloads/ \;`
- найти файлы с заданным именем, игнорируя регистр
`find /home -iname name.txt`
- найти директорию с заданным именем
`find / -type d -name Directory`

UNIX::find

- *find*: универсальная команда поиска;
- Общий шаблон: *find* место_поиска ключ-свойство значение_свойства действие_над_ найденным
- Найти в папке folder файлы, которые были изменены за последние 10 дней:
`find folder/ -mtime -10`
- Найти в текущем каталоге файлы, соответствующие шаблону *2*
`find . -name *2*`
- Найти в текущем каталоге все файлы, оканчивающиеся на txt. Каждый найденный файл передается команде cp (файл подставляется вместо фигурных скобок). Команда cp копирует файл в каталог Downloads. Конец аргумента exes обозначается обратным слэшем и точкой с запятой.
`find . -name '*.txt' -exec cp "{}" ../Downloads/ \;`
- найти файлы с заданным именем, игнорируя регистр
`find /home -iname name.txt`
- найти директорию с заданным именем
`find / -type d -name Directory`
- найти файлы с разрешением 777
`find . -type f -perm 0777`

UNIX::find

- найти файлы с разрешением 777
`find . -type f -perm 0777`
- найти файлы, не обладающие разрешением 777
`find / -type f ! -perm 777`

UNIX::find

- найти файлы с разрешением 777
`find . -type f -perm 0777`
- найти файлы, не обладающие разрешением 777
`find / -type f ! -perm 777`
- найти файлы, предназначенные только для чтения
`find / -perm /u=r`

UNIX::find

- найти файлы с разрешением 777
`find . -type f -perm 0777`
- найти файлы, не обладающие разрешением 777
`find / -type f ! -perm 777`
- найти файлы, предназначенные только для чтения
`find / -perm /u=r`
- найти только исполняемые файлы
`find / -perm /a=x`

UNIX::find

- найти файлы с разрешением 777
`find . -type f -perm 0777`
- найти файлы, не обладающие разрешением 777
`find / -type f ! -perm 777`
- найти файлы, предназначенные только для чтения
`find / -perm /u=r`
- найти только исполняемые файлы
`find / -perm /a=x`
- найти и удалить файлы с определенным именем
`find . -type f -name "word.txt" -exec rm -f`

UNIX::find

- найти файлы с разрешением 777
`find . -type f -perm 0777`
- найти файлы, не обладающие разрешением 777
`find / -type f ! -perm 777`
- найти файлы, предназначенные только для чтения
`find / -perm /u=r`
- найти только исполняемые файлы
`find / -perm /a=x`
- найти и удалить файлы с определенным именем
`find . -type f -name "word.txt" -exec rm -f`
- найти все пустые файлы
`find /tmp -type f -empty`

UNIX::задания

- 1 Найти ключ, ограничивающий глубину поиска, и написать скрипт, ищущий все файлы с заданным расширением в заданной папке с заданной глубиной (команда `find`).
- 2 Отсортируйте все строки файла `in.txt` в лексикографическом порядке и выведите первые 10 строк, а затем последние 10 строк.
- 3 Каждая строка файла `students.txt` содержит текст вида
Имя:число_баллов(без пробелов и иных слов) Выведите имена студентов, набравших не меньше 10 баллов
- 4 Каждая строка файла `students.txt` содержит текст вида
Имя:число_баллов(без пробелов и иных слов) Выведите имена студентов, набравших меньше 10 баллов
- 5 В файле `names.txt` содержатся имена в формате Фамилия пробел Имя пробел Отчество. Найдите все варианты людей, имя которых заканчивается на адя, фамилия - на ова, а отчество - на вич.
- 6 В файле `5.txt` содержатся строки вида: Фамилия пробел балл_за_первую_пятиминутку пробел балл_за_вторую_пятиминутку пробел балл_за_третью_пятиминутку пробел балл_за_четвертую_пятиминутку
Выведите имена студентов, кто в трех пятиминутках подряд набрал 4 или 5 баллов
- 7 Выведите все строки файла `users.txt`, содержащие в начале слово `bad` без учета регистра
- 8 Выведите все строки файла `users.txt`, содержащие в конце слово `bad` без учета регистра
- 9 Выведите числа от 1 до 10 всеми известными способами, по одному числу в строке