

# **УРОК 9. FIND, LOCATE, BASH SCRIPTS - LOOPS AND CYCLES**

|  |          |
|--|----------|
| <b>КОМАНДА FIND</b>                    | <b>2</b> |
| <b>КОМАНДА LOCATE</b>                  | <b>3</b> |
| <b>ИСПОЛЬЗОВАНИЕ FIND</b>              | <b>4</b> |
| <b>ИСПОЛЬЗОВАНИЕ LOCATE</b>            | <b>7</b> |
| <b>BASH SCRIPTS - LOOPS AND CYCLES</b> | <b>8</b> |
| <b>РАБОТА СО СКРИПТАМИ</b>             | <b>9</b> |



## КОМАНДА FIND



**Команда `find` предназначена для поиска файлов и каталогов в файловой системе.**

Ее основная особенность заключается в том, что она ищет файлы рекурсивно, начиная с указанного места (каталога).

Unset

```
find /path/to/search -name "*.txt"
```

Особенности команды `find`:

- `find` производит поиск в реальной файловой системе, что может занимать больше времени, особенно на больших дисках.
- Она может выполнять более сложные запросы с использованием различных критериев, таких как имя файла, тип, размер, права доступа и т. д.



## КОМАНДА LOCATE



**Команда locate также предназначена для поиска файлов, но она работает на основе базы данных, содержащей информацию о файлах и каталогах в системе.**

Это позволяет ей работать значительно быстрее, чем find, особенно при поиске файлов по имени.

Unset

```
locate *.txt
```

Особенности команды locate:

- locate использует предварительно построенную базу данных, что делает поиск значительно быстрее.
- Поскольку locate использует базу данных, ее результаты могут быть неактуальными в случае изменений в файловой системе после последнего обновления базы данных.
- Она ограничена поиском по имени файла, поэтому для более сложных запросов используется find.



## ИСПОЛЬЗОВАНИЕ FIND

### 1. Поиск файлов по имени или части имени:

Unset

```
find /path/to/search -name "file.txt"
```

```
find /path/to/search -name "*.jpg"
```

### 2. Поиск файлов по типу:

Unset

```
find /path/to/search -type f
```

```
find /path/to/search -type d
```

-type:

f – простые файлы;

d – каталоги;

l – символические ссылки;

b – блочные устройства (dev);

c – символьные устройства (dev);

p – именованные каналы;

s – сокеты;

### 3. Поиск файлов по размеру:



Unset

```
find /path/to/search -name "*.txt" -size +1M
```

Найдет все файлы с расширением ".txt" и размером больше 1 мегабайта.

#### 4. Поиск файлов по дате изменений:

Unset

```
find /path/to/search -mtime -5
```

-mtime +N: Найти файлы, которые были модифицированы более чем N дней назад.

-mtime -N: Найти файлы, которые были модифицированы менее чем N дней назад.

-mtime N: Найти файлы, которые были модифицированы точно N дней назад.

Unset

```
find /home/ -mtime -5 -name "*.txt" -size +15k 2>/dev/null
```

find будет искать файлы внутри каталога /home/ и его подкаталогов, которые были изменены менее 5 дней назад, имеют расширение .txt и размер больше 15 килобайт. Любые ошибки, возникающие при поиске, будут проигнорированы и не отображены на экране.

#### 5. Удаление найденных файлов:



Unset

```
find -type f -name "*.old" -delete
```

Удаление файлов с расширением .old .

Unset

```
find /path/to/search -type f -mtime +7 -delete
```

Найдет все обычные файлы в указанном каталоге и его подкаталогах, которые были модифицированы более 7 дней назад, и удалит их.

6. Выполнение действий с найденными файлами:

Unset

```
find /path/to/search -type f -exec chmod 644 {} \;
```

Найдет все обычные файлы в указанном каталоге и его подкаталогах, а затем установит права доступа 644 для каждого найденного файла.

Unset

```
find /root/ -type f -name "*.txt" -exec cp {} /home \;
```

find найдет все файлы с расширением .txt внутри каталога /root/ и его подкаталогов, а затем скопирует каждый найденный файл в каталог /home.



## ИСПОЛЬЗОВАНИЕ LOCATE

Перед использованием locate рекомендуется выполнить команду `sudo updatedb` для обновления базы данных. Она перестроит базу данных, чтобы она содержала актуальную информацию о файлах в системе.

1. Поиск файла по имени или расширению:

Unset

```
locate myfile.txt  
locate *.jpg
```

2. Поиск файлов в определенном каталоге:

Unset

```
locate -r '/path/to/directory/*\.txt$'
```



## BASH SCRIPTS - LOOPS AND CYCLES

В Bash циклы позволяют выполнять один и тот же блок кода несколько раз, в зависимости от условий или значений переменных.

### 1. Циклы for:



**Цикл for выполняет блок кода для каждого элемента в заданном списке или диапазоне.**

```
Unset
for i in {1..5}
do
    echo "Number: $i"
done
```

### 2. Циклы while:



**Цикл while выполняет блок кода, пока заданное условие верно.**

```
Unset
counter=1
while [ $counter -le 5 ]
do
    echo "Counter: $counter"
    ((counter++))
done
```

Бесконечный цикл - однострочник:

```
Unset
while true; do echo "Alive" && sleep 1;done
```





## РАБОТА СО СКРИПТАМИ

Разберем скрипт создания директорий и копирования файлов в них: скрипт создает текстовые файлы, содержащие текущее время, копирует их в каталог с названием, соответствующим текущей дате, а затем выводит содержимое этих файлов на экран.

```
Unset
#!/bin/bash
#
DATE=`date +%d-%m-%y`
#echo $DATE
for i in {1..5}
do
date +%H-%M-%S' > File-$i.txt
sleep 5
done
mkdir -p $DATE
cp File*.txt $DATE
#cd $DATE
for FILE in $DATE/*
do
cat $FILE
done
```

- Устанавливает переменную DATE для текущей даты в формате "день-месяц-год".
- В цикле for создает 5 текстовых файлов с названиями вида "File-N.txt", где N - номер файла в диапазоне от 1 до 5. Каждый файл содержит текущее время в формате "час-минута-секунда".
- После создания каждого файла скрипт ждет 5 секунд с помощью команды sleep.



- Создает каталог с именем, соответствующим текущей дате, используя переменную DATE.
- Копирует все файлы с именем "File\*.txt" в созданный каталог, используя команду cp.
- В цикле for для каждого файла в каталоге с текущей датой считывает содержимое каждого файла и выводит его содержимое на экран с помощью команды cat.