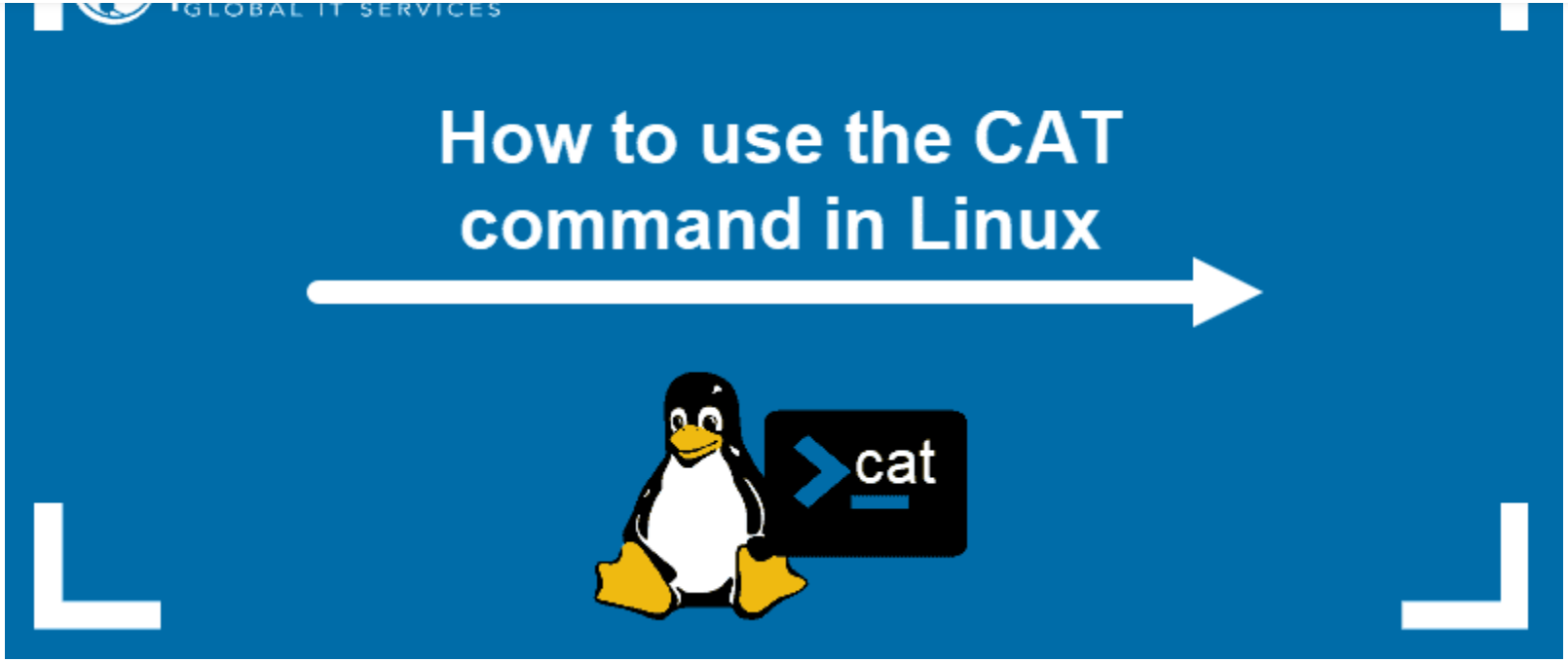
How To Use The Linux Cat Command (With Examples)

Posted July 13, 2020 LINUX COMMAND LINE COMMANDS

Introduction

If you have worked in Linux, you surely have seen a code snippet that uses the `cat` command. Cat is short for concatenate. This command displays the contents of one or more files without having to open the file for editing.

In this article, learn how to use the `cat` command in Linux.



Prerequisites

- A system running Linux
- Access to a terminal window / command line

cat Command Syntax

```
cat [options] filename(s)
```

`[options]` – This lets you issue additional instructions to the `cat` command. For example, to display the contents of a file with each line numbered, use the `–n` option:

```
cat –n filename
```

`filename(s)` – Specify the name of the file (or files) that you want to display. If you use more than one filename, each file will be displayed.

Linux Cat Command Examples

Create two sample files with some content. Open a terminal window, and enter the following:

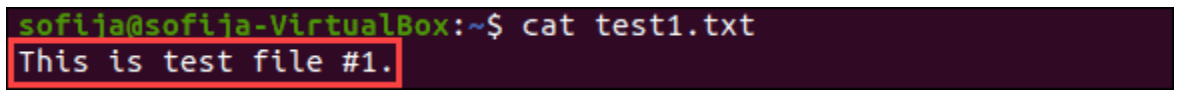
```
echo “This is test file #1.” > test1.txt
```

Display Contents of File

To display the contents of test1.txt using the `cat` command:

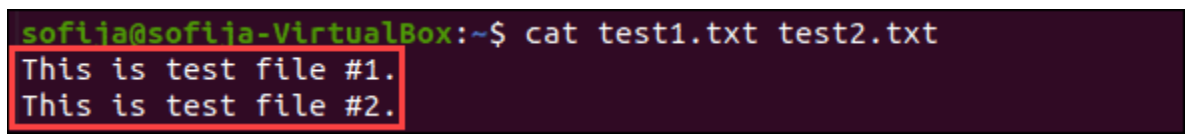
```
cat test1.txt
```

The output displays the content as in the image below:



Display the contents of both files:

```
cat test1.txt test2.txt
```



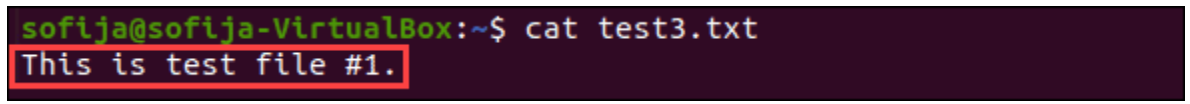
Instead of displaying the contents of a file on the screen, `cat` can put them in a file.

```
cat test1.txt > test3.txt
```

If the destination filename doesn't exist, it will be created. If you run `cat` on **test3.txt**, you should see the contents from **test1.txt**:

```
cat test3.txt
```

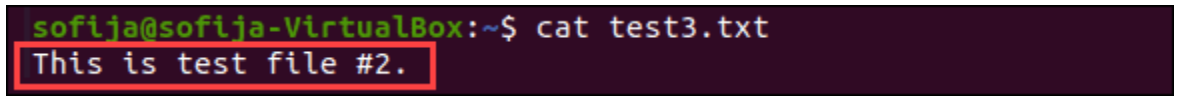
The output displays:



If a file is exported that already exists, this will **overwrite the contents of the file**:

```
cat test2.txt > test3.txt
```

The test3.txt file now has the following content:



Append File Contents to Another File

The `cat` command can add the contents of a file to the end of another file. Instead of using a single `>` sign, use a double `>>` sign:

```
cat test1.txt >> test3.txt
```

Open the **test3** file by running:

```
cat test3.txt
```

The content of **test3** followed by **test1** should display.

Note: If you want to remove the sample files, take a look at how to remove files and directories using the Linux command line.

Combining Operations

The functions of the `cat` command can be combined. For example, to combine the output of two files, and store the result in a new file:

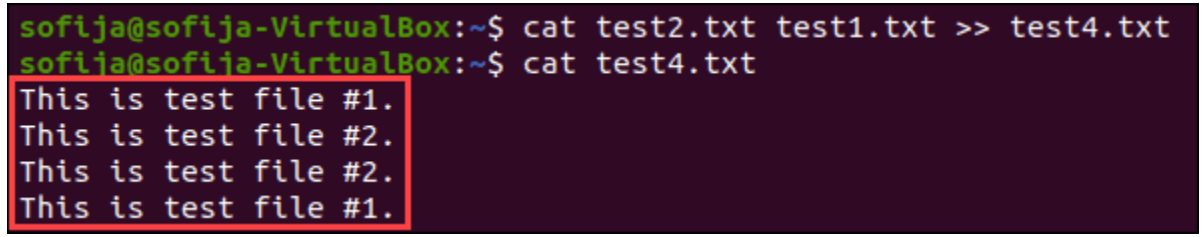
```
cat test1.txt test2.txt > test4.txt
```

```
cat test4.txt
```

Alternately, you can append multiple files to the end of an existing file:

```
cat test2.txt test1.txt >> test4.txt
```

```
cat test4.txt
```



Note that the order specified is the order the files in which they are added to the destination file.

Open a Prompt to Type Text Into a New File

1. Open the prompt for a new **test5** file with the command:

```
cat > test5.txt
```

2. Then, type:

```
This is some text I want to add to file #5.
```

3. Press **Enter** at the end of the first line. Then type:

```
This is a second line of text.
```

3. To exit the prompt and write the changes to the file, hold the **Ctrl** key and press `d`.

4. Open the file to check the content:

```
cat test5.txt
```

```
sofija@sofija-VirtualBox:~$ cat test5.txt
This is some text I want to add to file #5.
This is a second line of text.
```

You can use a similar command to append text to an existing file:

```
cat >> test5.txt
```

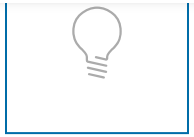
```
This is a third line of text.
```

Hold **Ctrl** and hit ***d*** .

Show the content of the **test5** file:

```
cat test5.txt
```

```
sofija@sofija-VirtualBox:~$ cat test5.txt
This is some text I want to add to file #5.
This is a second line of text.
This is a third line of text.
```



Note: Once you have created multiple files, you may want to group them in a single directory. Take a look at how to use mkdir command to make or create a Linux directory.

Managing Large Files

If you use `cat` on a very large file, you'll end up with a huge string of data that's hard to read. You can break it into pages using `| more`:

```
cat test5.txt | more
```

This displays a single page of the file. When you press a key, it will scroll to the next page.

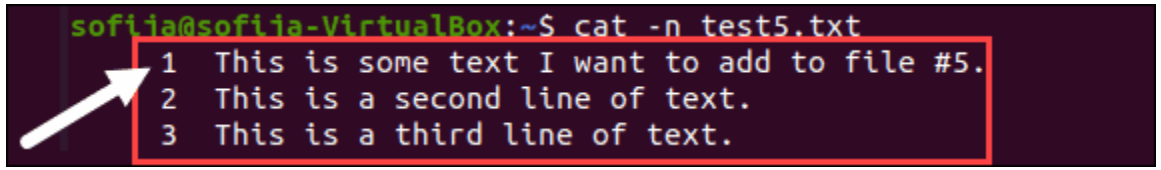
If you'd like the ability to scroll forward and backward through the display, use `| less`.

```
cat test5.txt | less
```

You may find it useful to have line numbers in the output, especially for large files. To enable line numbering, add the `-n` option to the `cat` command:

```
cat -n test5.txt
```

The output should appear as in the image below:



Other Options

To omit blank lines from the output of `cat` with the `-s` option:

```
cat -s test5.txt
```

To signify the end of each line (and the end of the file) with the `$` sign:

If you have trouble remembering the options, use the `--help` command:

```
cat --help
```

Conclusion

You should now have a good understanding of how to use the `cat` command in Linux.

Want to master more Linux commands? Check out our Linux Commands Cheat Sheet.

Next you should also read

Command In Linux/UNIX

March 28, 2019

This guide details the most useful grep commands for Linux / Unix systems. After going through all the…

READ MORE

tar.gz Files from Linux Command Line

November 14, 2019

This article shows which commands best to use when compressing and decompressing files from the command line.…

READ MORE

Linux Using Terminal/Command Line

June 27, 2019

Creating a file in Linux might seem straightforward, but there are some surprising and clever techniques. In…

READ MORE

Editors for Programming & Coding

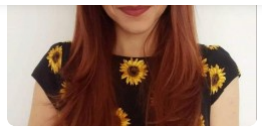
September 3, 2019

A text editor is an application that lets you type text. All Linux distributions come with built-in editors,…

READ MORE

Author

Sofija Simic



streamlining intricate software installations.