

[Shell Scripting](#) [Kali Linux](#) [Ubuntu](#) [Red Hat](#) [CentOS](#) [Docker in Linux](#) [Kubernetes in Linux](#) [Linux interview](#)

# How to Copy Files and Directories in Linux | cp Command

Last Updated : 12 Jul, 2024



In the world of Linux computers, a common and important job is copying files. The key tool for this task is the “cp” command. In this simple guide, we’ll explore how to copy a file in Linux, looking at the basic steps, different choices you can make, and giving clear examples. We’ll also take a peek behind the scenes to understand how the “cp” command works. Let’s get started on this easy journey to become familiar with the ins and outs of copying files in Linux!

## Table of Content

- [Syntax of cp Command](#)
- [How to Copy files in Linux with the cp Command](#)
- [1. Copying Between Two Files in Linux](#)
- [2. Copy files to a Directory in Linux](#)
- [3. How to Copy Directories in Linux](#)
- [Options Available in `cp` Command in Linux](#)

## Syntax of cp Command

The basic syntax for copying a file using the cp command is as follows:

```
cp source_file destination
```

This command creates a copy of the ``source_file`` at the specified ``destination``. If the destination is a [directory](#), the file is copied into that directory.

## How to Copy files in Linux with the cp Command

The ``cp`` command is a versatile tool used in Unix-like operating systems for copying files and directories. It offers three principal modes of operation, each

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

# 1. Copying Between Two Files in Linux

If the `cp` command contains two file names, it copies the contents of the first file to the second file. If the second file doesn't exist, it is created, and the content is copied into it. However, if the second file already exists, it is overwritten without warning.

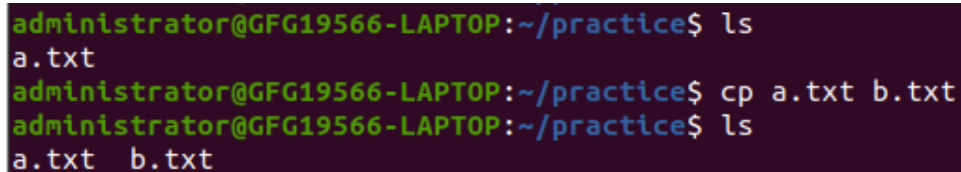
```
cp Src_file Dest_file
```

- If `Dest_file` does not exist, it is created.
- If `Dest_file` already exists, it is overwritten without any warning.

## Example 1:

- Initially, there is only one file (`a.txt`) in the directory.
- The `cp` command is used to copy the contents of `a.txt` to `b.txt`.
- After the command execution, both `a.txt` and the newly created `b.txt` coexist in the directory.

```
cp a.txt b.txt
```



```
administrator@GFG19566-LAPTOP:~/practice$ ls
a.txt
administrator@GFG19566-LAPTOP:~/practice$ cp a.txt b.txt
administrator@GFG19566-LAPTOP:~/practice$ ls
a.txt b.txt
```

*copy a file in Linux*

We used `ls` command to display all the file in the current directory.

## Example 2:

- Initially, there are two files (`a.txt` and `c.txt`) in the directory.
- The `cp` command is used to copy the contents of `a.txt` to `c.txt`.
- After the command execution, the content of `c.txt` is overwritten with the content of `a.txt`.

```
cp a.txt c.txt
```

```
administrator@GFG19566-LAPTOP:~/practice$ ls
a.txt  c.txt
administrator@GFG19566-LAPTOP:~/practice$ cat a.txt
Hello Welcome to GFG
administrator@GFG19566-LAPTOP:~/practice$ cat c.txt
administrator@GFG19566-LAPTOP:~/practice$ cp a.txt c.txt
administrator@GFG19566-LAPTOP:~/practice$ cat c.txt
Hello Welcome to GFG
administrator@GFG19566-LAPTOP:~/practice$
```

*Copy a file in Linux*

We used `ls` command to display all the file in the current directory and used `cat` command to display the content in the text file.

## 2. Copy files to a Directory in Linux

When the `cp` command has one or more source file arguments and is followed by a destination directory argument, it copies each source file to the destination directory with the same name. If the destination directory does not exist, it is created. If it already exists, the files are overwritten without warning.

```
cp Src_file1 Src_file2 Src_file3 Dest_directory
```

### Example:

Suppose we have to copy three files name “**a.txt**”, “**b.txt**” and “**c.txt**” to a directory name “**new**”

```
cp a.txt b.txt c.txt new/
```

```
administrator@GFG19566-LAPTOP:~/practice$ ls
a.txt  b.txt  c.txt  new
administrator@GFG19566-LAPTOP:~/practice$ ls new/
administrator@GFG19566-LAPTOP:~/practice$ cp a.txt b.txt c.txt new/
administrator@GFG19566-LAPTOP:~/practice$ ls new/
a.txt  b.txt  c.txt
```

*Copy multiple files to another directory*

We used `ls` command to display all the file in the “new” directory to confirm the successful copy of file in that directory.

## 3. How to Copy Directories in Linux

In this mode, if the `cp` command contains two directory names, it copies all files

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

```
cp -R Src_directory Dest_directory
```

```
administrator@GFG19566-LAPTOP:~/practice$ ls
a.txt b.txt c.txt new1
administrator@GFG19566-LAPTOP:~/practice$ cp -R /home/administrator/practice/ /home/administrator/Downloads/new3
administrator@GFG19566-LAPTOP:~/practice$ ls /home/administrator/Downloads/new3
a.txt b.txt c.txt new1
```

copying files between two directories

The behavior depends on whether `Dest_directory` exists or not. If it doesn't exist, `cp` creates it and copies the content of `Src_directory` recursively. If `Dest_directory` exists, the copy of `Src_directory` becomes a sub-directory under `Dest_directory`

### Options Available in cp Command in Linux

There are many options of `cp` command, here we will discuss some of the useful options:

| Option   | Detail                                                                                            |
|----------|---------------------------------------------------------------------------------------------------|
| -i       | Interactive copying with a warning before overwriting the destination file.                       |
| -b       | Creates a backup of the destination file in the same folder with a different name and format.     |
| -f       | Forces copying, even if the user lacks writing permission; deletes destination file if necessary. |
| -r or -R | Copies directory structure recursively.                                                           |
| -p       | Preserves file characteristics (modification time, access time,                                   |

| Option           | Detail                                                                         |
|------------------|--------------------------------------------------------------------------------|
| <code>`*`</code> | Uses the * wildcard to represent all files and directories matching a pattern. |

## 1. Copy a File in Linux Using `-i`` Option

**`-i(interactive)`:** `i` stands for Interactive copying. With this option the system first warns the user before overwriting the destination file. **`cp`** prompts for a response, if you press **`y`** then it overwrites the file and with any other option leaves it uncopied.

### Basic Syntax:

```
cp -i [Source_file] [Destination_file]
```

### Example:

```
cp -i a.txt b.txt
```

```
administrator@GFG19566-LAPTOP:~/practice$ ls
a.txt  b.txt
administrator@GFG19566-LAPTOP:~/practice$ cat a.txt
hello welcome to gfg
administrator@GFG19566-LAPTOP:~/practice$ cat b.txt
Hello this is b.txt file
administrator@GFG19566-LAPTOP:~/practice$ cp -i a.txt b.txt
cp: overwrite 'b.txt'? y
administrator@GFG19566-LAPTOP:~/practice$ cat b.txt
hello welcome to gfg
```

*Copy a File in Linux Using `-i``*

Here,

- ``ls`` command shows existing files: ``a.txt`` and ``b.txt``.
- ``cat a.txt`` displays the content of ``a.txt``.
- ``cat b.txt`` displays the content of ``b.txt``.
- ``cp -i a.txt b.txt`` initiates an interactive copy.
- System prompts to confirm overwrite of ``b.txt``.

## 2. Copy a File in Linux Using `-f` Option

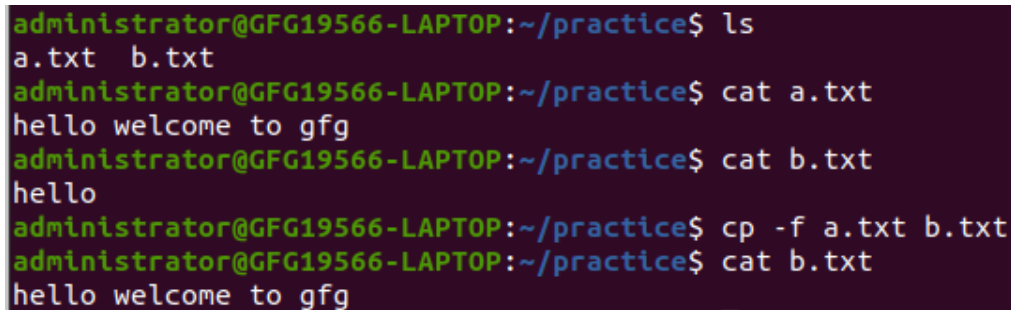
**-f(force):** If the system is unable to open destination file for writing operation because the user doesn't have writing permission for this file then by using **-f** option with **cp** command, destination file is deleted first and then copying of content is done from source to destination file.

### Basic Syntax:

```
cp -f [Source_file] [Destination_file]
```

### Example:

```
cp -f a.txt b.txt
```

A terminal window screenshot showing the execution of the cp -f command. The prompt is administrator@GFG19566-LAPTOP:~/practice\$. The first command is ls, which shows a.txt and b.txt. The second command is cat a.txt, which outputs 'hello welcome to gfg'. The third command is cat b.txt, which outputs 'hello'. The fourth command is cp -f a.txt b.txt. The fifth command is cat b.txt, which outputs 'hello welcome to gfg'.

*Copy a File in Linux Using `-f`*

Here,

- `ls` command shows existing files: `a.txt` and `b.txt`.
- `cat a.txt` displays the content of `a.txt`.
- `cat b.txt` displays the content of `b.txt`.
- `cp -f a.txt b.txt` initiates a forceful copy.
- Destination file (b.txt) is overwritten without prompting.
- `cat b.txt` shows the updated content, which now matches `a.txt`.

## 3. Copy a File in Linux Using `-r` or `-R` Option

Copying directory structure recursively. With this option **cp** command shows its recursive behavior by copying the entire directory structure recursively.

### Basic Syntax:

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

```
cp -r [Directory_name1] [Directory_name2]
```

**Example:**

```
cp -r geeksforgeeks gfg
```

#### 4. Copy a File in Linux Using `-p` Option

**-p(preserve):** With **-p** option **cp** preserves the following characteristics of each source file in the corresponding destination file: the time of the last data modification and the time of the last access, the ownership (only if it has permissions to do this), and the file permission-bits.

**Note:** For the preservation of characteristics, you must be the **root user** of the system, otherwise characteristics change.

**Basic Syntax:**

```
cp -p [Source_file] [Destination_file]
```

**Example:**

```
cp -p a.txt c.txt
```

#### 5. Copy a File in Linux Using `*` Option

**Copying using \* wildcard:** The star wildcard represents anything i.e., all files and directories. Suppose we have many texts documents in a directory and want to copy it to another directory, it takes lots of time if we copy files 1 by 1 or command becomes too long if specify all these file names as the argument, but by using \* wildcard it becomes simple.

**Basic Syntax:**

```
cp *.txt [Destination Directory or file]
```

**Example:**

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

```
cp *.txt Folder1
```

```
administrator@GFG19566-LAPTOP:~/practice$ ls
a.txt b.txt c.txt new
administrator@GFG19566-LAPTOP:~/practice$ cp *.txt new/
administrator@GFG19566-LAPTOP:~/practice$ ls new/
a.txt b.txt c.txt
```

*Copy a File in Linux Using '\*'*

## Conclusion

The `cp` command is an essential tool which is used for copying files or groups of files and directories in Unix-Like operating systems. If we talk about its syntax it takes at least two filenames in as an argument (source and destination). As mentioned, the command has three principles: copying two file names, copying one or more arguments, and copying two directory names. Then we also mention the multiple options available while using `cp` command: `-i`, `-b`, `-f`, `-r`, `-p`. To work with easy in Unix shell for file management one should know the proper working of `cp` command.

## How to Copy Files and Directories in Linux | cp Command – FAQs

### How Do You Copy Files and Directories in Linux?

*The cp command is used in Linux to copy files and directories from one location to another. Below are the steps and options commonly used with this command:*

#### **Copying Files**

*To copy a single file from one location to another, you can use:*

```
cp source_file_path destination_file_path
```

*For example:*

```
cp /home/user/file1.txt /home/user/backup/file1.txt
```

### **Copying Directories**

To copy a directory, including all its contents (files and subdirectories), you use the `-r` (recursive) option:

```
cp -r source_directory_path destination_directory_path
```

For example:

```
cp -r /home/user/documents /home/user/backup/documents
```

This copies the entire `documents` directory and its contents into the `backup` directory.

## **What Command is Used to Copy Files and Directories?**

The `cp` command is used to copy both files and directories, often with various options to tailor its functionality:

- `-r`: Recursive, for copying directories.
- `-a`: Archive, similar to `-r` and preserves the structure and attributes of files but is generally used for backing up.
- `-v`: Verbose, shows what `cp` is doing (useful for tracking what gets copied).

## **What is the `cp -R` Command in Linux?**

The `cp -R` command is an alternative to `cp -r`, and it functions the same way by copying directories recursively. Some systems differentiate slightly in how they handle symbolic links and special files between `-R` and `-r`, but for most users and typical usage, they behave the same.

*In the terminal, the `cp` command is used to copy files and directories:*

- **Syntax:** `cp [options] source destination`
- **Options:** Modify behavior (like `-r` for recursive copy).

## How Do I Copy a Directory?

*To copy a directory in Linux, use the `cp` command with the `-r` option. Here's how you might copy a directory from one place to another:*

```
cp -r /path/to/original_directory /path/to/copy_directory
```

*This command ensures that the entire directory, including all nested files and subdirectories, is copied to the new location.*



## Similar Reads

### Ways to Permanently and Securely Delete 'Files and Directories' in Linux

Linux is an open-source operating system that is loved by tech geeks. It is a versatile operating system that can be used on thousands of different hardware...

🕒 8 min read

### How to Copy a File to Multiple Directories in Linux

In this article, we will discuss how to copy a file to multiple directories in Linux. xargs, find, tee, shell loop, and parallel are the commands that can help you to...

🕒 5 min read

### UrlBuster - Linux tool to find Web Hidden Files or Directories Finder

Hidden files and directories on the target server can contain some sensitive and crucial information about the target. This revealed information can also...

🕒 5 min read

### How to Securely Copy Files in Linux | scp Command

scp (secure copy) command in Linux system is used to copy file(s) between servers in a secure way. The SCP command or secure copy allows the secure...

🕒 8 min read

### How to Exclude Files and Directories When Creating a tar.gz File

GNU tar is a type of archiving program that is designed to store multiple files into a single file and to manipulate such archives. The archive can be either a regular...

🕒 2 min read

### Systemd Services: Monitoring Files and Directories

It is very important to keep our services running smoothly in Linux system administration and management. Systemd is a versatile tool used in Linux that...

🕒 5 min read

sed stands for stream editor. It is a basic utility command in Linux that has basic functionality like inserting, deleting, finding, replacing, and searching text string...

🕒 4 min read

## How to Sync File and Directories to Cloud Storage in Linux Using Rclone Tool?

Rclone is a command-line tool for synchronizing files and directories between Google Drive, Amazon S3, Openstack Swift / Rackspace cloud files / Memset...

🕒 3 min read

## How to Recursively Grep all Directories and Subdirectories in Linux

In this article, we will demonstrate how to grep recursively through all directories and subdirectories. But before I do that, let me define what the term "grep"...

🕒 3 min read

## How to Transparently Overlaid Two Directories Using UnionFS in Linux?

To transparently overlay two directories one on top of the other using UnionFS following steps must be followed: Step 1: Enter into the superuser mode and...

🕒 1 min read

### Article Tags :

[Linux-Unix](#)[linux-command](#)[Linux-file-commands](#)



Sector- 136, Noida, Uttar Pradesh (201305)  
| Registered Address:- K 061, Tower K,  
Gulshan Vivante Apartment, Sector 137,  
Noida, Gautam Buddh Nagar, Uttar  
Pradesh, 201305



## Company

About Us  
Legal  
In Media  
Contact Us  
Advertise with us  
GFG Corporate Solution  
Placement Training Program  
GeeksforGeeks Community

## DSA

Data Structures  
Algorithms  
DSA for Beginners  
Basic DSA Problems  
DSA Roadmap  
Top 100 DSA Interview Problems  
DSA Roadmap by Sandeep Jain  
All Cheat Sheets

## Web Technologies

HTML  
CSS  
JavaScript  
TypeScript  
ReactJS  
NextJS  
Bootstrap  
Web Design

## Computer Science

## Languages

Python  
Java  
C++  
PHP  
GoLang  
SQL  
R Language  
Android Tutorial  
Tutorials Archive

## Data Science & ML

Data Science With Python  
Data Science For Beginner  
Machine Learning  
ML Maths  
Data Visualisation  
Pandas  
NumPy  
NLP  
Deep Learning

## Python Tutorial

Python Programming Examples  
Python Projects  
Python Tkinter  
Web Scraping  
OpenCV Tutorial  
Python Interview Question  
Django

## DevOps

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Software Engineering  
Digital Logic Design  
Engineering Maths  
Software Development  
Software Testing

Docker  
Kubernetes  
Azure  
GCP  
DevOps Roadmap

### System Design

High Level Design  
Low Level Design  
UML Diagrams  
Interview Guide  
Design Patterns  
OOAD  
System Design Bootcamp  
Interview Questions

### Interview Preparation

Competitive Programming  
Top DS or Algo for CP  
Company-Wise Recruitment Process  
Company-Wise Preparation  
Aptitude Preparation  
Puzzles

### School Subjects

Mathematics  
Physics  
Chemistry  
Biology  
Social Science  
English Grammar  
Commerce  
World GK

### GeeksforGeeks Videos

DSA  
Python  
Java  
C++  
Web Development  
Data Science  
CS Subjects

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved