

File Handling Commands

File handling is an essential aspect of programming and is used to manage files stored on a computer. Learn about these essential commands.

 by Thrinesh Dasari



Introduction to File Handling

1 What is file handling?

File handling is used to manage files that are stored on a computer.

3 What can you do with it?

You can use file handling to read, write, and modify files as well as manage them.

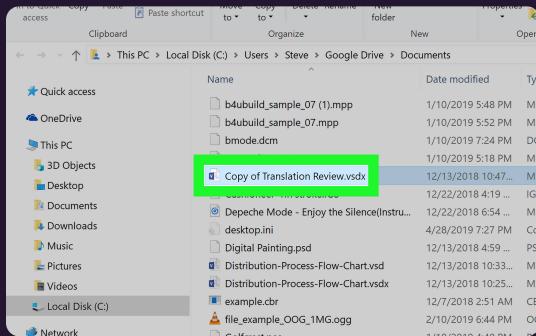
2 Why is it important?

File handling allows for the creation, manipulation, and storage of files on a computer system.

4 How does it work?

File handling commands allow for the opening, reading, writing, copying, renaming, and deleting of files in a computer system using programming languages.

Opening and Closing Files



How to open a file?

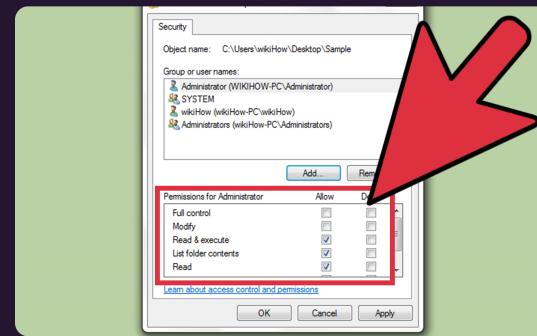
Use the "open" command. You can specify the mode as read, write, or append depending on what you intend to do with the file.

CLOSING a FILE

- Close files you are no longer working on
- Close a file by clicking the Office Button and then clicking the Close command
- As a standard practice, you should save your file before closing it

How to close a file?

To close a file, use the "close" command. This will release the resources used by the file and save any changes made to it.



Permissions and file access

When opening files, you need to specify the mode for permissions and access control. This allows you to control who can read, write, or execute the file.

Reading from a File

How to read a file?

Use the "read" command to read data from a file. You can specify the size of the data to read and where to start reading from.

Reading line by line

Read a line at a time using the "readline" command. This enables proper handling of large files that can't be read at once.

Reading all at once

If you have files that are small in size, you can also read the entire content at once using the "readlines" command.

Saving data to a variable

Use a variable to save the data, so that it can be used later in your program.

Writing to a File

Writing over existing file content

You can write over existing data in a file using the "write" command by specifying the position to start writing from.

1

How to write to a file?

Use the "write" command to write data to a file. You can specify the data to write and the position to start writing from.

2

Appending to a file

Use the "append" mode to add new data to the end of an existing file without overwriting existing data.

3

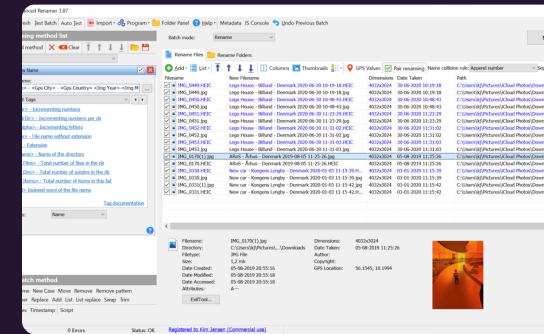


Copying and Renaming a File



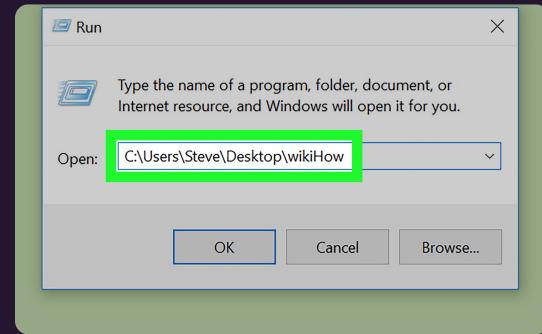
Copying a file in Python

To copy a file, import the `shutil` library and use the "copy" command. This will create an exact duplicate of the file.



Renaming a file in Python

You can rename a file in Python by using the "os" library and the "rename" command.



Specifying file paths

Remember to include the path for the file that you want to copy or rename.

Deleting a File

1 Deleting a file

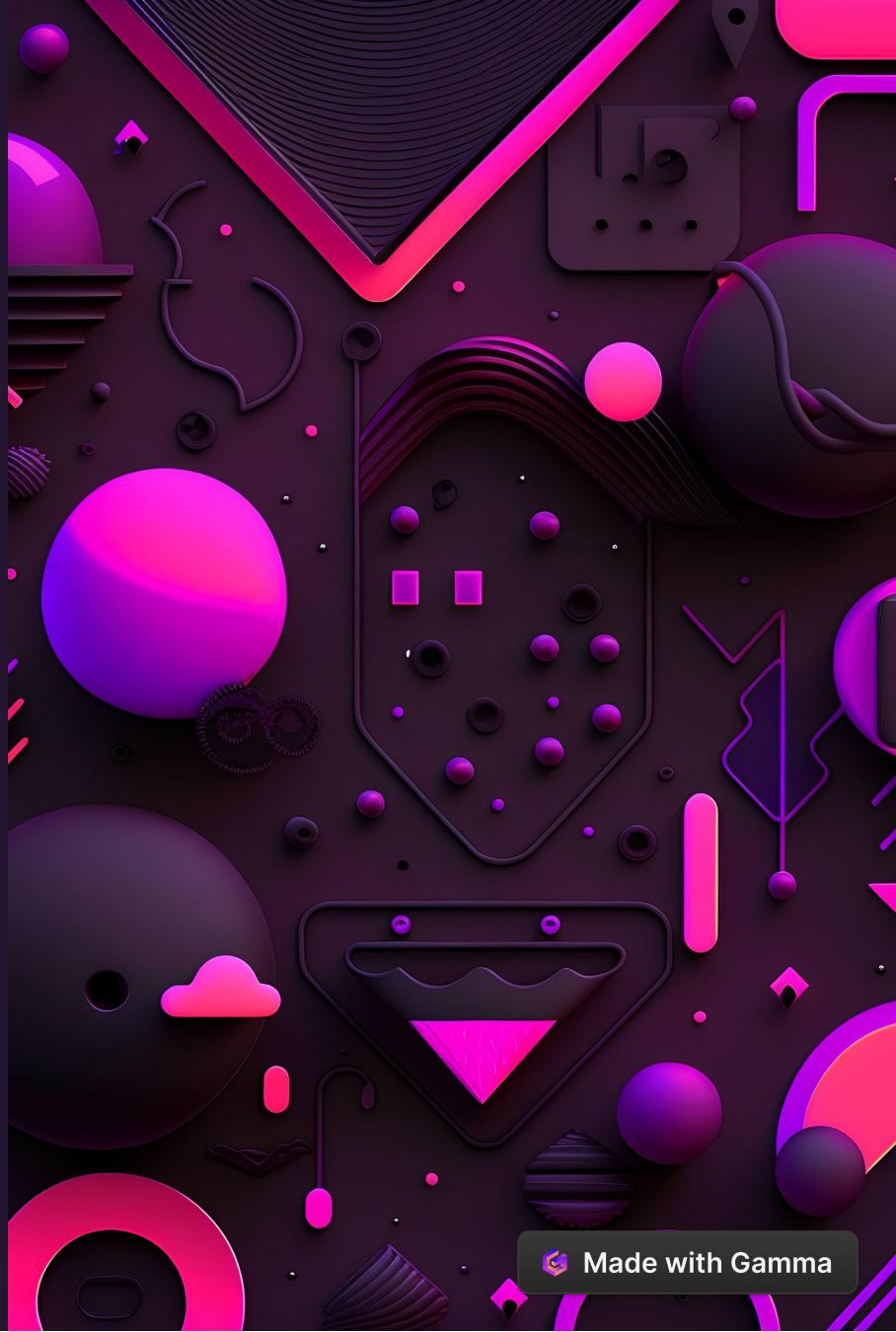
Use the "remove" command in the "os" library. This will completely delete the file, so be careful when using it.

2 Permission issues

Make sure that the file doesn't have any access restrictions before you attempt to delete it. Otherwise, you may receive an error.

3 Backing up data

Always back up your files before deleting them, so that you can recover the data if you need to.



Error Handling in File Handling

Understanding errors

File handling can produce errors if there are insufficient permissions, incorrect file paths or data types, or incorrect command syntax.

Error handling mechanisms

To handle errors, use try-except statements. This will catch any errors that might occur and allow your program to continue running.

Debugging errors

Use debugging tools such as print statements and debugging software to help you identify and fix any errors in your code.

ls

What is it?

The `ls` command is a common command-line utility used in Unix-like operating systems, including Linux and macOS, to list files and directories in a specified directory.

Customizable Output

It provides various options and arguments to customize the output.

Basic Syntax:

Here's the basic syntax of the `ls` command:

```
ls [options]  
[file/directory]
```

pwd

The **pwd** command, which stands for "print working directory," is a command-line utility used in Unix-like operating systems, including Linux and macOS, to display the current working directory, i.e., the directory in which you are currently located within the file system.



cd

The `cd` command, which stands for "change directory," is a fundamental command in Unix-like operating systems, including Linux and macOS. It allows you to change your current working directory to a specified directory, allowing you to navigate through the file system.

cat

The **cat** command, short for "concatenate," is a commonly used command-line utility in Unix-like operating systems, including Linux and macOS. It is primarily used for displaying the contents of one or more text files to the terminal. However, it can also be used for combining and copying the contents of files.

Touch

The **touch** command in Linux is used to create empty files and update the timestamps (access and modification) of existing files. It's a simple yet useful command for various file-related tasks.

`cp`

The `cp` command in Linux is used to copy files and directories from one location to another. It allows you to duplicate files and directories, making it a fundamental tool for managing and backing up data.

`mv`

The `mv` command in Linux is used for moving and renaming files and directories. It's a versatile command that allows you to change the location or name of a file or directory.

`rm`

The `rm` command in Linux is used to remove (delete) files and directories. It's a powerful command that should be used with caution because deleted files are typically not recoverable unless you have a backup.



grep

The `grep` command in Linux is a powerful text-search tool used to search for text patterns or regular expressions within files or streams of text data. It is particularly useful for finding specific lines of text in files, filtering and extracting data, and performing complex text processing tasks.