

# Files

---

## What is a file?

- A file is a chunk of data that contains text or binary data like graphics and audio files
- Many formats for different types of data
- File system - structure data in ways that makes it easier to find files
- Linux inherited Unix philosophy that everything is a file, including devices -Regular files -binary or text - saved on a storage device, read and written to depending on permissions - files also have meta-data so the operating system knows what type of file it is - type of file - permissions - size - creation data
  - Directory file
    - list of other files, organizes the system
    - reading the file is reading a list of other files
    - associates data blocks with file names
  - Devices
    - printers, screens, storage devices, and every other device is represented in the OS as a file
      - Block Device File
        - drivers communicate by sending blocks of data
        - ie. harddisks, ssds, USB cameras
      - Character Device File
        - drivers communicate by sending single characters (bytes, octets)
        - ie. serial ports, parallel ports, sound cards, keyboards, printers
        - writing data to the printer file sends that data to the printer
        - Can be virtual files, not representing a real device at all
          - ie. `/dev/zero` device is a virtual device that outputs an endless supply of zeros
            - Could be used to wipe a hard drive by writing zeros to it
          - `/dev/null` is a virtual device that swallows everything written to it, making it disappear
            - Can write output from a command to the null device and it won't be outputted
  - Virtual Files
    - files that don't exist anywhere, but can be read and written to (weird)
      - ie. `/proc/cpuinfo`
        - has a size of 0 bytes, created on the fly by the OS
  - Pipe File
    - Output of an application is a file that hasn't been saved yet
    - A pipe file can allow one application to read data from another completely unrelated application
    - As one application writes data to the pipe file, the other application reads data from it
- Everything being a file allows you view important system info and write data to devices with simple tools

- It allows simple tools to be combined to solve more complex problems by sharing data in pipe files, all without additional software

[next](#)