# Introduction

Searching for files efficiently is crucial in Linux system administration. Whether looking for files based on name, size, modification time, or other attributes, Linux provides powerful tools like **find, locate, du**, and **ls** to streamline the process.

# Common Practical Examples

## 1. Searching for Files by Name

### Find a file by exact name in the current directory and subdirectories

find . -name "myfile.txt"

### Find a file by name, ignoring case sensitivity

find /home -iname "myfile.txt"

### Find multiple files with different extensions

find /var/log -type f \( -name "\*.log" -o -name "\*.txt" \)

## 2. Searching for Files by Size

### Find files larger than 100MB

find / -type f -size +100M

### Find files smaller than 10KB

find /home/user -type f -size -10k

### Find files between 50MB and 200MB

find /var -type f -size +50M -size -200M

## 3. Searching for Files by Modification Time

### Find files modified in the last 7 days

find /etc -type f -mtime -7

### Find files not modified in the last 30 days

find /home -type f -mtime +30

### Find files accessed in the last 24 hours

find /var -type f -atime -1

## 4. Listing Files and Checking Size

### List all files in a directory with human-readable sizes

ls -lh /home/user

### List only directories

ls -d \*/

### Find the top 10 largest files in a directory

du -ah /var/log | sort -rh | head -10

### Check total disk usage of a directory

du -sh /home/user

## 5. Using the Locate Command for Faster Searching

### Update the locate database (run as root)

updatedb

### Find a file using locate (faster than find)

locate myfile.txt

### Find files containing a specific word

locate --ignore-case "config"

# Additional Notes

* **find** is the most powerful search tool, but it can be slower.
* **locate** is much faster but requires an updated database.
* **du** helps in identifying space-consuming files.
* **ls -lh** provides a quick glance at file sizes and details.