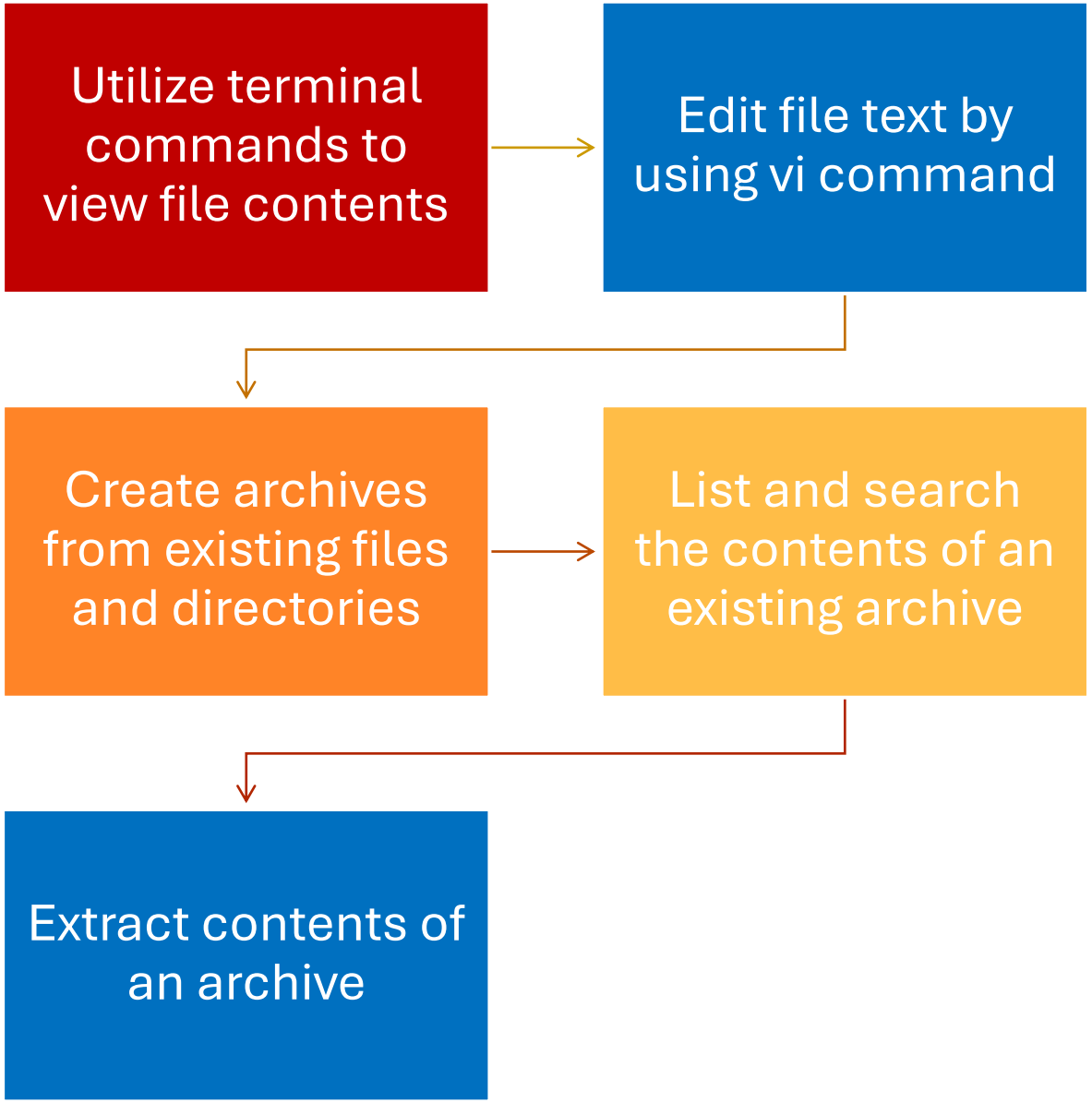
Viewing Files

By: Michael Ilodigwe

Objectives



cat

Displaying Entire File Contents

Concatenates and displays the entire content of one or more files.

Options

-n	Number all output lines
-b	Number nonempty output lines.
-s	Suppress repeated adjacent empty lines.

Syntax

- **cat [filename]**

cat

Example 1 & 2

Example 1: Using `cat` with `-n` option

The `-n` option numbers all output lines, including empty lines:

bash Copy code

```
$ cat -n file.txt
     1  This is line 1.
     2  This is line 2.
     3
     4  This is line 4.
```

In this example, `file.txt` contains four lines, including an empty line. The `-n` option prepends line numbers to each line of output.

Example 2: Using `cat` with `-b` option

The `-b` option numbers non-blank lines only:

bash Copy code

```
$ cat -b file.txt
     1  This is line 1.
     2  This is line 2.
        This is line 4.
```

Here, the output skips numbering the empty line (line 3 in `file.txt`) because `-b` only numbers lines containing text.

cat

Example 3

Example 3: Using `cat` with `-s` option

The `-s` option squeezes multiple adjacent empty lines into one:

bash Copy code

```
$ cat -s file.txt
This is line 1.

This is line 2.

This is line 4.
```

In this example, `file.txt` has two empty lines between lines 1 and 2. Using `-s` condenses these into a single empty line, making the output cleaner.

more

Viewing Text File Page by Page

Allows viewing text files page by page, enabling better readability of large files.

Options

+n	Start displaying from line n
/pattern	Search for the next occurrence of pattern.
-n	Display n lines at a time.

Syntax

- **more [filename]**

more

Example

less

Viewing Text Files with Scrolling Capability

Similar to more, less allows viewing text files page by page with additional scrolling capability.

Options

N	Display line numbers
-I	Ignore case in searches.
-F	Quit if entire file can be displayed on one screen.

Syntax

- **less [filename]**

less

Example

head

Displays lines at the top of a file

Used to Display the beginning lines of a text file or input

Options

-n	Display the first N lines (default is 10
-c	Display the first N bytes.
-q	Suppresses headers when multiple files are listed.
-v	Always display headers when multiple files are listed.

Syntax

- **head [option] [filename]**
- Example: head –n 5 file.txt
  - Displays the first 5 lines

head

Example

tail

Displays lines at the end of the file

Used to Display the last lines of a text file or input

Options

-n	Display the last N lines (default is 10)
-c	Display the last N bytes.
-f	Output appended data as the file grows (follow mode).
-q	Suppress headers when multiple files are listed.
-v	Always display headers when multiple files are listed.

Syntax

- **tail [option] [filename]**
- Example: tail –n 5 file.txt :
  - Displays the last 5 lines of file.txt

tail

Example

find

Find files in Terminal

Used to search for files and directories in a filesystem

Syntax | find [path] [expression]

- Path is the starting point
- Expression: Specifies the search criteria

Common Expressions

-name	Search for files by name using shell-style wildcards.
-type	Search for files of a specific type (e.g., f for regular files, d for directories).
-size	Search for files based on their size.
-user	Search for files owned by a specific user.
-mtime	Search for files modified within the last n days.

Advanced Options:

-exec	Execute a command for each matched file.
-print	Print the path of each matched file (default action if no other action is specified).

find

Example

view

Viewing a file

Viewing a file in read only

Syntax

- view <file>

view

Example

Demonstration

How to view file text

Vi

Text Editing Files

vi

Text Editing Files

Vi is a text editor used for creating and editing text files in Linux.

Syntax

- vi [filename]

Commands to use in Vi

Navigation

You can utilize various letters for different purpose Here is a breakdown

h	Move the cursor left
j	Move the cursor down.
k	Move the cursor up.
l	Move the cursor right.

Commands to use in Vi

Editing

i	Enter insert mode before the cursor.
a	Enter insert mode after the cursor.
o	Open a new line below the current line and enter insert mode.
O	Open a new line above the current line and enter insert mode.
r	Replace the character under the cursor.

Commands to use in Vi

Deletion

x: Delete the character under the cursor.

dd: Delete the current line.

D: Delete from the cursor position to the end of the line.

dw: Delete the word under the cursor.

Commands to use in Vi

Undo

u: Undo the last change.

Ctrl + r: Redo the last undone change.

Commands to use in Vi

Copying, Cutting, Pasting

Copying, Cutting, and Pasting

yy	Copy the current line.
yw	Copy the word under the cursor.
p	Paste after the cursor.
P	Paste before the cursor.

Commands to use in Vi

Saving and Exiting

Saving and Exiting	
w	Save the changes to the file.
q	Quit without saving.
wq	Save and quit.
q!	Quit Vi without saving changes

Demonstration

Using vi to edit text

Tar Command

Introduction to Compression

Tar Command

Definition

tar stands for “tape archive” a command-line utility for creating and manipulating archive files.

An archive is a file that combines files and directories

Tar Command

-x

Extracts files from an archive.

-t

Lists contents of an archive.

-z

Compresses using gzip.

-j

Compresses using bzip2.

-f

Specifies the filename.

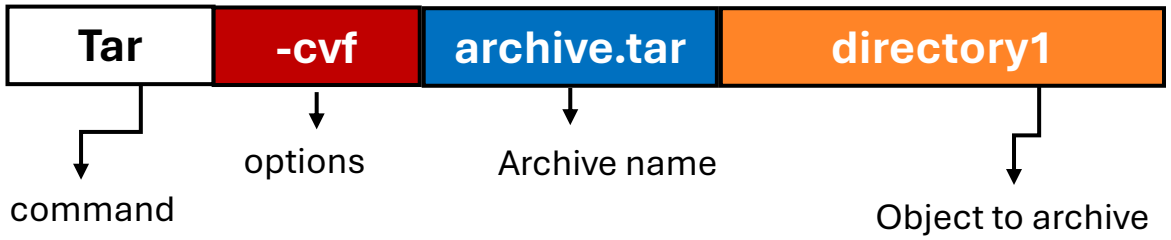
Creating an Archive

tar [option(s)] [archive_name] [objects_to_archive]

Options

c	creates a new archive
v	shows verbose output (optional)
f	specifies the archive file name.

Example



Demonstration

How to create an archive

Compressing an Archive

gzip | gunzip

When disk space is unavailable, gzip compression is beneficial

gzip archive.tar will compress the archive with a **.gz** extension

gunzip archive.tar.gz will unzip an archive and **remove** the **.gz** extension

Demonstration

File Compression

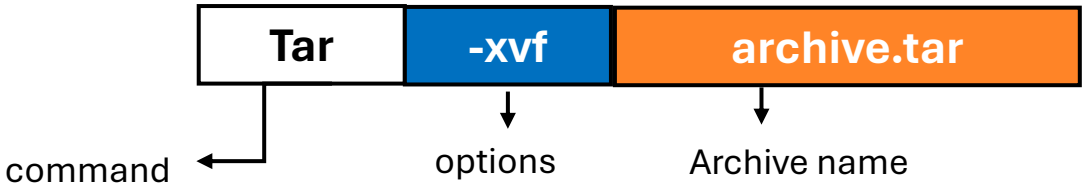
Extracting an Archive

tar [option(s)] [archive_name]

Options

c	creates a new archive
v	shows verbose output (optional)
f	specifies the archive file name.
z	Compresses using gzip
x	Extracts files from an archived file

Example



Demonstration

Archive Extraction

Tips and Best Practices

Include Directories

- Use -C to include directories.
- Example: tar -cvf archive.tar -C /path/to/directory .

Handling Large Archives

- Use -z or -j for compression to reduce archive size.
- Example: tar -czvf archive.tar.gz directory

Verifying Archives

- Use -t to list contents and verify archive integrity.
- Example: tar -tvf archive.tar