**Files, Directories, and Paths**

* **date**: Displays the current system date and time
* **echo**: Prints text or variables to the terminal.
* **pwd** (Print Working Directory): Shows the full path of the current directory.
* **cd** (Change Directory): Changes to the specified directory (cd /home/user)
* **cd .** Refers to the current directory (no effect)
* **cd ..** Moves up one level in the directory hierarchy
* **ls**: lists files and directories in the current directory
* **ls -l**: (Long Listing Format): Displays detailed information about files and directories (permissions, owner, size, date, etc.)
* **man ls**: Opens the manual (help documentation) for the ls command.

**File Naming/Content Representation**

* **\***:Matches zero or more characters in filenames
  + **\*.txt** matches all .txt files.
* **?**:Matches exactly one character.
  + **file?.txt** matches file1.txt, fileA.txt.
* **[]**:Matches any one character from the set inside the brackets.
  + **file[12].txt** matches file1.txt, file2.txt
* **{}**: Used for brace expansion (e.g., creating multiple files or directories).
  + **echo file{1,2,3}.txt** → file1.txt file2.txt file3.txt

**Content Creation**

* **mkdir:** Creates a new directory (mkdir new\_folder)
* **cp**: Copies files or directories (cp file1.txt file2.txt)
* **mv:** Moves or renames files/directories (mv old.txt new.txt)
* **rm**: Removes files (rm file.txt)
* **rmdir**: Removes an empty directory (rmdir empty\_folder)

**Accessing Content**

* **more**: Views file content one page at a time; forward only
* **less**:More advanced than more; allows both forward and backward navigation.
* **head**: Displays the first few lines of a file; default: 10 lines (head file.txt)
* **tail**: Displays the last few lines of a file (tail file.txt)
* **wc** (Word Count): Counts lines, words, and characters (wc file.txt)
* **cat**: Concatenates and displays file contents (cat file.txt)

**Redirecting Content**

* **stdin** (Standard Input): usually from the keyboard
* **stdout** (Standard Output): usually to the terminal.
* **stderr** (Standard Error Output): used for error messages.
* **<**: Redirects input from a file (command < input.txt)
* **>**: Redirects output to a file; overwrites (echo "Hello" > out.txt)
* **|** (Pipe): Sends output of one command as input to another (ls | grep "txt")

**Querying Content**

* **sort**: Sorts lines in a file (sort file.txt)

|  |  |
| --- | --- |
| Flag | Meaning |
| -k N | Sort by field number N |
| n | Numeric sort (e.g., 2 < 10) |
| r | Reverse order |
| -k A -k B | Primary sort by A, secondary by B |

* + **sort -k 3 months**: sort starting at the 3rd field (season)
    - lines are sorted alphabetically by the season: fall, spring, summer, winter
    - Within each season group, the order of months is not guaranteed; just by season
  + **sort -k 3 -k 2n months**: sort primarily by the 3rd field (season), then secondarily by the 2nd field (month number) numerically
  + **sort -k 3 -k 2rn months**: primary sort by season, secondary sort by month number, numerically and in reverse
* **uniq**: Removes duplicate lines; requires sorted input (uniq sorted.txt)
* **cut**: Extracts specific fields/columns from text (cut -d',' -f1 file.csv)
* **grep**: Searches for patterns in files using regular expressions (grep "pattern" file.txt)

**Comparing Content**

* **diff**: Compares two files line by line and shows the differences (diff file1.txt file2.txt)
  + Output is in the form of instructions to convert one file to another
  + Useful for seeing what changed between two files
* **Comm**: Compares two sorted files line by line and displays lines only in file1, lines only in file2, and lines common to both (comm file1.txt file2.txt)
  + Options:
  + -1 suppress column 1 (lines unique to file1)
  + -2 suppress column 2 (lines unique to file2)
  + -3 suppress column 3 (lines common to both)
  + **comm -1 -2 file1.txt file2.txt**: show only common lines
  + **comm -2 -3 file1.txt file2.txt**: show lines only in file1
  + **comm -1 -3 file1.txt file2.txt**: show lines only in file2

**Archiving Content**

* **gzip**: Compresses a file using the GNU zip algorithm (gzip file.txt → create file.txt.gz)
  + Replaces the original file with a .gz version
  + Commonly used for fast, lightweight compression
* **gunzip**: Decompresses .gz files created with gzip (gunzip file.txt.gz → restore file.txt)
  + Restores the original file
  + Often used in pipelines for reading compressed logs or data
* **bzip2**: Compresses file using the Burrows-Wheeler algorithm (bzip2 file.txt → create file.txt.bz2)
  + Provides better compression ratio than gzip, but slower
  + Original file is replaced by .bz2 file
* **bunzip2**: Decompresses .bz2 files created with bzip2 (bunzip2 file.txt.bz2 → restore file.txt)
  + Restores the original uncompressed file
  + Used to read .bz2 archives or prepare for extraction
* **tar**: Archives multiple files/directories into one .tar file (tar -cvf archive.tar file1 file2 dir/)
  + -c to create, -x to extract, -v for verbose, -f to specify filename
  + Does not compress by default; often combined with gzip or bzip2
  + Compress with gzip: (tar -czvf archive.tar.gz file1 dir/)
  + Extract gzip archive: (tar -xzvf archive.tar.gz)
  + Compress with bzip2: (tar -cjvf archive.tar.bz2 file1 dir/)
  + Extract bzip2 archive: (tar -xjvf archive.tar.bz2)