

Basic CLI commands

| Command | Description |
|--------------------|---|
| whoami | Determine the current username (who am I?) |
| pwd | Present working directory (where am I?) |
| Ls (-la) | List the content of a directory (with details) |
| cd <i>pathname</i> | Change directory (folder) in the file system |
| cd / | Move to the root folder of the file system |
| cd .. | Move one level up (one folder) in the file system |
| cp | Copy a file to another folder |
| mv | Move a file to another folder |
| mkdir | Creates a new directory (folder) |
| cat | Display a file |
| rm <i>filename</i> | Removes a file |
| rm -r (or rmdir) | Remove a directory |
| clear | Clears the CLI window |
| exit | Closes the CLI window |
| man <i>command</i> | Shows the manual for a given command |

Useful tricks to avoid excessive command typing



- Use **copy/paste**: ctrl+shift+c (copy) and ctrl+shift+v (paste).
- Use **Up/Down arrow keys**: Cycle through recently executed commands.
- Use the **TAB key**: Autocomplete file/directory name.
- **history** command: list all recently used commands. Users can copy and paste a desired command to execute it again.
- The wildcard ***** symbol represents a string of any character of any length.



Default login path: /home/username \equiv ~
e.g., /home/username/Documents \equiv ~/Documents

Basic CLI commands: File manipulation

> or >> : Redirecting output
| : Piping output

Text files: Human-readable, can be viewed and modified using a text editor

- Text documents (e.g., README files)
- Data in text format (e.g., FASTA, FASTQ, VCF, ...)
- Scripts:
 - Bash scripts (*.sh or *.csh)
 - Python scripts (*.py)
 - R scripts (*.R)

Bash script

A bash script is a file containing a sequence of commands that are executed by the bash program line by line. It allows you to perform a series of actions, such as navigating to a specific directory, creating a folder, and launching a process using the command line.

Script manipulation

| | |
|------------------------------|---|
| <code>#!/bin/bash</code> | Bash script header |
| <code>chmod +x script</code> | Give execution permission to a <i>bash</i> script |
| <code>./script.sh</code> | Run a <i>bash</i> script |

```
-rwxrwxr-x 1 mgem mgem 13 Aug 21 12:42 script.sh
```

↑ ↑ ↑ Execution permission

| Create and print file | |
|---|--|
| <code>nano first.txt</code> | Create a text file (.txt) named <i>first</i> |
| <code>echo "Hello world!" > first.txt</code> | Insert “Hollo word” into <i>first.txt</i> file |
| <code>cat first.txt</code> | Print content of <i>first.txt</i> file |
| <code>cat file1.txt file2.txt > file3.txt</code> | Concatenate the contents of the file1/2 into <i>file3.txt</i> |
| <code>echo “RandomText“ >> first.txt</code> | Append the content of <i>first.txt</i> file |
| <code>grep “o” first.txt</code> | Search for pattern “o” matches |
| <code>cat first.txt grep "o"</code> | Piping the output of <code>cat</code> into <code>grep</code> command |
| <code>wc -l first.txt</code> | Count the number of lines in <i>first</i> file |
| <code>grep "o" first.txt wc -l</code> | Count number of lines with pattern “o” |
| <code>head first.txt</code> | Display top line of a file |
| <code>tail first.txt</code> | Display bottom of a file |

| Compression and Archives | |
|--|--|
| <code>gzip first.txt</code> | Compress <i>first</i> file into <i>first.txt.gz</i> file |
| <code>gunzip first.txt.gz</code> | Uncompress <i>first</i> file |
| <code>zip -r folder.zip folder/</code> | Compress <i>folder</i> to <i>folder.zip</i> file |
| <code>unzip folder.zip</code> | Uncompress folder <i>folder.zip</i> |
| <code>tar -cvzf folder.tar.gz folder/</code> | Compress <i>folder</i> into <i>folder.tar.gz</i> |
| <code>tar -xvzf folder.tar.gz</code> | Extract <i>folder.tar.gz</i> |