# Unix Basics

## Exploring Files and Directories, Orientation and Navigation

Objective: To practise **ls**, **cd**, **pwd**, **mkdir**, **rm**, and **rm -rf** by creating, navigating, and cleaning up directories.

* ls – List files in a directory.
* cd – Change directory.
* pwd – Print working directory.
* mkdir – Create a new directory.
* rm -rf – Delete files/directories (use with caution).

## Exercise 1.

1. Check where you are now. What command did you use?

You should be in your current working directory.

1. List its contents. What command did you use?

See which files and folders already exist.

1. Create a new directory called **ProgSD**. What command did you use?
2. Move into the new directory. What command did you use?
3. Verify your location (Confirm that you are inside **ProgSD**). What command did you use?
4. Create a subdirectory inside **ProgSD** called **ProgSDExam**. What command did you use?
5. List the contents again (You should now see **ProgSDExam).** What command did you use?
6. List the contents of **ProgSD** again so that it shows all the permissions, ownership , size, timestamps. What command did you use?
7. Navigate into **ProgSDExam** and confirm. What two commands did you use?
8. Go back up one level to **ProgSD**. What command did you use?
9. Remove the **ProgSDExam** directory (and its contents, if any). What command did you use?
10. Create the following directory structure **work/docs/reports** in one step. What command did you use?
11. Navigate into **reports**, create a text file **myfile.txt** with a single line. What commands did you use? (Note: You can also create a file *inside* reports without cd).
12. Delete only **myfile.txt**. What command did you use?
13. Delete **work/docs/reports**.

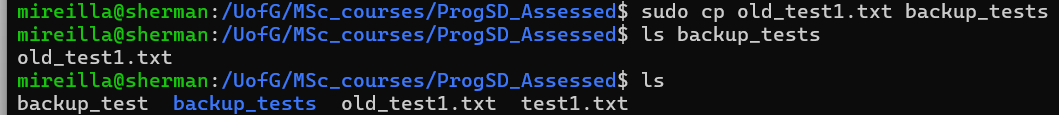
### Copy (cp), move (mv)

The cp (copy) command copies files from any location to another. If the final destination is a directory, **copy** copies the other files into that directory.

A screen shot of a computer

AI-generated content may be incorrect.

Example 1: The file is copied into the **backup\_tests** directory



Example 2: A copy of the file **old\_test1.txt** is made and given a different name: **old\_test50.txt**. If a file with the name **old\_test50.txt** already exists, it will be overwritten.

A screen shot of a computer program

AI-generated content may be incorrect.

The **mv** (move) command moves and renames files. It can also be used to rename a file by moving it to a new name in the same directory.

A screen shot of a computer

AI-generated content may be incorrect.

## Exercise 2.

1. Create a directory structure starting in your current working directory with the following structure:

**adventure** (main folder)  
├── **forest/cave**  
└── **village/house**

1. Navigate into **forest/cave** and confirm where you are. What two commands did you use?
2. List what is inside the **cave**. What command did you use?
3. Write a **single** command to back up to the **adventure** folder and confirm your location. What two commands did you use for that single line (Check the practical task that shows how to combine commands in a single line)?
4. Create another directory called **lake** inside **adventure**. What command did you use?
5. List everything in **adventure,** including their **permissions, ownership, size** and **timestamps**. What command did you use?
6. List everything in **adventure**, including hidden files. What command did you use?
7. Remove only the **lake** directory. What command did you use?
8. Navigate into **village/house** and confirm location. What two commands did you use?
9. **Go back up to adventure** and then remove the entire **adventure** folder and all its contents.

## Orientation & Navigation

### More shell basics: ls, cd, pwd, man, cp, mv, rm, cat, less, redirection (>, >>, |)

Create a new directory called lab1 and move to that directory.



Inside it, create three files:



List the files in the directory and save the output to a file list.txt



Append the current working directory to the same file:



View the file with **cat** and scroll through it with **less**.

Use **man ls** to explore the manual page for **ls**.

### Question: Why is redirection useful in log analysis?

Because log files are usually huge, and you do not want to read them all at once on screen. In log analysis, redirection turns raw logs into focused, reusable datasets.

* With redirection (> and >>), you can save command outputs into new files for later analysis.
  + Example: grep "Failed password" /var/log/auth.log > failed.log → collects only failed login attempts.
* With pipes (|), you can chain commands to filter, sort, and count results.
  + Example: cat failed.log | awk '{print $11}' | sort | uniq -c | sort -nr → shows top IPs causing login failures.