Introduction to Operating Systems

2. Files Management

2026-11-03

USO Team

Bibliography

1. **Brian Ward**, *How LINUX Works*, 3rd Edition, No Starch Press, 2021
   - Chapter 2 - *Basic Commands and Directory Hierarchy*
   - Chapter 4 - *Disks and Filesystems*
     - Sections 4.1 and 4.2
2. **Razvan Deaconescu, Razvan Rughinis, Mihai Carabas, Alexandru Radovici**, *Utilizarea Sistemelor de Operare*, Printech 2021, download
   - Chapter 2 - *Utilizarea sistemului de fișiere*
     - Sections 2.1 - 2.3

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Best file compression around

`DEL *.*` = *100% compression*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Files

how the file system is laid out

Bibliography

for this section

1. **Brian Ward**, *How LINUX Works*, 3rd Edition, No Starch Press, 2021
   - Chapter 2 - *Basic Commands and Directory Hierarchy*
     - Section 2.19 - *Linux Directory Hierarchy Essentials*
2. **Razvan Deaconescu, Razvan Rughinis, Mihai Carabas, Alexandru Radovici**, *Utilizarea Sistemelor de Operare*, Printech 2021, download
   - Chapter 2 - *Utilizarea sistemului de fișiere*
     - Section 2.1.2 - *Structura ierarhică a sistemului de fișiere*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

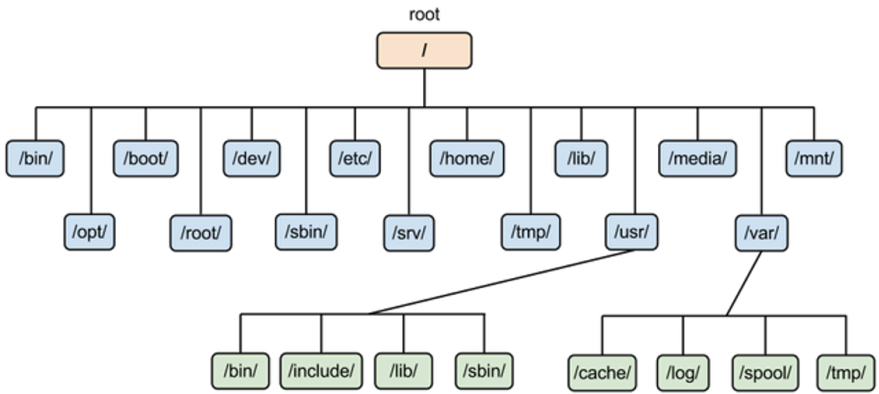
Windows Layout

- Hierarchical structure starting from **drive letters** (`C:\`, `D:\`, etc.)
- Each partition is an **independent file system**
- Uses **backslashes (`\`)** and is **case-insensitive** by default

```
C:\
├── Windows
│   ├── system32
│   └── ...
├── Program Files
│   ├── App1
│   └── App2
├── Users
│   ├── Alice
│   ├── Bob
│   └── Public
└── Temp
```

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout



⚠ no drive letters, **one single root**

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained

```
/
├── bin
├── boot
├── dev
├── etc
├── home
│   ├── ana
│   └── student
├── lib
├── media
│   ├── cdrom
│   └── usb
├── mnt
│   └── temp
├── opt
├── proc
├── root
├── sbin
├── sys
├── tmp
├── usr
│   ├── bin
│   ├── lib
│   └── share
└── var
```

The file system tree

- every file and directory lives under **one single root** `/`
- each directory has a *job*: programs, configuration, user data ...

> **Windows**
>
> one tree **per drive**: `C:\`, `D:\`, ... every partition gets a drive letter

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained

```
/
├── bin
├── boot
├── dev
├── etc
├── home
│   ├── ana
│   └── student
├── lib
├── media
│   ├── cdrom
│   └── usb
├── mnt
│   └── temp
├── opt
├── proc
├── root
├── sbin
├── sys
├── tmp
├── usr
│   ├── bin
│   ├── lib
│   └── share
└── var
```

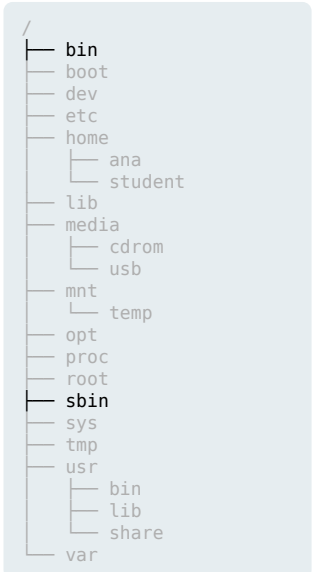
`/` - root

- the top of the hierarchy, the start of every absolute path
- other partitions and disks are *mounted* in directories below it

> **Windows**
>
> `C:\` - the root of the drive Windows is installed on

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



`/bin` and `/sbin` - essential programs

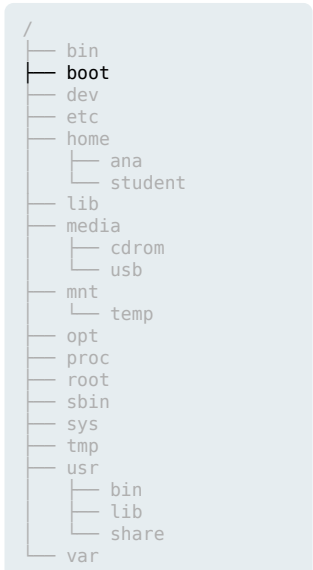
- `/bin` - commands for everyone: `ls`, `cp`, `mv`, `cat`
- `/sbin` - commands for the administrator: `mount`, `fsck`, `reboot`
- 💡 on modern distributions they point to `/usr/bin` and `/usr/sbin`

Windows

`C:\Windows\System32` - `cmd.exe`, `notepad.exe`, `diskpart.exe`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



`/boot` - starting the system

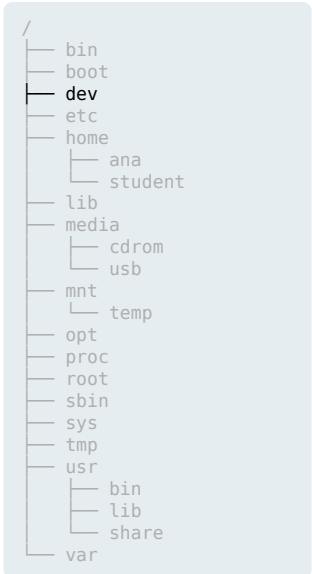
- the kernel: `vmlinuz-6.x`
- the initial RAM disk: `initrd.img-6.x`
- the boot loader configuration: `grub/grub.cfg`

Windows

- the EFI partition (`\EFI\Microsoft\Boot\bootmgfw.efi`)
- `C:\Windows\Boot`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



`/dev` - devices

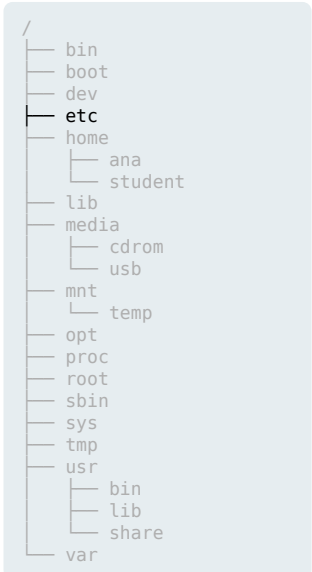
- peripherals seen as files, created by the kernel
- `/dev/sda` - the first disk, `/dev/sda1` - its first partition
- `/dev/null` - discards everything written to it
- `/dev/tty` - the current terminal

Windows

no directory, devices have special names: `\\.\PhysicalDrive0`, `COM1`, `NUL`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



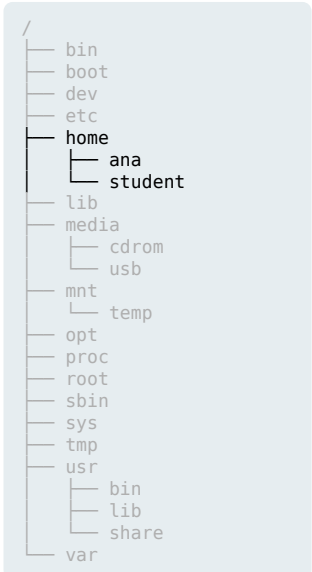
`/etc` - configuration

- system wide settings, usually **text files**
- `/etc/passwd` - the users
- `/etc/hostname` - the name of the computer
- `/etc/fstab` - what partitions to mount and where

> **Windows**
>
> The *Windows Registry* (`HKEY_LOCAL_MACHINE`, ...)

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



`/home` - the users' files

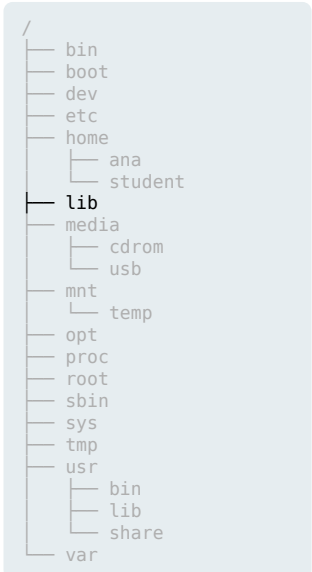
- one directory for every user: `/home/alice`
- documents, downloads, `Movies`, settings (`.bashrc`)
- 🔒 a user can usually write **only** in their own directory

Windows

- `C:\Users\Ana` - `Videos`, `Downloads`, `Documents`
- `C:\Users\Student`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



`/lib` - libraries

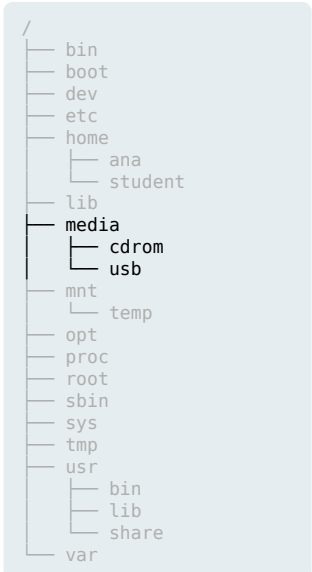
- code shared by the programs in `/bin` and `/sbin`
- `libc.so.6` - the C standard library
- `/lib/modules/6.x/` - kernel modules (drivers)

Windows

- dynamic libraries `.dll` files in `C:\Windows\System32`
- drivers in `C:\Windows\System32\drivers`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



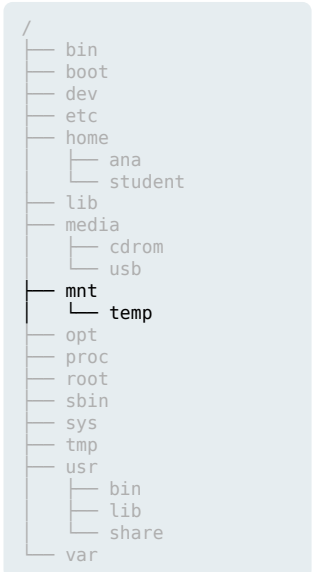
`/media` - removable media

- USB sticks, CDs, SD cards are mounted here *automatically*
- `/media/alice/USB_STICK/`

> **Windows**
>
> a new drive letter: `E:\`, `F:\`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



`/mnt` - temporary mounts

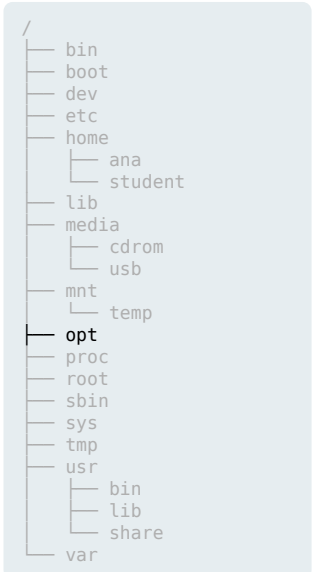
- the administrator mounts file systems here *manually*
- `mount /dev/sdb1 /mnt/temp`

Windows

a drive letter, or a volume mounted in an empty NTFS directory (*Disk Management*)

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



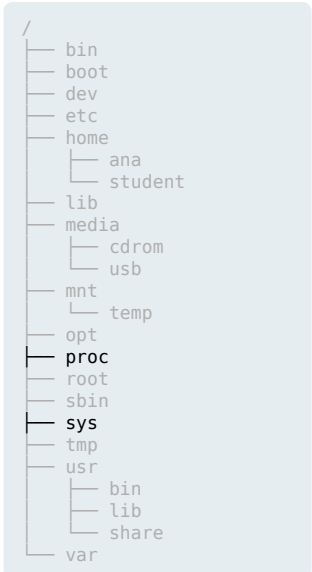
`/opt` - optional applications

- third party applications that keep all their files together
- `/opt/google/chrome/`

> **Windows**
>
> `C:\Program Files\Google\Chrome`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



`/proc` and `/sys` - virtual file systems

- generated by the kernel in memory, **not stored on disk**
- `/proc/cpuinfo` - the processor, `/proc/1234/` - a running process
- `/sys/class/net/` - the network interfaces

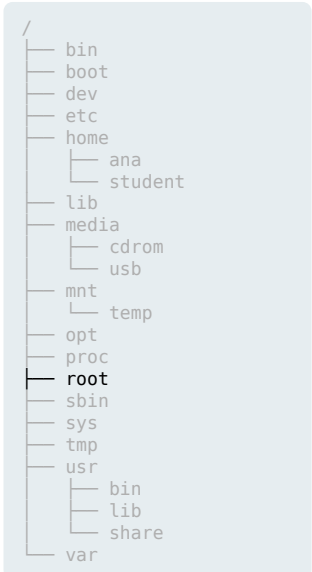
Windows

no files, the same information comes from

- *Task Manager*
- *Device Manager*
- PowerShell (`Get-Process`)

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



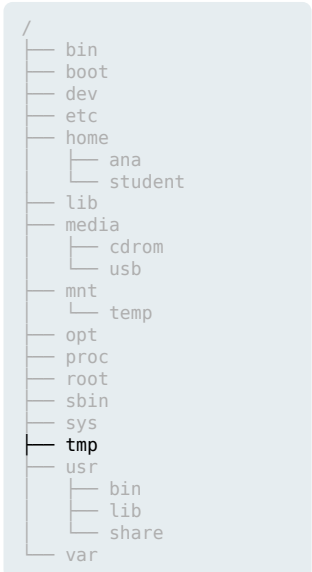
`/root` - the administrator's home

- the home directory of the `root` user
- ⚠️ not the same as `/`, the root of the file system

> **Windows**
>
> `C:\Users\Administrator`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



`/tmp` - temporary files

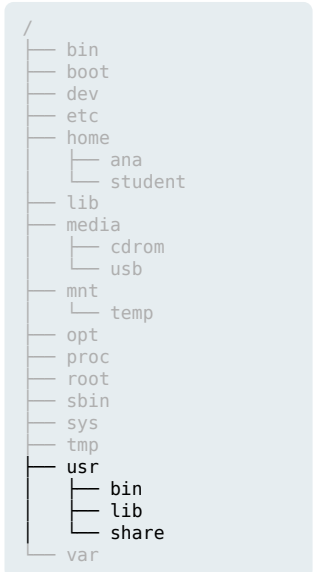
- every user and application can write here
- 🧹 usually emptied when the system restarts

Windows

- `C:\Users\Alice\AppData\Local\Temp` (`%TEMP%`)
- `C:\Windows\Temp`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



`/usr` - installed software

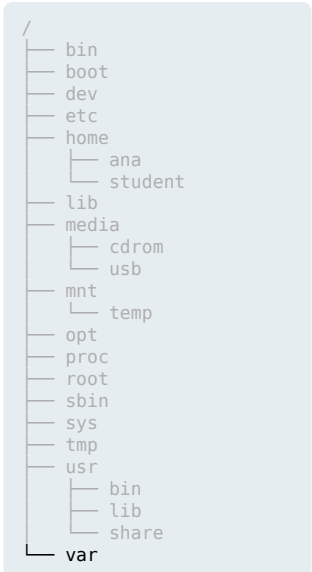
- `/usr/bin` - most of the programs: `firefox`, `python3`, `git`
- `/usr/lib` - their libraries
- `/usr/share` - icons, documentation, *manual pages*

Windows

- `C:\Program Files`
- `C:\Program Files (x86)`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained



`/var` - data that changes

- files that grow while the system is running
- `/var/log/` - logs, `/var/cache/` - caches
- `/var/lib/` - application data (e.g. databases)

> **Windows**
>
> `C:\ProgramData` for application data, `C:\Windows\Logs` and the *Event Viewer* for logs

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Linux (POSIX) Layout Explained

```
/
├── bin
├── boot
├── dev
├── etc
├── home
│   ├── ana
│   └── student
├── lib
├── media
│   ├── cdrom
│   └── usb
├── mnt
│   └── temp
├── opt
├── proc
├── root
├── sbin
├── sys
├── tmp
├── usr
│   ├── bin
│   ├── lib
│   └── share
└── var
```

In short

	Linux	Windows
programs	`/bin`, `/sbin`, `/usr`, `/opt`	`C:\Windows`, `C:\Program Files`
configuration	`/etc`	Registry
user data	`/home`, `/root`	`C:\Users`
changing data	`/var`, `/tmp`	`C:\ProgramData`, `%TEMP%`
virtual	`/dev`, `/proc`, `/sys`	*no files*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Path

absolute and relative

Bibliography

for this section

1. **Brian Ward**, *How LINUX Works*, 3rd Edition, No Starch Press, 2021
   - Chapter 2 - *Basic Commands and Directory Hierarchy*
     - Section 2.4 - *Navigating Directories*
2. **Razvan Deaconescu, Razvan Rughinis, Mihai Carabas, Alexandru Radovici**, *Utilizarea Sistemelor de Operare*, Printech 2021, download
   - Chapter 2 - *Utilizarea sistemului de fișiere*
     - Section 2.1.3 - *Căi relative și căi absolute*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

What Is a Path?

A **path** tells the operating system *where a file or directory is.*

Two main types:

- **Absolute paths** start from the root of the file system: `/home/student/Movies/the_odyssey.mkv`
- **Relative paths** start from your *current directory*: `Movies/the_odyssey.mkv`

💡 every running application has one single *current directory* at a time

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Absolute Path

full path

Shows the **complete location** of a file

Windows

Always starts from a **drive letter**

```
C:\Users\Alice\Videos\the_odyssey.mkv
D:\Trailers\avatar_fire_and_ash.mp4
```

Linux (POSIX)

Always starts from the **root directory** `/`

```
/home/alice/Movies/the_odyssey.mkv
/var/log/syslog
/usr/bin/python3
```

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Absolute Path

full path

Shows the **complete location** of a file

Windows

Always starts from a **drive letter**

```
C:\Users\Alice\Videos\the_odyssey.mkv
D:\Trailers\avatar_fire_and_ash.mp4
```

Linux (POSIX)

Always starts from the **root directory** `/`

```
/home/alice/Movies/the_odyssey.mkv
/var/log/syslog
/usr/bin/python3
```

🤔 **Just think of a real world address of a person. You need:**

🗄️ Country -> 📁 County -> 📁 City -> 📁 Street -> 📁 Street Number -> 📄 person

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Relative Path

to the current directory

- ✅ Shorter
- ⚠️ Depends on the current directory (*where you are*)
- 📦 allows an application to find its own files, regardless of where it is installed (`./textures`)

Windows

Does **NOT** start from a **drive letter**

```
# If you're in C:\Users\Alice
Videos\the_odyssey.mkv
```

Linux (POSIX)

Does **NOT** start from the **root directory** `/`

```
# If you're in /home/alice
Movies/the_odyssey.mkv
```

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Relative Path

to the current directory

- ✅ Shorter
- ⚠️ Depends on the current directory (*where you are*)
- 📦 allows an application to find its own files, regardless of where it is installed (`./textures`)

Windows

Does **NOT** start from a **drive letter**

```
# If you're in C:\Users\Alice
Videos\the_odyssey.mkv
```

Linux (POSIX)

Does **NOT** start from the **root directory** `/`

```
# If you're in /home/alice
Movies/the_odyssey.mkv
```

🤔 **Just think of a real world address of a person *relative to your own current location*:**

🧍 -> 📁 Street -> 📁 Street Number -> 📄 person

- in the example, you already know the 📁 Country and the 📁 City in which you are now

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Special Path Symbols

in Windows and Linux (POSIX)

Meaning	Windows	Linux (POSIX)	Notes
Current directory	`.`	`.`	Refers to the directory you are currently in
Parent directory	`..`	`..`	Moves one level up in the hierarchy
Directory separator	`\`	`/`	Used to separate directory names in a path
Root directory	Drive + `\` (e.g. `C:\`)	`/`	Top-level directory in the file system

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Special Path Symbols

in Windows and Linux (POSIX)

Meaning	Windows	Linux (POSIX)	Notes
Current directory	`.`	`.`	Refers to the directory you are currently in
Parent directory	`..`	`..`	Moves one level up in the hierarchy
Directory separator	`\`	`/`	Used to separate directory names in a path
Root directory	Drive + `\` (e.g. `C:\`)	`/`	Top-level directory in the file system

⚠️ `\` in POSIX systems is used for escaping characters (`\n ...`)

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Special Path Symbols

in Windows and Linux (POSIX)

Meaning	Windows	Linux (POSIX)	Notes
Current directory	`.`	`.`	Refers to the directory you are currently in
Parent directory	`..`	`..`	Moves one level up in the hierarchy
Directory separator	`\`	`/`	Used to separate directory names in a path
Root directory	Drive + `\` (e.g. `C:\`)	`/`	Top-level directory in the file system

⚠ `\` in POSIX systems is used for escaping characters (`\n ...`)

⚠ `~` in `bash`-like (`...sh`) command interpreters is used for the *user's home directory*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

from relative to absolute

`.` and `..`

🧭 where am I, where is my parent

Every directory has two *hidden* entries:

- 📍 `.` - the directory **itself**
- ⬆️ `..` - its **parent** directory

```
📁 /
└── 📁 home
    ├── 📁 alice                    <-- .. (parent)
    │   ├── 📁 Movies               <-- . (current) 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob
        └── 📄 notes.txt
```

we are in `/home/alice/Movies`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`.` and `..`

🧭 where am I, where is my parent

Every directory has two *hidden* entries:

- 📍 `.` - the directory **itself**
- ⬆️ `..` - its **parent** directory

🔁 `..` can be chained, each one goes **one level up**



every `..` climbs one more level

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`.` and `..`

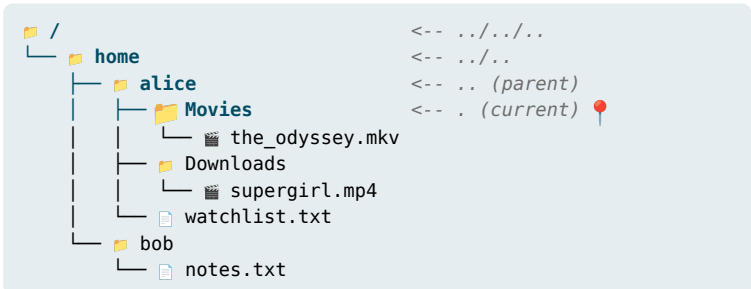
🧭 where am I, where is my parent

Every directory has two *hidden* entries:

- 📍 `.` - the directory **itself**
- ⬆️ `..` - its **parent** directory

🔁 `..` can be chained, each one goes **one level up**

⚠️ `/..` is still `/`, the root has no parent

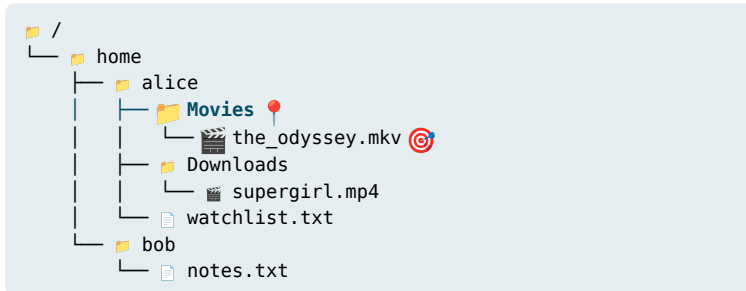


every `..` climbs one more level

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./the_odyssey.mkv`



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./the_odyssey.mkv`

1️⃣ `.` stay in `Movies`

```
📁 /
└── 📁 home
    ├── 📁 alice
    │   ├── 📁 Movies                <-- (1) . 📍
    │   │   └── 🎬 the_odyssey.mkv 🎯
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob
        └── 📄 notes.txt
```

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path



🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./the_odyssey.mkv`

1️⃣ `.` stay in `Movies`

2️⃣ open `the_odyssey.mkv` 🎯

```
📁 /
└── 📁 home
    ├── 📁 alice
    │   ├── 📁 Movies                    <-- (1) . 📍
    │   │   └── 🎬 the_odyssey.mkv       <-- (2) open 🎯
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob
        └── 📄 notes.txt
```

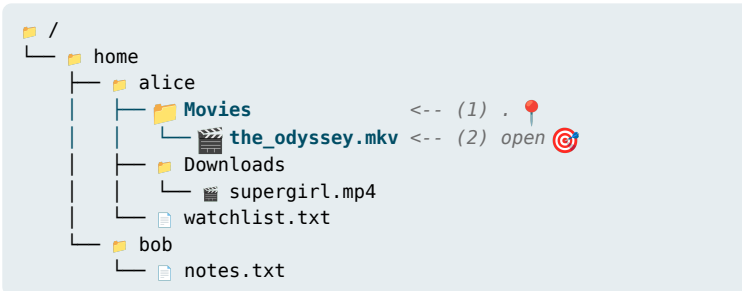
 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./the_odyssey.mkv`

1️⃣ `.` stay in `Movies`

2️⃣ open `the_odyssey.mkv` 🎯



💡 **Tip:** same as `the_odyssey.mkv`

🔗 **pwd** + / + **relative** → **/home/alice/Movies/./the_odyssey.mkv**

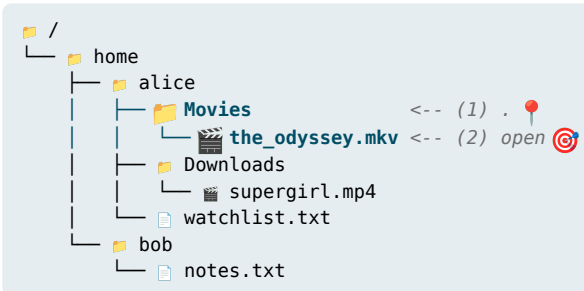
 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./the_odyssey.mkv`

1️⃣ `.` stay in `Movies`

2️⃣ open `the_odyssey.mkv` 🎯



> 💡 **Tip:** same as `the_odyssey.mkv`
>
> 🔗 **pwd** + / + **relative** — **/home/alice/Movies/./the_odyssey.mkv**
> ✂️ drop `.`, each `..` removes a directory — **/home/alice/Movies~~/..~~/the_odyssey.mkv**

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 **`./the_odyssey.mkv`**

1️⃣ `.` stay in `Movies`

2️⃣ open `the_odyssey.mkv` 🎯

```
📁 /
└── 📁 home
    ├── 📁 alice
    │   ├── 📁 Movies                <-- (1) . 📍
    │   │   └── 🎬 the_odyssey.mkv   <-- (2) open 🎯
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob
        └── 📄 notes.txt
```

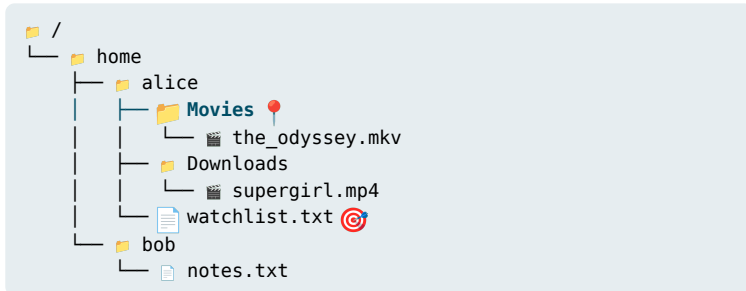
💡 **Tip:** same as `the_odyssey.mkv`

🔗 **`pwd`** + `/` + **`relative`**	**`/home/alice/Movies/./the_odyssey.mkv`**
✂️ drop `.`, each `..` removes a directory	**`/home/alice/Movies~~/..~~/the_odyssey.mkv`**
🌍 absolute path	`/home/alice/Movies/the_odyssey.mkv`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../watchlist.txt`



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../watchlist.txt`

1️⃣ `..` go up to `alice` ⬆️



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../watchlist.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ open `watchlist.txt` 🎯



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../watchlist.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ open `watchlist.txt` 🎯



💡 **Tip:** how the absolute path is calculated

🔗 **pwd** + / + **relative** `/home/alice/Movies/../watchlist.txt`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../watchlist.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ open `watchlist.txt` 🎯

```
📁 /
└── 📁 home
    ├── 📁 alice                    <-- (1) .. (up)
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt        <-- (2) open 🎯
    └── 📁 bob
        └── 📄 notes.txt
```

💡 **Tip:** how the absolute path is calculated

🔗 **pwd** + / + **relative**	**/home/alice/Movies/../watchlist.txt**
✂️ drop `.`, each `..` removes a directory	**/home/alice~~/Movies~~[1]~~/..~~[1]/watchlist.txt**

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../watchlist.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ open `watchlist.txt` 🎯

```
📁 /
└── 📁 home
    ├── 📁 alice                    <-- (1) .. (up)
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt        <-- (2) open 🎯
    └── 📁 bob
        └── 📄 notes.txt
```

💡 **Tip:** how the absolute path is calculated

🔗 **`pwd`** + `/` + **`relative`**	**`/home/alice/Movies/../watchlist.txt`**
✂️ drop `.`, each `..` removes a directory	**`/home/alice`**~~**`/Movies`**~~[1]~~**`/..`**~~[1]~~**`/watchlist.txt`**
🌍 absolute path	`/home/alice/watchlist.txt`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path



🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/supergirl.mp4`

```
📁 /
└── 📁 home
    ├── 📁 alice
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4 🎯
    │   └── 📄 watchlist.txt
    └── 📁 bob
        └── 📄 notes.txt
```

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/supergirl.mp4`

1️⃣ `..` go up to `alice` ⬆️



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/supergirl.mp4`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/supergirl.mp4`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️

3️⃣ open `supergirl.mp4` 🎯



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/supergirl.mp4`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️

3️⃣ open `supergirl.mp4` 🎯

```
📁 /
└── 📁 home
    ├── 📁 alice                    <-- (1) .. (up)
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads            <-- (2) Downloads (down)
    │   │   └── 🎬 supergirl.mp4    <-- (3) open 🎯
    │   └── 📄 watchlist.txt
    └── 📁 bob
        └── 📄 notes.txt
```

💡 **Tip:** how the absolute path is calculated

🔗 **pwd** + / + **relative** → **/home/alice/Movies/../Downloads/supergirl.mp4**

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/supergirl.mp4`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️

3️⃣ open `supergirl.mp4` 🎯

```
📁 /
└── 📁 home
    ├── 📁 alice                    <-- (1) .. (up)
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads            <-- (2) Downloads (down)
    │   │   └── 🎬 supergirl.mp4    <-- (3) open 🎯
    │   └── 📄 watchlist.txt
    └── 📁 bob
        └── 📄 notes.txt
```

💡 **Tip:** how the absolute path is calculated

🔗 **pwd** + / + **relative** — `/home/alice/Movies/../Downloads/supergirl.mp4`

✂️ drop `.`, each `..` removes a directory — `/home/alice~~/Movies~~[1]~~/..~~[1]/Downloads/supergirl.mp4`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/supergirl.mp4`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️

3️⃣ open `supergirl.mp4` 🎯

```
📁 /
└── 📁 home
    ├── 📁 alice                    <-- (1) .. (up)
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads            <-- (2) Downloads (down)
    │   │   └── 🎬 supergirl.mp4    <-- (3) open 🎯
    │   └── 📄 watchlist.txt
    └── 📁 bob
        └── 📄 notes.txt
```

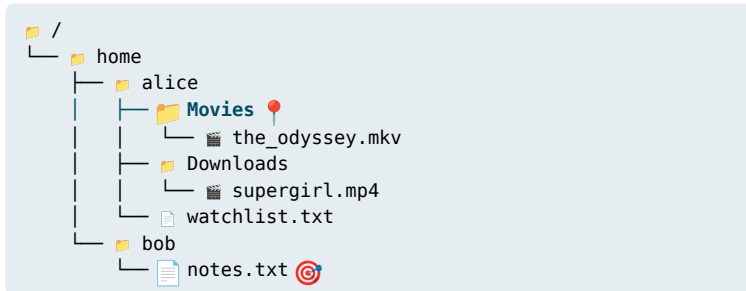
💡 **Tip:** how the absolute path is calculated

🔗 **`pwd`** + `/` + **`relative`**	**`/home/alice/Movies/../Downloads/supergirl.mp4`**
✂️ drop `.`, each `..` removes a directory	**`/home/alice`~~`/Movies`~~[1]~~`/..`~~[1]**`/Downloads/supergirl.mp4`
🌍 absolute path	`/home/alice/Downloads/supergirl.mp4`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../../bob/notes.txt`



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `..` go up to `home` ⬆️



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

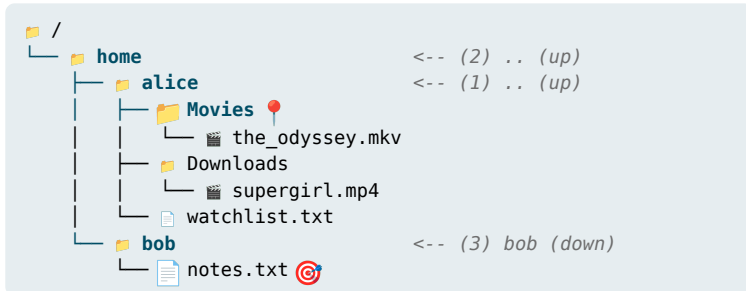
Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `..` go up to `home` ⬆️

3️⃣ `bob` go down ⬇️



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `..` go up to `home` ⬆️

3️⃣ `bob` go down ⬇️

4️⃣ open `notes.txt` 🎯



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `..` go up to `home` ⬆️

3️⃣ `bob` go down ⬇️

4️⃣ open `notes.txt` 🎯

```
📁 /
└── 📁 home                          <-- (2) .. (up)
    ├── 📁 alice                     <-- (1) .. (up)
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob                       <-- (3) bob (down)
        └── 📄 notes.txt             <-- (4) open 🎯
```

💡 **Tip:** how the absolute path is calculated

🔗 **pwd** + / + **relative** → **/home/alice/Movies/../../bob/notes.txt**

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `..` go up to `home` ⬆️

3️⃣ `bob` go down ⬇️

4️⃣ open `notes.txt` 🎯

```
📁 /
└── 📁 home                          <-- (2) .. (up)
    ├── 📁 alice                     <-- (1) .. (up)
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob                       <-- (3) bob (down)
        └── 📄 notes.txt             <-- (4) open 🎯
```

💡 **Tip:** how the absolute path is calculated

🔗 **pwd** + / + **relative**	**/home/alice/Movies/../../bob/notes.txt**
✂️ drop `.`, each `..` removes a directory	**/home/alice~~/Movies[1]/..[1]~~/../bob/notes.txt**

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `..` go up to `home` ⬆️

3️⃣ `bob` go down ⬇️

4️⃣ open `notes.txt` 🎯

```
📁 /
└── 📁 home                          <-- (2) .. (up)
    ├── 📁 alice                     <-- (1) .. (up)
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob                       <-- (3) bob (down)
        └── 📄 notes.txt             <-- (4) open 🎯
```

💡 **Tip:** how the absolute path is calculated

🔗 **pwd** + / + **relative** — `/home/alice/Movies/../../bob/notes.txt`

✂️ drop `.`, each `..` removes a directory — `/home~~/alice~~`[2]`~~/Movies~~`[1]`~~/..~~`[1]`~~/..~~`[2]`/bob/notes.txt`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Walking a Path

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `..` go up to `home` ⬆️

3️⃣ `bob` go down ⬇️

4️⃣ open `notes.txt` 🎯

```
📁 /
└── 📁 home                          <-- (2) .. (up)
    ├── 📁 alice                     <-- (1) .. (up)
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob                       <-- (3) bob (down)
        └── 📄 notes.txt             <-- (4) open 🎯
```

💡 **Tip:** how the absolute path is calculated

🔗 **pwd** + / + **relative**	**/home/alice/Movies/../../bob/notes.txt**
✂️ drop `.`, each `..` removes a directory	**/home~~/alice~~[2]~~/Movies~~[1]~~/..~~[1]~~/..~~[2]**/bob/notes.txt**
🌍 absolute path	/home/bob/notes.txt

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

`.` *and* `..` *anywhere in the path*

Bibliography

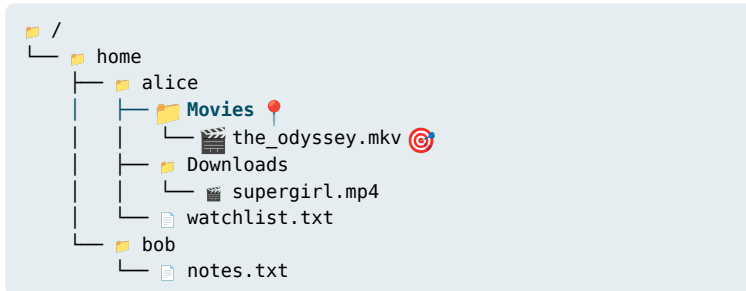
for this section

1. **Brian Ward**, *How LINUX Works*, 3rd Edition, No Starch Press, 2021
   - Chapter 2 - *Basic Commands and Directory Hierarchy*
     - Section 2.4 - *Navigating Directories*
2. **Razvan Deaconescu, Razvan Rughinis, Mihai Carabas, Alexandru Radovici**, *Utilizarea Sistemelor de Operare*, Printech 2021, download
   - Chapter 2 - *Utilizarea sistemului de fișiere*
     - Section 2.1.3 - *Căi relative și căi absolute*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./../Movies/./the_odyssey.mkv`



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./../Movies/./the_odyssey.mkv`

1️⃣ `.` stay in `Movies`



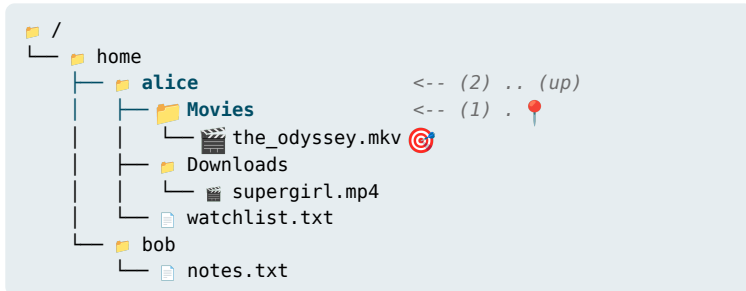
 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./../Movies/./the_odyssey.mkv`

1️⃣ `.` stay in `Movies`

2️⃣ `..` go up to `alice` ⬆️



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

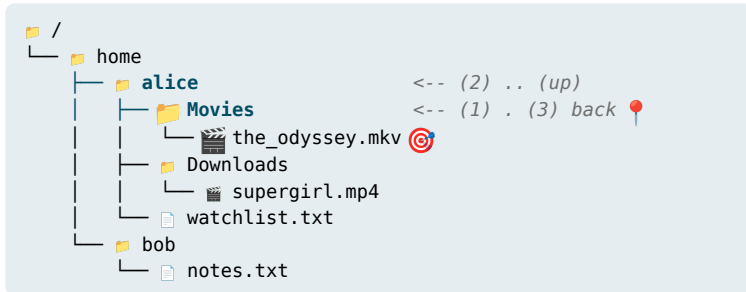
Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./../Movies/./the_odyssey.mkv`

1️⃣ `.` stay in `Movies`

2️⃣ `..` go up to `alice` ⬆️

3️⃣ `Movies` go back down ⬇️

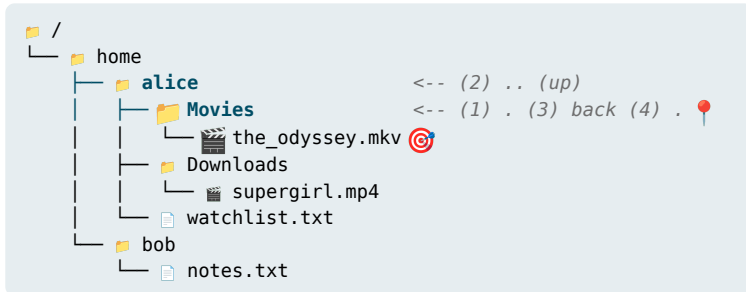


 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 **`./../Movies/.`**`/the_odyssey.mkv`

1. `.` stay in `Movies`
2. `..` go up to `alice` ⬆️
3. `Movies` go back down ⬇️
4. `.` stay in `Movies`



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./../Movies/./the_odyssey.mkv`

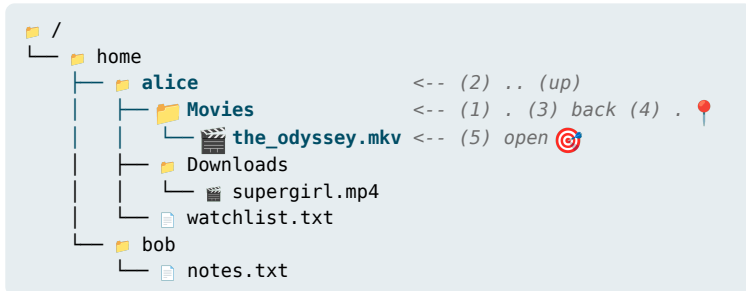
1️⃣ `.` stay in `Movies`

2️⃣ `..` go up to `alice` ⬆️

3️⃣ `Movies` go back down ⬇️

4️⃣ `.` stay in `Movies`

5️⃣ open `the_odyssey.mkv` 🎯



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./../Movies/./the_odyssey.mkv`

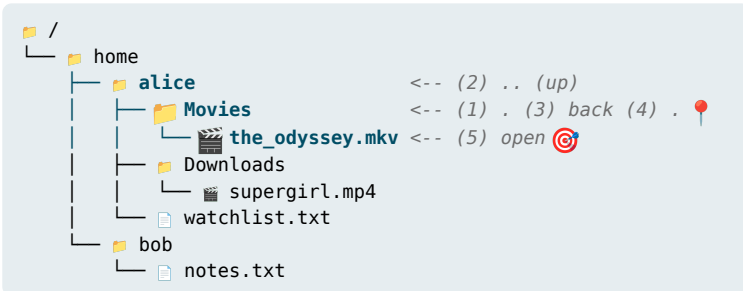
1️⃣ `.` stay in `Movies`

2️⃣ `..` go up to `alice` ⬆️

3️⃣ `Movies` go back down ⬇️

4️⃣ `.` stay in `Movies`

5️⃣ open `the_odyssey.mkv` 🎯



💡 **Tip:** `.` and `..` can appear *anywhere* in a path

🔗 `pwd` + `/` + `relative` → `/home/alice/Movies/./../Movies/./the_odyssey.mkv`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./../Movies/./the_odyssey.mkv`

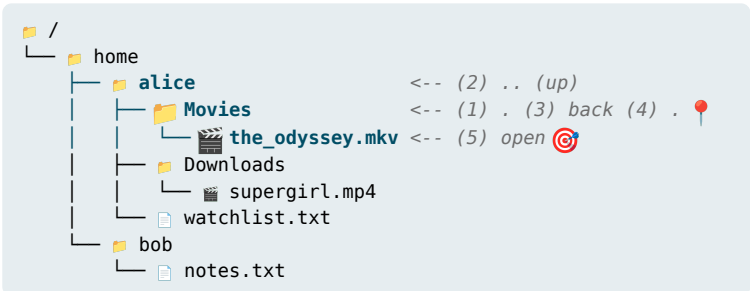
1️⃣ `.` stay in `Movies`

2️⃣ `..` go up to `alice` ⬆️

3️⃣ `Movies` go back down ⬇️

4️⃣ `.` stay in `Movies`

5️⃣ open `the_odyssey.mkv` 🎯



💡 **Tip:** `.` and `..` can appear *anywhere* in a path

🔗 **`pwd` + `/` + `relative`** `/home/alice/Movies/./../Movies/./the_odyssey.mkv`

✂️ drop `.`, each `..` removes a directory `/home/alice/Movies/./../Movies/./the_odyssey.mkv`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./../Movies/./the_odyssey.mkv`

1️⃣ `.` stay in `Movies`

2️⃣ `..` go up to `alice` ⬆️

3️⃣ `Movies` go back down ⬇️

4️⃣ `.` stay in `Movies`

5️⃣ open `the_odyssey.mkv` 🎯

```
📁 /
└── 📁 home
    ├── 📁 alice                    <-- (2) .. (up)
    │   ├── 📁 Movies               <-- (1) . (3) back (4) . 📍
    │   │   └── 🎬 the_odyssey.mkv  <-- (5) open 🎯
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob
        └── 📄 notes.txt
```

💡 **Tip:** `.` and `..` can appear *anywhere* in a path

🔗 `pwd` + `/` + `relative` → `/home/alice/Movies/./../Movies/./the_odyssey.mkv`

✂️ drop `.`, each `..` removes a directory → `/home/alice~~/Movies~~[1]~~/.~~~~/..~~[1]/Movies/./the_odyssey.mkv`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./../Movies/./the_odyssey.mkv`

1️⃣ `.` stay in `Movies`

2️⃣ `..` go up to `alice` ⬆️

3️⃣ `Movies` go back down ⬇️

4️⃣ `.` stay in `Movies`

5️⃣ open `the_odyssey.mkv` 🎯

```
📁 /
└── 📁 home
    ├── 📁 alice                    <-- (2) .. (up)
    │   ├── 📁 Movies               <-- (1) . (3) back (4) . 📍
    │   │   └── 🎬 the_odyssey.mkv  <-- (5) open 🎯
    │   ├── 📁 Downloads
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob
        └── 📄 notes.txt
```

💡 **Tip:** `.` and `..` can appear *anywhere* in a path

🔗 **`pwd` + `/` + `relative`** `/home/alice/Movies/./../Movies/./the_odyssey.mkv`

✂️ drop `.`, each `..` removes a directory `/home/alice~~/Movies~~[1]~~/.~~~~/..~~[1]/Movies~~/.~~/the_odyssey.mkv`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `./../Movies/./the_odyssey.mkv`

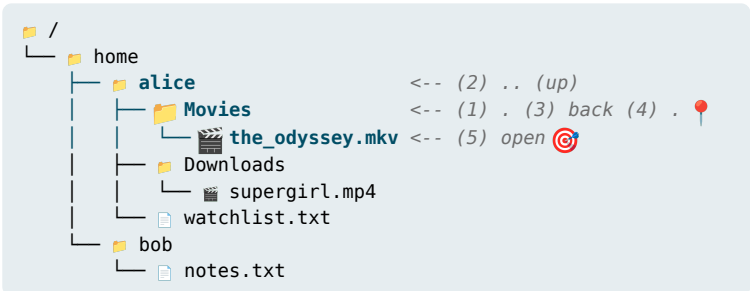
1️⃣ `.` stay in `Movies`

2️⃣ `..` go up to `alice` ⬆️

3️⃣ `Movies` go back down ⬇️

4️⃣ `.` stay in `Movies`

5️⃣ open `the_odyssey.mkv` 🎯



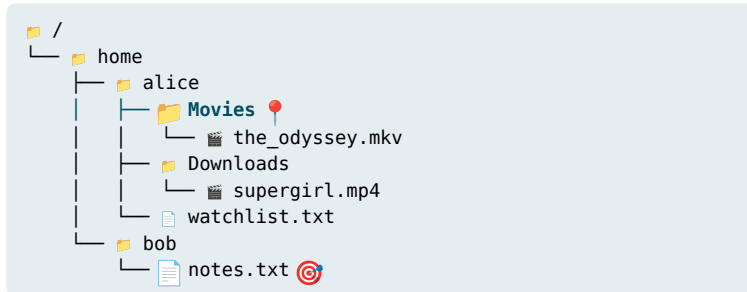
💡 **Tip:** `.` and `..` can appear *anywhere* in a path

🔗 `pwd` + `/` + `relative`	`/home/alice/Movies/./../Movies/./the_odyssey.mkv`
✂️ drop `.`, each `..` removes a directory	`/home/alice/Movies`[1] `/.` `/..`[1] `/Movies` `/.` `/the_odyssey.mkv`
🌍 absolute path	`/home/alice/Movies/the_odyssey.mkv`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/./../../bob/notes.txt`



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

◁ ●●●

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/./../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 **`../Downloads/.`**`/../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️

3️⃣ `.` stay in `Downloads`



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 **`../Downloads/./..`**`/../bob/notes.txt`

1. `..` go up to `alice` ⬆️
2. `Downloads` go down ⬇️
3. `.` stay in `Downloads`
4. `..` go back up to `alice` ⬆️



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 **`../Downloads/./../..`**`/bob/notes.txt`

1. `..` go up to `alice` ⬆️
2. `Downloads` go down ⬇️
3. `.` stay in `Downloads`
4. `..` go back up to `alice` ⬆️
5. `..` go up to `home` ⬆️

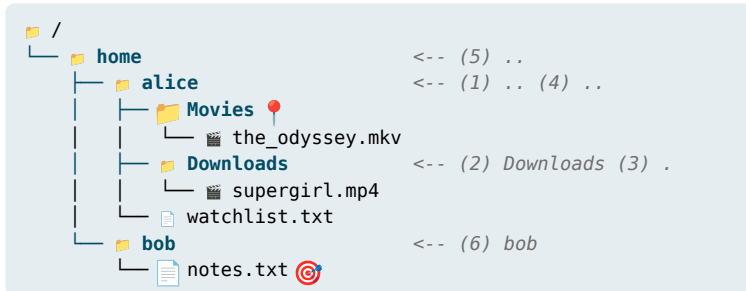


 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/./../../bob/notes.txt`

1. `..` go up to `alice` ⬆️
2. `Downloads` go down ⬇️
3. `.` stay in `Downloads`
4. `..` go back up to `alice` ⬆️
5. `..` go up to `home` ⬆️
6. `bob` go down ⬇️



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/./../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️

3️⃣ `.` stay in `Downloads`

4️⃣ `..` go back up to `alice` ⬆️

5️⃣ `..` go up to `home` ⬆️

6️⃣ `bob` go down ⬇️

7️⃣ open `notes.txt` 🎯



 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/./../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️

3️⃣ `.` stay in `Downloads`

4️⃣ `..` go back up to `alice` ⬆️

5️⃣ `..` go up to `home` ⬆️

6️⃣ `bob` go down ⬇️

7️⃣ open `notes.txt` 🎯



💡 **Tip:** `..` removes the last directory *still left* in the path

🔗 `pwd` + `/` + `relative` `/home/alice/Movies/../Downloads/./../../bob/notes.txt`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/./../../bob/notes.txt`

1. `..` go up to `alice` ⬆️
2. `Downloads` go down ⬇️
3. `.` stay in `Downloads`
4. `..` go back up to `alice` ⬆️
5. `..` go up to `home` ⬆️
6. `bob` go down ⬇️
7. open `notes.txt` 🎯

```
📁 /
└── 📁 home                          <-- (5) ..
    ├── 📁 alice                     <-- (1) .. (4) ..
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads             <-- (2) Downloads (3) .
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob                       <-- (6) bob
        └── 📄 notes.txt             <-- (7) open 🎯
```

💡 **Tip:** `..` removes the last directory *still left* in the path

🔗 `pwd` + `/` + `relative` → `/home/alice/Movies/../Downloads/./../../bob/notes.txt`

✂️ drop `.`, each `..` removes a directory → `/home/alice~~/Movies~~[1]~~/..~~[1]/Downloads/./../../bob/notes.txt`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/./../../bob/notes.txt`

1. `..` go up to `alice` ⬆️
2. `Downloads` go down ⬇️
3. `.` stay in `Downloads`
4. `..` go back up to `alice` ⬆️
5. `..` go up to `home` ⬆️
6. `bob` go down ⬇️
7. open `notes.txt` 🎯

```
📁 /
└── 📁 home                          <-- (5) ..
    ├── 📁 alice                     <-- (1) .. (4) ..
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads             <-- (2) Downloads (3) .
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob                       <-- (6) bob
        └── 📄 notes.txt             <-- (7) open 🎯
```

💡 **Tip:** `..` removes the last directory *still left* in the path

🔗 `pwd` + `/` + `relative` — `/home/alice/Movies/../Downloads/./../../bob/notes.txt`

✂️ drop `.`, each `..` removes a directory — `/home/alice~~/Movies~~[1]~~/..~~[1]/Downloads~~/.~~/../../bob/notes.txt`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/./../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️

3️⃣ `.` stay in `Downloads`

4️⃣ `..` go back up to `alice` ⬆️

5️⃣ `..` go up to `home` ⬆️

6️⃣ `bob` go down ⬇️

7️⃣ open `notes.txt` 🎯

```
📁 /
└── 📁 home                          <-- (5) ..
    ├── 📁 alice                     <-- (1) .. (4) ..
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads             <-- (2) Downloads (3) .
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob                       <-- (6) bob
        └── 📄 notes.txt             <-- (7) open 🎯
```

💡 **Tip:** `..` removes the last directory *still left* in the path

🔗 `pwd` + `/` + `relative` `/home/alice/Movies/../Downloads/./../../bob/notes.txt`

✂️ drop `.`, each `..` removes a directory `/home/alice~~/Movies~~[1] ~~/..~~[1] ~~/Downloads~~[2] ~~/.~~ ~~/..~~[2] /../bob/notes.txt`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/./../../bob/notes.txt`

1. `..` go up to `alice` ⬆️
2. `Downloads` go down ⬇️
3. `.` stay in `Downloads`
4. `..` go back up to `alice` ⬆️
5. `..` go up to `home` ⬆️
6. `bob` go down ⬇️
7. open `notes.txt` 🎯

```
📁 /
└── 📁 home                          <-- (5) ..
    ├── 📁 alice                     <-- (1) .. (4) ..
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads             <-- (2) Downloads (3) .
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob                       <-- (6) bob
        └── 📄 notes.txt             <-- (7) open 🎯
```

💡 **Tip:** `..` removes the last directory *still left* in the path

🔗 `pwd` + `/` + `relative` `/home/alice/Movies/../Downloads/./../../bob/notes.txt`

✂️ drop `.`, each `..` removes a directory `/home/~~alice~~[3] ~~/Movies~~[1] ~~/..~~[1] ~~/Downloads~~[2] ~~/.~~ ~~/..~~[2] ~~/..~~[3] /bob/notes.txt`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Tricky Paths

🗺️ from 📍 `/home/alice/Movies` walk 🧾 `../Downloads/./../../bob/notes.txt`

1️⃣ `..` go up to `alice` ⬆️

2️⃣ `Downloads` go down ⬇️

3️⃣ `.` stay in `Downloads`

4️⃣ `..` go back up to `alice` ⬆️

5️⃣ `..` go up to `home` ⬆️

6️⃣ `bob` go down ⬇️

7️⃣ open `notes.txt` 🎯

```
📁 /
└── 📁 home                          <-- (5) ..
    ├── 📁 alice                     <-- (1) .. (4) ..
    │   ├── 📁 Movies 📍
    │   │   └── 🎬 the_odyssey.mkv
    │   ├── 📁 Downloads             <-- (2) Downloads (3) .
    │   │   └── 🎬 supergirl.mp4
    │   └── 📄 watchlist.txt
    └── 📁 bob                       <-- (6) bob
        └── 📄 notes.txt             <-- (7) open 🎯
```

💡 **Tip:** `..` removes the last directory *still left* in the path

🔗 `pwd` + `/` + `relative`	`/home/alice/Movies/../Downloads/./../../bob/notes.txt`
✂️ drop `.`, each `..` removes a directory	`/home/~~alice~~[3]/~~Movies~~[1]/~~..~~[1]/~~Downloads~~[2]/~~.~~/~~..~~[2]/~~..~~[3]/bob/notes.txt`
🌍 absolute path	`/home/bob/notes.txt`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Manual

`man` *and paths as parameters*

Bibliography

for this section

1. **Brian Ward**, *How LINUX Works*, 3rd Edition, No Starch Press, 2021
   - Chapter 2 - *Basic Commands and Directory Hierarchy*
     - Section 2.13 - *Getting Online Help*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

man

📖 Manual Pages - `man pwd` (shortened to fit the slide)

```
PWD(1)                          User Commands                          PWD(1)

NAME
       pwd - print the name of the current working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Prints the absolute path of the current working directory,
       starting from the root directory /.
       -L, --logical
              keep the symbolic links that are part of the path
       -P, --physical
              replace symbolic links with the directories they point to
       --help show a short help text and exit

EXIT STATUS
       0 on success, any other value if an error occurred

EXAMPLES
       $ cd /home/alice/Movies && pwd
       /home/alice/Movies

SEE ALSO
       cd(1), ls(1)

shortened for this slide        2026                                   PWD(1)
```

📖 a manual page

```
$ man pwd
```

opens the manual page of `pwd`

every page has the **same parts**, in the same order

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

📖 Manual Pages - `man pwd` (shortened to fit the slide)

```
PWD(1)                          User Commands                          PWD(1)

NAME
       pwd - print the name of the current working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Prints the absolute path of the current working directory,
       starting from the root directory /.
       -L, --logical
              keep the symbolic links that are part of the path
       -P, --physical
              replace symbolic links with the directories they point to
       --help show a short help text and exit

EXIT STATUS
       0 on success, any other value if an error occurred

EXAMPLES
       $ cd /home/alice/Movies && pwd
       /home/alice/Movies

SEE ALSO
       cd(1), ls(1)

shortened for this slide        2026                           PWD(1)
```

🏷️ header and footer

`PWD(1)` the name and the *section* of the page

`User Commands` the name of section `1`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

man

📖 Manual Pages - `man pwd` (shortened to fit the slide)

```
PWD(1)                        User Commands                        PWD(1)

NAME
       pwd - print the name of the current working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Prints the absolute path of the current working directory,
       starting from the root directory /.
       -L, --logical
              keep the symbolic links that are part of the path
       -P, --physical
              replace symbolic links with the directories they point to
       --help show a short help text and exit

EXIT STATUS
       0 on success, any other value if an error occurred

EXAMPLES
       $ cd /home/alice/Movies && pwd
       /home/alice/Movies

SEE ALSO
       cd(1), ls(1)

shortened for this slide        2026                              PWD(1)
```

🪪 NAME

`pwd` and what it is, **in one line**

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

📖 Manual Pages - `man pwd` (shortened to fit the slide)

```
PWD(1)                        User Commands                        PWD(1)

NAME
       pwd - print the name of the current working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Prints the absolute path of the current working directory,
       starting from the root directory /.
       -L, --logical
              keep the symbolic links that are part of the path
       -P, --physical
              replace symbolic links with the directories they point to
       --help show a short help text and exit

EXIT STATUS
       0 on success, any other value if an error occurred

EXAMPLES
       $ cd /home/alice/Movies && pwd
       /home/alice/Movies

SEE ALSO
       cd(1), ls(1)

shortened for this slide        2026                               PWD(1)
```

✍ SYNOPSIS

how to call it

`[ ]` is optional

`...` can repeat

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

📖 Manual Pages - `man pwd` (shortened to fit the slide)

```
PWD(1)                          User Commands                          PWD(1)

NAME
       pwd - print the name of the current working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Prints the absolute path of the current working directory,
       starting from the root directory /.
       -L, --logical
              keep the symbolic links that are part of the path
       -P, --physical
              replace symbolic links with the directories they point to
       --help show a short help text and exit

EXIT STATUS
       0 on success, any other value if an error occurred

EXAMPLES
       $ cd /home/alice/Movies && pwd
       /home/alice/Movies

SEE ALSO
       cd(1), ls(1)

shortened for this slide        2026                          PWD(1)
```

📝 DESCRIPTION

what it does, in detail, and every option

`-P` a short option

`--physical` a long option, same meaning

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

man

📖 Manual Pages - `man pwd` (shortened to fit the slide)

```
PWD(1)                        User Commands                        PWD(1)

NAME
       pwd - print the name of the current working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Prints the absolute path of the current working directory,
       starting from the root directory /.
       -L, --logical
              keep the symbolic links that are part of the path
       -P, --physical
              replace symbolic links with the directories they point to
       --help show a short help text and exit

EXIT STATUS
       0 on success, any other value if an error occurred

EXAMPLES
       $ cd /home/alice/Movies && pwd
       /home/alice/Movies

SEE ALSO
       cd(1), ls(1)

shortened for this slide        2026                                PWD(1)
```

🚦 EXIT STATUS

`0` means **success**

anything else is an *error*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

man

📖 Manual Pages - `man pwd` (shortened to fit the slide)

```
PWD(1)                        User Commands                        PWD(1)

NAME
       pwd - print the name of the current working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Prints the absolute path of the current working directory,
       starting from the root directory /.
       -L, --logical
              keep the symbolic links that are part of the path
       -P, --physical
              replace symbolic links with the directories they point to
       --help show a short help text and exit

EXIT STATUS
       0 on success, any other value if an error occurred

EXAMPLES
       $ cd /home/alice/Movies && pwd
       /home/alice/Movies

SEE ALSO
       cd(1), ls(1)

shortened for this slide        2026                               PWD(1)
```

🧪 EXAMPLES

`$ ...` a command to try

`/home/...` what it prints

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

man

📖 Manual Pages - `man pwd` (shortened to fit the slide)

```
PWD(1)                        User Commands                        PWD(1)

NAME
       pwd - print the name of the current working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Prints the absolute path of the current working directory,
       starting from the root directory /.
       -L, --logical
              keep the symbolic links that are part of the path
       -P, --physical
              replace symbolic links with the directories they point to
       --help show a short help text and exit

EXIT STATUS
       0 on success, any other value if an error occurred

EXAMPLES
       $ cd /home/alice/Movies && pwd
       /home/alice/Movies

SEE ALSO
       cd(1), ls(1)

shortened for this slide        2026                               PWD(1)
```

🔗 SEE ALSO

related pages

`cd(1)` the page of `cd`, section `1`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

man

📖 Manual Pages - `man pwd` (shortened to fit the slide)

```
PWD(1)                          User Commands                          PWD(1)

NAME
       pwd - print the name of the current working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Prints the absolute path of the current working directory,
       starting from the root directory /.
       -L, --logical
              keep the symbolic links that are part of the path
       -P, --physical
              replace symbolic links with the directories they point to
       --help show a short help text and exit

EXIT STATUS
       0 on success, any other value if an error occurred

EXAMPLES
       $ cd /home/alice/Movies && pwd
       /home/alice/Movies

SEE ALSO
       cd(1), ls(1)

shortened for this slide        2026                                   PWD(1)
```

⌨ inside the viewer

↑ ↓ Space scroll

/ *word* Enter search

q quit

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Inside `man`



⌨ The Keys of the Manual Viewer

↕ move in the page

↑ ↓	one line up / down
Page Down Space	one page down
Page Up b	one page up
g G	the beginning / the end
h	help, with all the keys
q	quit

🔍 search in the page

/ *word* Enter	search forward
? *word* Enter	search backward
n	next match
N	previous match

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Inside `man`

⌨️ The Keys of the Manual Viewer

↕ move in the page

↑ ↓	one line up / down
Page Down Space	one page down
Page Up b	one page up
g G	the beginning / the end
h	help, with all the keys
q	quit

🔍 search in the page

/ *word* Enter	search forward
? *word* Enter	search backward
n	next match
N	previous match

💡 search for *what you want to do*, not for the option

`man ls`, then / `reverse` Enter jumps to `-r, --reverse` *reverse order while sorting*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Files are Paths



📂 commands receive paths, not just names

```
/
└── home
    └── alice
        ├── Downloads
        ├── Movies 📍
        │   └── the_odyssey.mkv
        └── watchlist.txt
```

we are in `/home/alice/Movies`

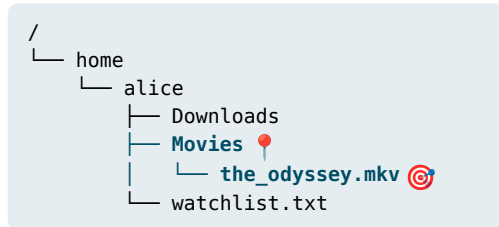
When a command expects a **file** or a **directory**, it accepts **any path** to it

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Files are Paths

📂 commands receive paths, not just names

🏷️ a **name** `the_odyssey.mkv`



in the current directory

When a command expects a **file** or a **directory**, it accepts **any path** to it

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Files are Paths



📂 commands receive paths, not just names

- 🏷️ a **name** `the_odyssey.mkv`
- ↩️ a **relative path** `../watchlist.txt`

```
/
└── home
    └── alice
        ├── Downloads
        ├── Movies 📍
        │   └── the_odyssey.mkv
        └── watchlist.txt 🎯
```

from the current directory

When a command expects a **file** or a **directory**, it accepts **any path** to it

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Files are Paths



📂 commands receive paths, not just names

- 🏷️ a **name** `the_odyssey.mkv`
- ↩️ a **relative path** `../watchlist.txt`
- 🌍 an **absolute path** `/home/alice/watchlist.txt`

```
/
└── home
    └── alice
        ├── Downloads
        ├── Movies 📍
        │   └── the_odyssey.mkv
        └── watchlist.txt 🎯
```

from the root `/`

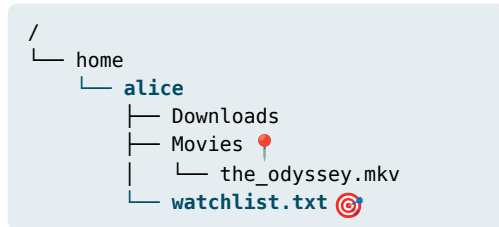
When a command expects a **file** or a **directory**, it accepts **any path** to it

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Files are Paths

📂 commands receive paths, not just names

- 🏷️ a **name** `the_odyssey.mkv`
- ↩️ a **relative path** `../watchlist.txt`
- 🌍 an **absolute path** `/home/alice/watchlist.txt`
- 🏠 a path with `~` `~/watchlist.txt`



from the home directory `/home/alice`

When a command expects a **file** or a **directory**, it accepts **any path** to it

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Same File, Many Paths

🎯 all these commands open the **same file**

```
$ cat ../watchlist.txt
Project Hail Mary
```

💡 `cat` prints the content of a file

```
/
└── home
    └── alice
        ├── Downloads
        ├── Movies 📍
        │   └── the_odyssey.mkv
        └── watchlist.txt 🎯
```

↩️ relative path

joined to `/home/alice/Movies`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Same File, Many Paths

🎯 all these commands open the **same file**

```
$ cat ../watchlist.txt
Project Hail Mary
$ cat /home/alice/watchlist.txt
Project Hail Mary
```

💡 `cat` prints the content of a file

```
/
└── home
    └── alice
        ├── Downloads
        ├── Movies 📍
        │   └── the_odyssey.mkv
        └── watchlist.txt 🎯
```

🌍 absolute path

starts from the root `/`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Same File, Many Paths

🎯 all these commands open the **same file**

```
$ cat ../watchlist.txt
Project Hail Mary
$ cat /home/alice/watchlist.txt
Project Hail Mary
$ cat ~/watchlist.txt
Project Hail Mary
```

💡 `cat` prints the content of a file

```
/
└── home
    └── alice
        ├── Downloads
        ├── Movies 📍
        │   └── the_odyssey.mkv
        └── watchlist.txt 🎯
```

🏠 home directory

`~` is `/home/alice`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Same File, Many Paths

🎯 all these commands open the **same file**

```
$ cat ../watchlist.txt
Project Hail Mary
$ cat /home/alice/watchlist.txt
Project Hail Mary
$ cat ~/watchlist.txt
Project Hail Mary
$ cat ../../alice/Downloads/../watchlist.txt
Project Hail Mary
```

💡 `cat` prints the content of a file

```
/
└── home
    └── alice
        ├── Downloads
        ├── Movies 📍
        │   └── the_odyssey.mkv
        └── watchlist.txt 🎯
```

🌀 a detour

through `Downloads` and back

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Paths in the Manual



📖 usually `FILE`, `DIR`, `SOURCE`, `DEST` is a path

```
SYNOPSIS
CD(1)     cd [DIR]
LS(1)     ls [OPTION]... [FILE]...
CP(1)     cp [OPTION]... SOURCE DEST
MV(1)     mv [OPTION]... SOURCE DEST
MKDIR(1)  mkdir [OPTION]... DIRECTORY...
RM(1)     rm [OPTION]... [FILE]...
```

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Paths in the Manual

📖 usually `FILE`, `DIR`, `SOURCE`, `DEST` is a path

```
SYNOPSIS
CD(1)      cd [DIR]
LS(1)      ls [OPTION]... [FILE]...
CP(1)      cp [OPTION]... SOURCE DEST
MV(1)      mv [OPTION]... SOURCE DEST
MKDIR(1)   mkdir [OPTION]... DIRECTORY...
RM(1)      rm [OPTION]... [FILE]...
```

relative and absolute paths can be **mixed** in the same command

```
$ cp ../watchlist.txt /tmp/
```

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Paths in the Manual

📖 usually `FILE`, `DIR`, `SOURCE`, `DEST` is a path

```
SYNOPSIS
CD(1)      cd [DIR]
LS(1)      ls [OPTION]... [FILE]...
CP(1)      cp [OPTION]... SOURCE DEST
MV(1)      mv [OPTION]... SOURCE DEST
MKDIR(1)   mkdir [OPTION]... DIRECTORY...
RM(1)      rm [OPTION]... [FILE]...
```

relative and absolute paths can be **mixed** in the same command

```
$ cp ../watchlist.txt /tmp/
$ mv /tmp/watchlist.txt ~/Downloads/
```

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Paths in the Manual

📖 usually `FILE`, `DIR`, `SOURCE`, `DEST` is a path

```
SYNOPSIS
CD(1)      cd [DIR]
LS(1)      ls [OPTION]... [FILE]...
CP(1)      cp [OPTION]... SOURCE DEST
MV(1)      mv [OPTION]... SOURCE DEST
MKDIR(1)   mkdir [OPTION]... DIRECTORY...
RM(1)      rm [OPTION]... [FILE]...
```

relative and absolute paths can be **mixed** in the same command

```
$ cp ../watchlist.txt /tmp/
$ mv /tmp/watchlist.txt ~/Downloads/
$ ls -l ~/Documents ../Downloads
```

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Paths in the Manual

📖 usually `FILE`, `DIR`, `SOURCE`, `DEST` is a path

```
SYNOPSIS
CD(1)       cd [DIR]
LS(1)       ls [OPTION]... [FILE]...
CP(1)       cp [OPTION]... SOURCE DEST
MV(1)       mv [OPTION]... SOURCE DEST
MKDIR(1)    mkdir [OPTION]... DIRECTORY...
RM(1)       rm [OPTION]... [FILE]...
```

relative and absolute paths can be **mixed** in the same command

```
$ cp ../watchlist.txt /tmp/
$ mv /tmp/watchlist.txt ~/Downloads/
$ ls -l ~/Documents ../Downloads
```

⚠ relative paths

a **relative** path depends on *where you are* when you run the command

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

⚠ *read the manual*

always read the manual **before using AI and searching online** for commands that you want to use

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

⚠️ *read the manual*

always read the manual **before using AI and searching online** for commands that you want to use

1️⃣ the manual

```
$ man ls
```

written for the **version installed** on your computer

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

⚠️ *read the manual*

always read the manual **before using AI and searching online** for commands that you want to use

1️⃣ the manual

```
$ man ls
```

written for the **version installed** on your computer

2️⃣ the short help

```
$ ls --help
```

a summary of the options, for most commands

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

⚠️ *read the manual*

always read the manual **before using AI and searching online** for commands that you want to use

1️⃣ the manual

```
$ man ls
```

written for the **version installed** on your computer

2️⃣ the short help

```
$ ls --help
```

a summary of the options, for most commands

3️⃣ search online, ask AI

🔎 web 🤖 AI

might be for **another version** or **wrong**, check the manual

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

Navigation

`pwd`, `cd`, `ls`, `tree` and `find`

Bibliography

for this section

1. **Brian Ward**, *How LINUX Works*, 3rd Edition, No Starch Press, 2021
   - Chapter 2 - *Basic Commands and Directory Hierarchy*
     - Section 2.3.1 - `ls`
     - Section 2.4.1 - `cd`
     - Section 2.5.3 - `pwd`
     - Section 2.5.6 - `find`
2. **Razvan Deaconescu, Razvan Rughinis, Mihai Carabas, Alexandru Radovici**, *Utilizarea Sistemelor de Operare*, Printech 2021, download
   - Chapter 2 - *Utilizarea sistemului de fișiere*
     - Section 2.3.1 - *Afișarea și schimbarea directorului curent*
     - Section 2.3.2 - *Listarea fișierelor*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

pwd

Print Working Directory - 📍

Shows the **absolute path** of your **current directory**

```
$ pwd
```

```
[alice@computer: ~/Movies] $ pwd
/home/alice/Movies
```

```
/
└── home
    └── alice
        ├── Movies 📍
        └── Downloads
```

pwd prints the path from / to 📍

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

pwd

Print Working Directory - 📍

Shows the **absolute path** of your **current directory**

```
$ pwd
```

```
[alice@computer: ~/Movies] $ pwd
/home/alice/Movies
```

🤔 mind the prompt `[alice@computer: ~/Movies] $`
it already shows the current directory

```
/
└── home
    └── alice
        ├── Movies 📍
        └── Downloads
```

`pwd` prints the path from `/` to 📍

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

cd

📁 Change (Current) Directory

```
$ cd [directory]    # `[directory]` means that the `directory` parameter is optional
```

```
$ cd /home/alice/Downloads  # absolute path
```

```
/
└── home
    └── alice
        ├── Movies
        └── Downloads 📍
```

absolute: start from `/`, go down

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

cd



📁 Change (Current) Directory

```
$ cd [directory]    # `[directory]` means that the `directory` parameter is optional
```

```
$ cd /home/alice/Downloads  # absolute path
$ cd ../Movies              # relative path
```

```
/
└── home
    └── alice
        ├── Movies 📍
        └── Downloads
```

`..` *goes up to* `alice`*, then into* `Movies`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

cd

📁 Change (Current) Directory

```
$ cd [directory]    # `[directory]` means that the `directory` parameter is optional
```

```
$ cd /home/alice/Downloads  # absolute path
$ cd ../Movies              # relative path
$ cd -                      # previous directory
/home/alice/Downloads
```

```
/
└── home
    └── alice
        ├── Movies
        └── Downloads 📍
```

`cd -` goes to `Downloads`, where we were before, and prints its path

🔁 `cd -` goes where you were **before** the last `cd`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`cd`

📁 Change (Current) Directory

```
$ cd [directory]    # `[directory]` means that the `directory` parameter is optional
```

```
$ cd /home/alice/Downloads  # absolute path
$ cd ../Movies              # relative path
$ cd -                      # previous directory
/home/alice/Downloads
$ cd -
/home/alice/Movies
```

```
/
└── home
    └── alice
        ├── Movies 📍
        └── Downloads
```

`cd -` again goes to `Movies`, not to the directory before `Downloads`

🔁 `cd -` goes where you were **before** the last `cd`

⚠️ `cd -` is **not** "back": it switches between two directories

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

cd



📁 Change (Current) Directory

```
$ cd [directory]    # `[directory]` means that the `directory` parameter is optional
```

```
$ cd /home/alice/Downloads  # absolute path
$ cd ../Movies              # relative path
$ cd -                      # previous directory
/home/alice/Downloads
$ cd -
/home/alice/Movies
$ cd ~                      # home directory
```

```
/
└── home
    └── alice 📍
        ├── Movies
        └── Downloads
```

`~` *is the home directory of the user,* `/home/alice`

🔁 `cd -` goes where you were **before** the last `cd`

⚠️ `cd -` is **not** "back": it switches between two directories

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

cd

📁 Change (Current) Directory

```
$ cd [directory]    # `[directory]` means that the `directory` parameter is optional
```

```
$ cd /home/alice/Downloads  # absolute path
$ cd ../Movies              # relative path
$ cd -                      # previous directory
/home/alice/Downloads
$ cd -
/home/alice/Movies
$ cd ~                      # home directory
$ cd                        # home directory
```

```
/
└── home
    └── alice 📍
        ├── Movies
        └── Downloads
```

no parameter, go to the home directory

🔁 `cd -` goes where you were **before** the last `cd`

⚠️ `cd -` is **not** "back": it switches between two directories

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`ls`

📂 Listing Files and Directories

```
$ ls [option]... [file]...
```

```
$ ls
Downloads  Movies  watchlist.txt
```

```
~ (/home/alice) 📍
├── .bashrc
├── Downloads
│   └── supergirl.mp4
├── Movies
│   └── the_odyssey.mkv
└── watchlist.txt
```

only the entries of the directory,
not what is inside `Movies`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

ls

📂 Listing Files and Directories

```
$ ls [option]... [file]...
```

```
$ ls
Downloads  Movies  watchlist.txt
$ ls watchlist.txt Movies
watchlist.txt

Movies:
the_odyssey.mkv
```

```
~ (/home/alice) 📍
├── .bashrc
├── Downloads
│   └── supergirl.mp4
├── Movies
│   └── the_odyssey.mkv
└── watchlist.txt
```

a file is listed by its name,
a directory by what is inside *it*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`ls`

📂 Listing Files and Directories

```
$ ls [option]... [file]...
```

```
$ ls
Downloads  Movies  watchlist.txt
$ ls watchlist.txt Movies
watchlist.txt

Movies:
the_odyssey.mkv
$ ls -a
.  ..  .bashrc  Downloads  Movies  watchlist.txt
```

```
~ (/home/alice) 📍
├── .bashrc
├── Downloads
│   └── supergirl.mp4
├── Movies
│   └── the_odyssey.mkv
└── watchlist.txt
```

names starting with `.` *are* hidden

`-a` - include hidden files (`.` *files*)

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

ls

📂 Listing Files and Directories

```
$ ls [option]... [file]...
```

```
$ ls
Downloads  Movies  watchlist.txt
$ ls watchlist.txt Movies
watchlist.txt

Movies:
the_odyssey.mkv
$ ls -a
.  ..  .bashrc  Downloads  Movies  watchlist.txt
$ ls -l -h ~
drwxr-xr-x  2 alice alice 4.0K Sep 20 18:02 Downloads
drwxr-xr-x  2 alice alice 4.0K Sep 26 21:15 Movies
-rw-r--r--  1 alice alice  21K Sep 28 21:40 watchlist.txt
```

```
~ (/home/alice) 📍
├── .bashrc
├── Downloads        d
│   └── supergirl.mp4
├── Movies           d
│   └── the_odyssey.mkv
└── watchlist.txt    -
```

1st letter: `d` directory, `-` file, sizes in `K`, `M`, `G` instead of bytes

`-a` - include hidden files (`.` *files*)

`-l` - long listing, `-h` - human-readable sizes

💡 `ls -l -h ~` is the same as `ls -lh ~`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`tree`

🌳 List a Directory and Everything Inside

```
$ tree [option]... [directory]...
```

```
$ tree
.
├── Downloads
│   └── supergirl.mp4
├── Movies
│   └── the_odyssey.mkv
└── watchlist.txt

3 directories, 3 files
```

`ls` lists *one* directory, `tree` also goes **inside** every directory

📦 not installed by default? Ubuntu: `sudo apt install tree` Fedora: `sudo dnf install tree`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

tree

🌳 List a Directory and Everything Inside

```
$ tree [option]... [directory]...
```

```
$ tree Movies Downloads
Movies
└── the_odyssey.mkv
Downloads
└── supergirl.mp4

2 directories, 2 files
```

`ls` lists *one* directory, `tree` also goes **inside** every directory

`Movies Downloads` - only what is inside these directories

📦 not installed by default? Ubuntu: `sudo apt install tree` Fedora: `sudo dnf install tree`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`tree`

🌳 List a Directory and Everything Inside

```
$ tree [option]... [directory]...
```

```
$ tree -L 1
.
├── Downloads
├── Movies
└── watchlist.txt

3 directories, 1 file
```

`ls` lists *one* directory, `tree` also goes **inside** every directory

`Movies Downloads` - only what is inside these directories

`-L 1` - only 1 level deep

📦 not installed by default? Ubuntu: `sudo apt install tree` Fedora: `sudo dnf install tree`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

tree

🌳 List a Directory and Everything Inside

```
$ tree [option]... [directory]...
```

```
$ tree -d
.
├── Downloads
└── Movies

3 directories
```

`ls` lists *one* directory, `tree` also goes **inside** every directory

`Movies Downloads` - only what is inside these directories

`-L 1` - only 1 level deep

`-d` - only directories

📦 not installed by default? Ubuntu: `sudo apt install tree` Fedora: `sudo dnf install tree`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

tree

🌳 List a Directory and Everything Inside

```
$ tree [option]... [directory]...
```

```
$ tree -a -L 1 ~
/home/alice
├── .bashrc
├── Downloads
├── Movies
└── watchlist.txt

3 directories, 2 files
```

`ls` lists *one* directory, `tree` also goes **inside** every directory

`Movies Downloads` - only what is inside these directories

`-L 1` - only 1 level deep

`-d` - only directories

`-a` - include hidden files, `~` - list the home directory

📦 not installed by default? Ubuntu: `sudo apt install tree` Fedora: `sudo dnf install tree`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

find

🔍 Search for Files and Directories

```
$ find [directory]... [filter]...
```

```
$ find Movies
Movies
Movies/the_odyssey.mkv
```

```
~ (/home/alice) 📍
├── Downloads
│   └── supergirl.mp4
├── Movies
│   └── the_odyssey.mkv
└── watchlist.txt
```

the directory and everything inside it

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

find

🔍 Search for Files and Directories

```
$ find [directory]... [filter]...
```

```
$ find . -name "*.mkv"
./Movies/the_odyssey.mkv
```

`-name` - by name, `*` means *any characters*

```
~ (/home/alice) 📍
├── Downloads
│   └── supergirl.mp4
├── Movies
│   └── the_odyssey.mkv
└── watchlist.txt
```

search all of `.`, keep the names that end with `.mkv`

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`find`

🔍 Search for Files and Directories

```
$ find [directory]... [filter]...
```

```
$ find Movies Downloads -type f
Movies/the_odyssey.mkv
Downloads/supergirl.mp4
```

`-name` - by name, `*` means *any characters*

`-type f` - only files, `-type d` - only directories

```
~ (/home/alice) 📍
├── Downloads
│   └── supergirl.mp4
├── Movies
│   └── the_odyssey.mkv
└── watchlist.txt
```

only the files, in both directories

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

find

🔍 Search for Files and Directories

```
$ find [directory]... [filter]...
```

```
$ find ~ -type f -name "*.txt"
/home/alice/watchlist.txt
```

`-name` - by name, `*` means *any characters*

`-type f` - only files, `-type d` - only directories

several filters: a result must match **all** of them

```
~ (/home/alice) 📍
├── Downloads
│   └── supergirl.mp4
├── Movies
│   └── the_odyssey.mkv
└── watchlist.txt
```

start from an absolute path, get absolute paths

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

File Management

`cp`, `mv`, `mkdir`, `touch`, `nano`, `rmdir` and `rm`

Bibliography

for this section

1. **Brian Ward**, *How LINUX Works*, 3rd Edition, No Starch Press, 2021
   - Chapter 2 - *Basic Commands and Directory Hierarchy*
     - Section 2.3 - *Basic Commands* (`cp`, `mv`, `touch`, `rm`)
     - Section 2.4 - *Navigating Directories* (`mkdir`, `rmdir`)
     - Section 2.12 - *Text Editors*
2. **Razvan Deaconescu, Razvan Rughinis, Mihai Carabas, Alexandru Radovici**, *Utilizarea Sistemelor de Operare*, Printech 2021, download
   - Chapter 2 - *Utilizarea sistemului de fișiere*
     - Section 2.3.4 - *Crearea fișierelor/directoarelor*
     - Section 2.3.5 - *Copiere / mutare / redenumire / ștergere*

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

cp

📋 Copy Files and Directories

```
$ cp [option]... source destination_file_or_directory
$ cp [option]... source... destination_directory
```

```
$ cp watchlist.txt /home/alice/backup/
```

```
/home/alice 📍
├── watchlist.txt
├── Movies/
│   ├── the_odyssey.mkv
│   └── toy_story_5.mkv
└── backup/
    └── watchlist.txt    +
```

copy a file into a directory

the source is **not changed**, a new copy is created

📍 current directory source + created - ~~removed~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

cp

📋 Copy Files and Directories

```
$ cp [option]... source destination_file_or_directory
$ cp [option]... source... destination_directory
```

```
$ cp watchlist.txt /home/alice/backup/
$ cp Movies/the_odyssey.mkv Movies/toy_story_5.mkv backup/
```

```
/home/alice 📍
├── watchlist.txt
├── Movies/
│   ├── the_odyssey.mkv
│   └── toy_story_5.mkv
└── backup/
    ├── watchlist.txt
    ├── the_odyssey.mkv  +
    └── toy_story_5.mkv  +
```

the source is **not changed**, a new copy is created

📍 current directory source + created ~~– removed~~

*several sources: the **last** parameter must be a **directory***

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

cp

📋 Copy Files and Directories

```
$ cp [option]... source destination_file_or_directory
$ cp [option]... source... destination_directory
```

```
$ cp watchlist.txt /home/alice/backup/
$ cp Movies/the_odyssey.mkv Movies/toy_story_5.mkv backup/
$ cp -r Movies/ /home/alice/backup/
```

```
/home/alice 📍
├── watchlist.txt
├── Movies/
│   ├── the_odyssey.mkv
│   └── toy_story_5.mkv
└── backup/
    ├── watchlist.txt
    ├── the_odyssey.mkv
    ├── toy_story_5.mkv
    └── Movies/                +
        ├── the_odyssey.mkv    +
        └── toy_story_5.mkv    +
```

`-r` copies the directory and everything inside

💡 without `-r`, `cp` refuses to copy directories

the source is **not changed**, a new copy is created

📍 current directory **source** **+ created** **~~- removed~~**

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

mv

📦 Move or Rename Files

```
$ mv [option]... source destination
$ mv [option]... source... destination_directory
```

```
$ mv movie.mkv project_hail_mary.mkv
```

```
/home/alice 📍
├── ~~movie.mkv~~                -
├── project_hail_mary.mkv  +
├── supergirl.mp4
├── michael.mp4
└── Movies/
    └── the_odyssey.mkv
```

rename a file

💡 renaming is moving to a new name in the same directory

📍 current directory + **created** - ~~removed~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

mv

📦 Move or Rename Files

```
$ mv [option]... source destination
$ mv [option]... source... destination_directory
```

```
$ mv movie.mkv project_hail_mary.mkv
$ mv project_hail_mary.mkv Movies/
```

💡 renaming is moving to a new name in the same directory

📍 current directory + created - ~~removed~~

```
/home/alice 📍
├── ~~project_hail_mary.mkv~~        -
├── supergirl.mp4
├── michael.mp4
└── Movies/
    ├── the_odyssey.mkv
    └── project_hail_mary.mkv  +
```

move a file to a directory

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

mv

📦 Move or Rename Files

```
$ mv [option]... source destination
$ mv [option]... source... destination_directory
```

```
$ mv movie.mkv project_hail_mary.mkv
$ mv project_hail_mary.mkv Movies/
$ mv supergirl.mp4 michael.mp4 Movies/
```

💡 renaming is moving to a new name in the same directory

📍 current directory + created ~~- removed~~

```
/home/alice 📍
├── supergirl.mp4                -
├── michael.mp4                  -
└── Movies/
    ├── the_odyssey.mkv
    ├── project_hail_mary.mkv
    ├── supergirl.mp4            +
    └── michael.mp4              +
```

move several files to a directory

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

mv

📦 Move or Rename Files

```
$ mv [option]... source destination
$ mv [option]... source... destination_directory
```

```
$ mv movie.mkv project_hail_mary.mkv
$ mv project_hail_mary.mkv Movies/
$ mv supergirl.mp4 michael.mp4 Movies/
$ mv Movies/ Movies_2026/
```

💡 renaming is moving to a new name in the same directory

📍 current directory + created - ~~removed~~

```
/home/alice 📍
├── Movies/                    -
└── Movies_2026/               +
    ├── the_odyssey.mkv
    ├── project_hail_mary.mkv
    ├── supergirl.mp4
    └── michael.mp4
```

rename a directory, the contents stay inside

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

mkdir

📁 Create Directories

```
$ mkdir [option]... directory...
```

```
$ mkdir Movies
```

```
/home/alice 📍
├── Movies/          +
└── watchlist.txt
```

make a directory

📍 current directory + created – ~~removed~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

mkdir

📁 Create Directories

```
$ mkdir [option]... directory...
```

```
$ mkdir Movies
$ mkdir -p Movies/2026/animation
```

💡 `-p` also creates the missing *parent* directories

📍 current directory + **created** ~~– removed~~

```
/home/alice 📍
├── Movies/
│   └── 2026/              +
│       └── animation/     +
└── watchlist.txt
```

`Movies/` exists, only `2026/animation/` is made

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`touch` and `nano`

📄 Create Files

```
$ touch [option]... file...
$ nano [option]... [file]
```

```
$ touch notes.txt
```

```
/home/alice 📍
├── Movies/
├── notes.txt      +
└── watchlist.txt
```

`touch` *makes an* ***empty*** *file*

💡 `touch` on a file that exists only updates its *date*

📍 current directory **+ created** ~~- removed~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`touch` and `nano`

📄 Create Files

```
$ touch [option]... file...
$ nano [option]... [file]
```

```
$ touch notes.txt
$ ls -l notes.txt
-rw-r--r-- 1 alice alice 0 Sep 30 14:20 notes.txt
```

```
/home/alice 📍
├── Movies/
├── notes.txt
└── watchlist.txt
```

size `0`: there is nothing inside

💡 `touch` on a file that exists only updates its *date*

📍 current directory + **created** - ~~removed~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`touch` and `nano`

📄 Create Files

```
$ touch [option]... file...
$ nano [option]... [file]
```

```
$ touch notes.txt
$ ls -l notes.txt
-rw-r--r-- 1 alice alice 0 Sep 30 14:20 notes.txt
$ nano todo.txt
```

✏️ inside `nano`: `Ctrl` + `O` save, `Ctrl` + `X` exit

📍 current directory **+ created** ~~- removed~~

```
/home/alice 📍
├── Movies/
├── notes.txt
├── todo.txt        +
└── watchlist.txt
```

`nano` opens a text editor,
*the file is made when you **save***

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`rm` and `rmdir`



🗑️ Remove Files and Directories

`rm` - deletes files or directories (⚠️ **destructive**!)

```
$ rm [option]... file...
```

`rmdir` - deletes only **empty directories**

```
$ rmdir [option]... directory...
```

```
$ rm dune_3_teaser.mp4
```

```
/home/alice 📍
├── ~~dune_3_teaser.mp4~~                    -
├── trailer1.mp4
├── trailer2.mp4
├── Trailers/
│   ├── avatar_fire_and_ash.mp4
│   └── the_odyssey.mp4
└── Watched/
```

delete a file

📍 current directory **+ created** ~~**- removed**~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`rm` and `rmdir`

🗑️ Remove Files and Directories

`rm` - deletes files or directories (⚠️ **destructive**!)

```
$ rm [option]... file...
```

`rmdir` - deletes only **empty directories**

```
$ rmdir [option]... directory...
```

```
$ rm dune_3_teaser.mp4
$ rm trailer1.mp4 trailer2.mp4
```

```
/home/alice 📍
├── ~~trailer1.mp4~~                    -
├── ~~trailer2.mp4~~                    -
├── Trailers/
│   ├── avatar_fire_and_ash.mp4
│   └── the_odyssey.mp4
└── Watched/
```

delete several files

📍 current directory + **created** - ~~removed~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`rm` and `rmdir`

🗑️ Remove Files and Directories

`rm` - deletes files or directories (⚠️ **destructive**!)

```
$ rm [option]... file...
```

`rmdir` - deletes only **empty directories**

```
$ rmdir [option]... directory...
```

```
$ rm dune_3_teaser.mp4
$ rm trailer1.mp4 trailer2.mp4
$ rm -r Trailers/
```

```
/home/alice 📍
├── ~~Trailers/~~                          -
│   ├── ~~avatar_fire_and_ash.mp4~~        -
│   └── ~~the_odyssey.mp4~~                -
└── Watched/
```

`-r` deletes the directory and everything inside

💡 `rmdir Trailers` fails, `Trailers` is not empty

📍 current directory **+ created** ~~- removed~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`rm` and `rmdir`

🗑️ Remove Files and Directories

`rm` - deletes files or directories (⚠️ **destructive**!)

```
$ rm [option]... file...
```

`rmdir` - deletes only **empty directories**

```
$ rmdir [option]... directory...
```

```
$ rm dune_3_teaser.mp4
$ rm trailer1.mp4 trailer2.mp4
$ rm -r Trailers/
$ rmdir Watched
```

```
/home/alice 📍
└── ~~Watched/~~  -
```

delete an empty directory

💡 `rmdir Trailers` fails, `Trailers` is not empty

📍 current directory **+ created** **- ~~removed~~**

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

`rm` and `rmdir`

🗑️ Remove Files and Directories

`rm` - deletes files or directories (⚠️ **destructive**!)

```
$ rm [option]... file...
```

`rmdir` - deletes only **empty directories**

```
$ rmdir [option]... directory...
```

```
$ rm dune_3_teaser.mp4
$ rm trailer1.mp4 trailer2.mp4
$ rm -r Trailers/
$ rmdir Watched
```

```
/home/alice 📍
└── ~~Watched/~~  -
```

delete an empty directory

💡 `rmdir Trailers` fails, `Trailers` is not empty

⚠️ there is no *trash*, deleting is **permanent**!

📍 current directory **+ created** ~~- removed~~

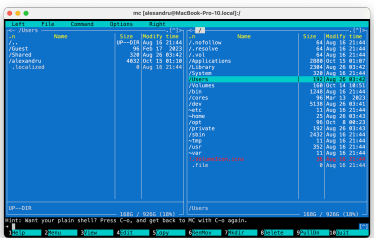
 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

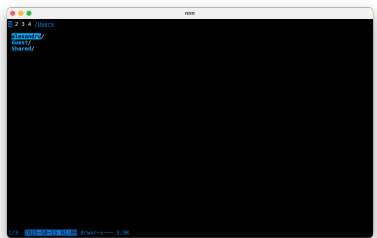
File Managers

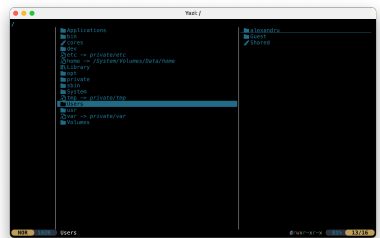
`mc`, `nnn` *and* `yazi`

File Management Software

- Midnight Commander (`mc`)
- n³ - *The unorthodox terminal file manager* (`nnn`)
- Yazi - ⚡ *Blazing fast terminal file manager written in Rust, based on async I/O.* (`yazi`)





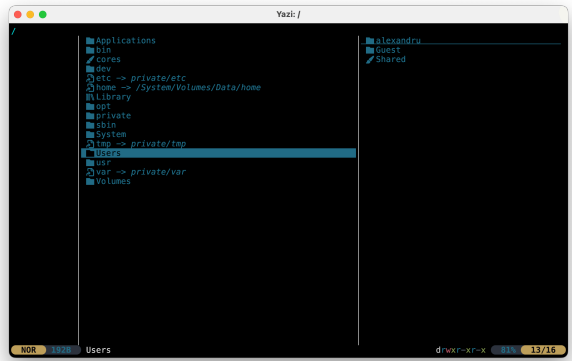


 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

yazi

🦆 a terminal file manager: the file commands with a few keys

key	action	like
move		
↑ ↓	choose a file	
← / →	parent directory / open	`cd`
`.`	show hidden files	`ls -a`
manage files		
Space	select, for several files	
`a`	create a file, `name/` for a directory	`touch`, `mkdir`
`r`	rename	`mv`
`y` then `p`	copy, then paste	`cp`
`x` then `p`	cut, then paste	`mv`
`d`	move to the trash	
`D`	delete permanently	`rm`
tabs		
`t` `t`	open a new tab	
`1` ... `9`	go to a tab	
`[` `]`	previous / next tab	



run `yazi`, `q` to quit, `~` for help

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

yazi



a terminal file manager: the file commands with a few keys

key	action	like
move		
↑ ↓	choose a file	
← / →	parent directory / open	`cd`
`.`	show hidden files	`ls -a`
manage files		
Space	select, for several files	
`a`	create a file, `name/` for a directory	`touch`, `mkdir`
`r`	rename	`mv`
`y` then `p`	copy, then paste	`cp`
`x` then `p`	cut, then paste	`mv`
`d`	move to the trash	
`D`	delete permanently	`rm`
tabs		
`t` `t`	open a new tab	
`1` ... `9`	go to a tab	
`[` `]`	previous / next tab	

1 Downloads

```
~/Downloads 📍
├── supergirl.mp4
└── michael.mp4
```

2 Movies

```
~/Movies 📍
└── the_odyssey.mkv
```

`t` `t` a **new tab**, each tab has its own *current directory* 📍

📍 current directory **source** **+ created** ~~**– removed**~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

yazi

🦆 a terminal file manager: the file commands with a few keys

key	action	like
move		
↑ ↓	choose a file	
← / →	parent directory / open	`cd`
`.`	show hidden files	`ls -a`
manage files		
Space	select, for several files	
`a`	create a file, `name/` for a directory	`touch`, `mkdir`
`r`	rename	`mv`
`y` then `p`	copy, then paste	`cp`
`x` then `p`	cut, then paste	`mv`
`d`	move to the trash	
`D`	delete permanently	`rm`
tabs		
`t` `t`	open a new tab	
`1` … `9`	go to a tab	
`[` `]`	previous / next tab	

1 Downloads

```
~/Downloads 📍
├── supergirl.mp4
└── michael.mp4
```

2 Movies

```
~/Movies 📍
└── the_odyssey.mkv
```

in tab `1`: Space select `supergirl.mp4`, `y` copy

📍 current directory source + created ~~– removed~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

yazi

🦆 a terminal file manager: the file commands with a few keys

key	action	like
move		
↑ ↓	choose a file	
← / →	parent directory / open	`cd`
`.`	show hidden files	`ls -a`
manage files		
Space	select, for several files	
`a`	create a file, `name/` for a directory	`touch`, `mkdir`
`r`	rename	`mv`
`y` then `p`	copy, then paste	`cp`
`x` then `p`	cut, then paste	`mv`
`d`	move to the trash	
`D`	delete permanently	`rm`
tabs		
`t` `t`	open a new tab	
`1` ... `9`	go to a tab	
`[` `]`	previous / next tab	

1 Downloads

```
~/Downloads 📍
├── supergirl.mp4
└── michael.mp4
```

2 Movies

```
~/Movies 📍
└── the_odyssey.mkv
```

`2` go to tab `2`

📍 current directory source + created – ~~removed~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

yazi

🦆 a terminal file manager: the file commands with a few keys

key	action	like
move		
↑ ↓	choose a file	
← / →	parent directory / open	`cd`
`.`	show hidden files	`ls -a`
manage files		
Space	select, for several files	
`a`	create a file, `name/` for a directory	`touch`, `mkdir`
`r`	rename	`mv`
`y` then `p`	copy, then paste	`cp`
`x` then `p`	cut, then paste	`mv`
`d`	move to the trash	
`D`	delete permanently	`rm`
tabs		
`t` `t`	open a new tab	
`1` ... `9`	go to a tab	
`[` `]`	previous / next tab	

1 Downloads

```
~/Downloads 📍
├── supergirl.mp4
└── michael.mp4
```

2 Movies

```
~/Movies 📍
├── supergirl.mp4    +
└── the_odyssey.mkv
```

`p` paste: a **copy**, like `cp`

📍 current directory source + created – ~~removed~~

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

yazi

🦆 a terminal file manager: the file commands with a few keys

key	action	like
move		
↑ ↓	choose a file	
← / →	parent directory / open	`cd`
`.`	show hidden files	`ls -a`
manage files		
Space	select, for several files	
`a`	create a file, `name/` for a directory	`touch`, `mkdir`
`r`	rename	`mv`
`y` then `p`	copy, then paste	`cp`
`x` then `p`	cut, then paste	`mv`
`d`	move to the trash	
`D`	delete permanently	`rm`
tabs		
`t` `t`	open a new tab	
`1` ... `9`	go to a tab	
`[` `]`	previous / next tab	

1 Downloads

```
~/Downloads 📍
├── supergirl.mp4
└── michael.mp4      -
```

2 Movies

```
~/Movies 📍
├── michael.mp4      +
├── supergirl.mp4
└── the_odyssey.mkv
```

`1` `x` cut `michael.mp4`, `2` `p` paste: **moved**, like `mv`

📍 current directory source + created - removed

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0

We talked about

- File System Layout
- Paths
- Manual Pages
- Navigation
- File Management
- File Managers

 Copyright (C) 2026 Wyliodrin & Politehnica Bucharest, CC-BY-SA 4.0