# File permissions in Linux

## Project description

I used Linux commands to update, delete, or remove permissions from files and folders. These commands helped manage access rights, organize files, and secure the system by limiting who can view or modify certain items.

## Check file and directory details

To check the details of files and directories, I used the ls -l command, which displays information like permissions

## Describe the permissions string

The permissions string is a 10-character representation that shows file type and access permissions.  
 Example: -rw-r--r--

* The first character indicates the type (- for file, d for directory).
* The next three characters show the owner's permissions (rw- = read and write).
* The following three are for the group (r-- = read-only).
* The last three are for others (r-- = read-only).  
   Each permission group may contain r (read), w (write), and x (execute).

## Change file permissions

I used the chmod command to modify file permissions. For example, chmod g - w updates.txt

## Change file permissions on a hidden file

Hidden files (those starting with a dot, can have their permissions changed the same way. I used chmod 600 .project.txt to give the owner read and write permissions, and no permissions to group or others.

## Change directory permissions

To change directory permissions, I used chmod with execute (x) permission to allow access. For example, chmod 755 myfolder lets the owner read, write, and enter the directory, while group and others can only read and enter it.

## Summary

This project helped me practice using Linux commands to manage file and directory permissions. I learned how to check details, interpret permission strings, and safely change access settings. These skills are important for system administration and securing files in a multi-user environment.