How to Change Permissions

This document is about how to change the permissions on your files using the `chmod` command in the terminal.

First digit	Second digit	Third digit
The owner of the file	Users in the owner's group	Users who are not in the owner's group

To use this table, find where a row and column intersect. For example, to make a file readable and writable, but not executable, find the "Read" column and the "Write" row. The number is 6, so if you typed `chmod 666 file.sh`, nobody could execute `file.sh`, but everybody could read and write. I've taken out the numbers that don't make sense (for example, having write access but not read access)

Full control: 7	Read	Write	Execute
Read	4 (read-only)	6	5
Write	6		
Execute	5		1

Here are some useful bit combinations (assuming you own the file):

Combination	What to use it for	What it does
755	Executable files	I can read, write, and execute, but nobody else can write (this is a read-only script that can be executed).
644	Regular files	Everybody can read, but only I can write.
600	Regular files	I can read and write, but it's off-limits to everyone else.
777	Executable files	Anybody can do anything.
666	Regular files	Anybody can read and write, but nobody can execute.
664	Regular files	I can read and write, users in my group can read and write, but it's read-only for everyone else.

You can use `ls -l file.txt` to view the permissions and other details of `file.txt`. Here's a screenshot of what it looks like (the $ is the prompt):