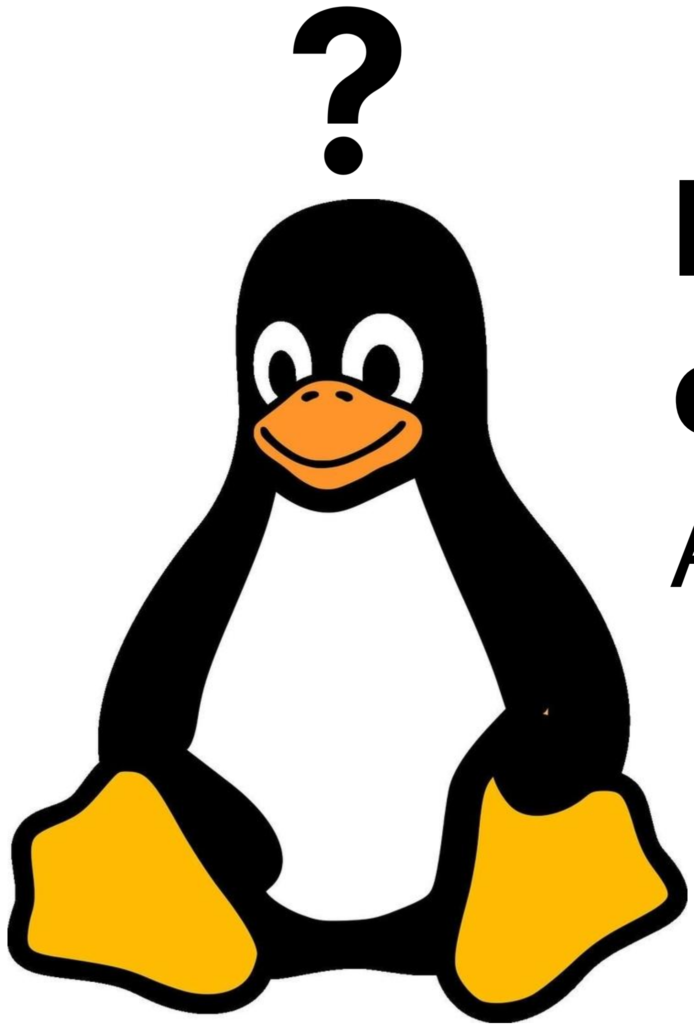


Is unix a penguin, or what?

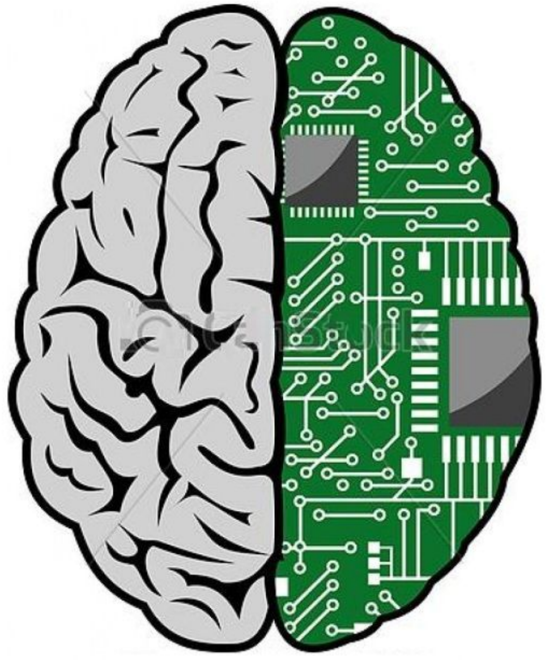
A short presentation

Whats Unix?

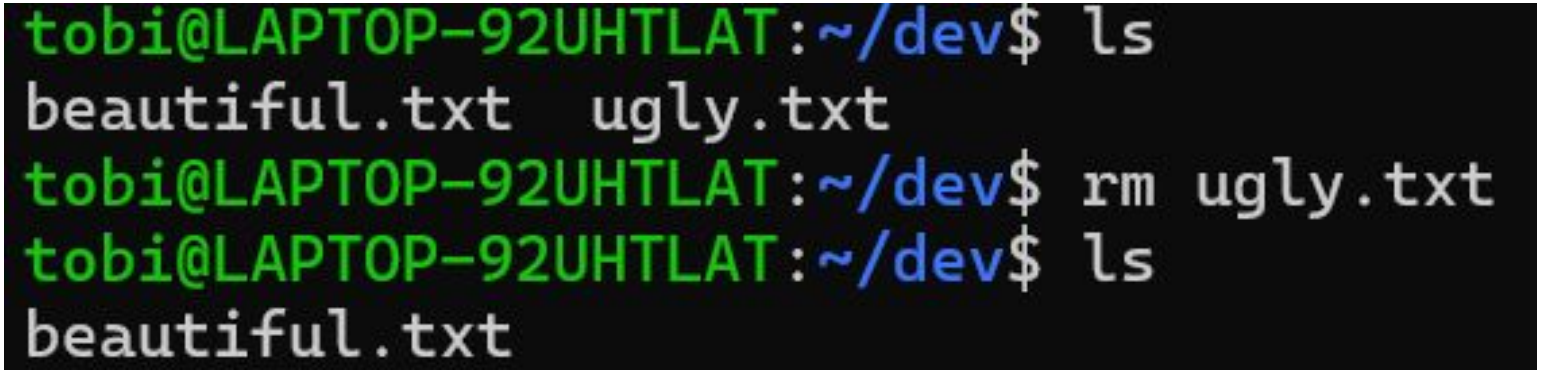
```
tobi@LAPTOP-92UHTLAT:~/dev$ Lets talk to the kernel!
```

Unix is an operating system, with 3 main parts

- The kernel.
  - The center of operating system
  - Allocates time and memory to programs
  - Handles filestore and communication in response to system calls
- The shell
  - Interface between user and kernel
  - Command line interface
- The programs
  - Set of instructions



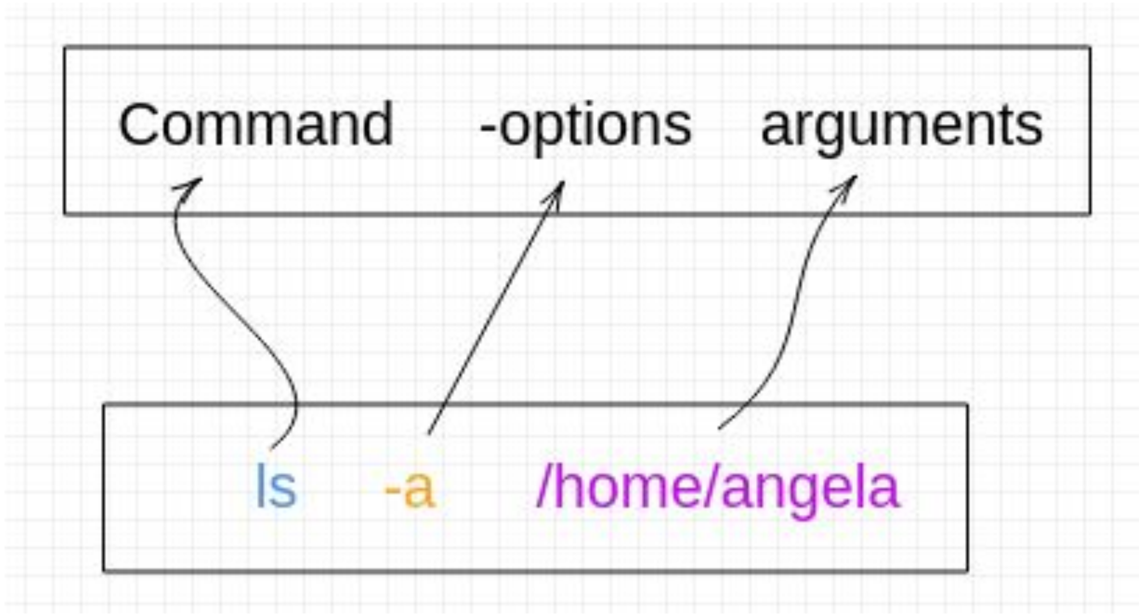
Example

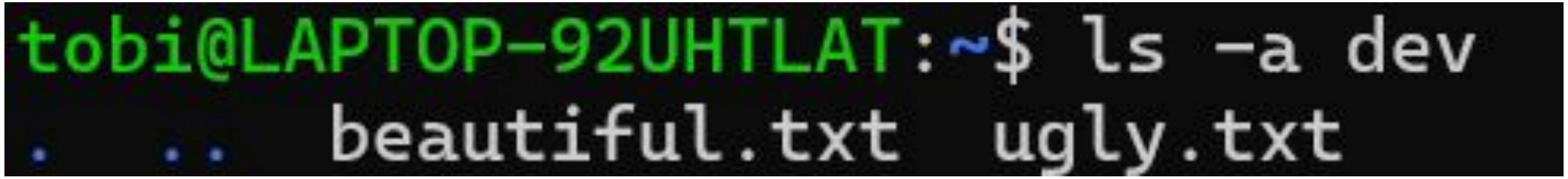


```
tobi@LAPTOP-92UHTLAT:~/dev$ ls
beautiful.txt  ugly.txt
tobi@LAPTOP-92UHTLAT:~/dev$ rm ugly.txt
tobi@LAPTOP-92UHTLAT:~/dev$ ls
beautiful.txt
```

1. User types “rm ugly.txt” in the shell
2. The shell searches for the program “rm”
3. The shell communicates to the kernel with system calls
4. Tells the kernel to perform “rm” program on ugly.txt
5. The kernel allocates resources for “rm” program
6. ugly.txt gets removed
7. And the shell is ready for the next command

Unix commands

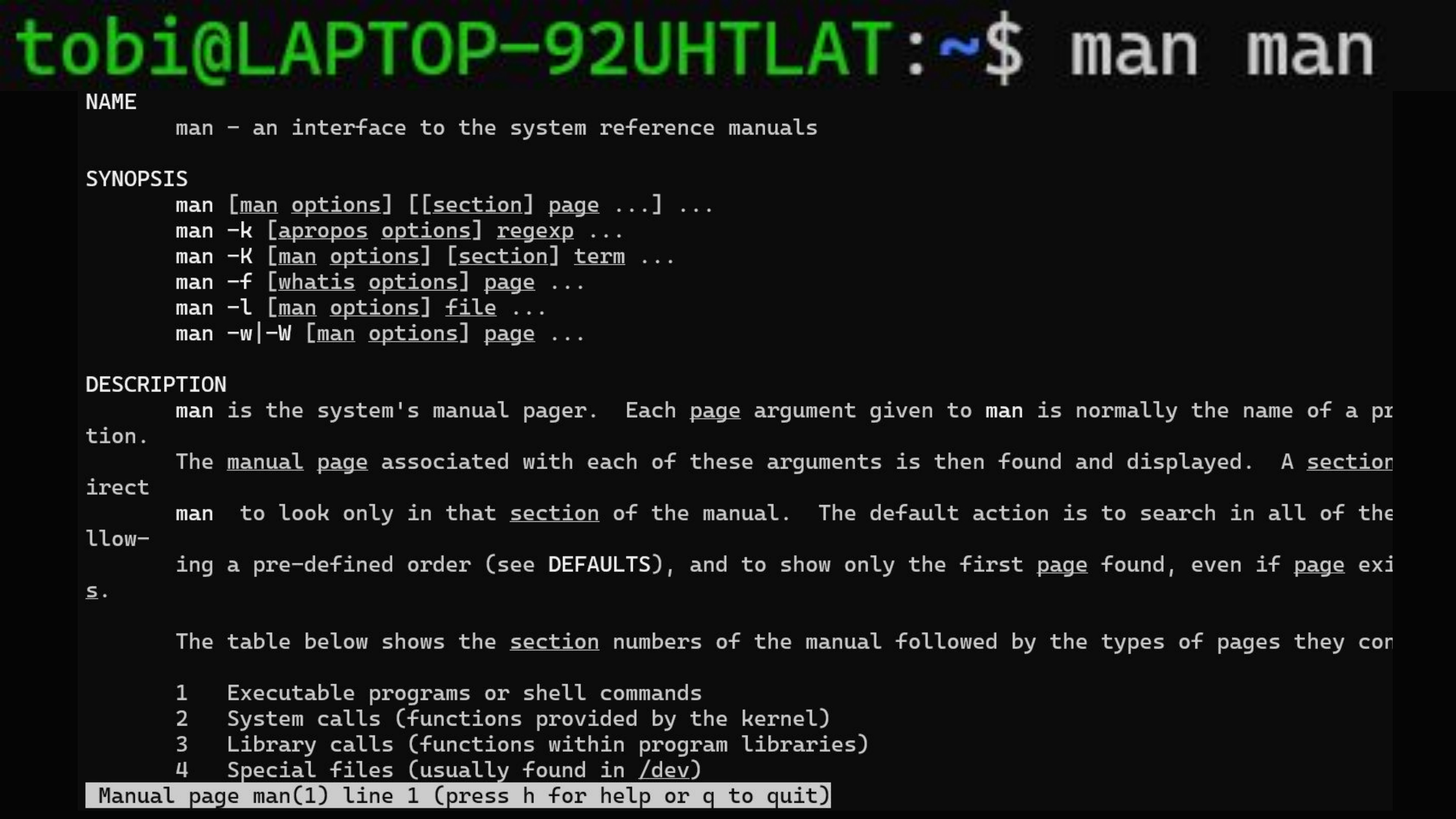




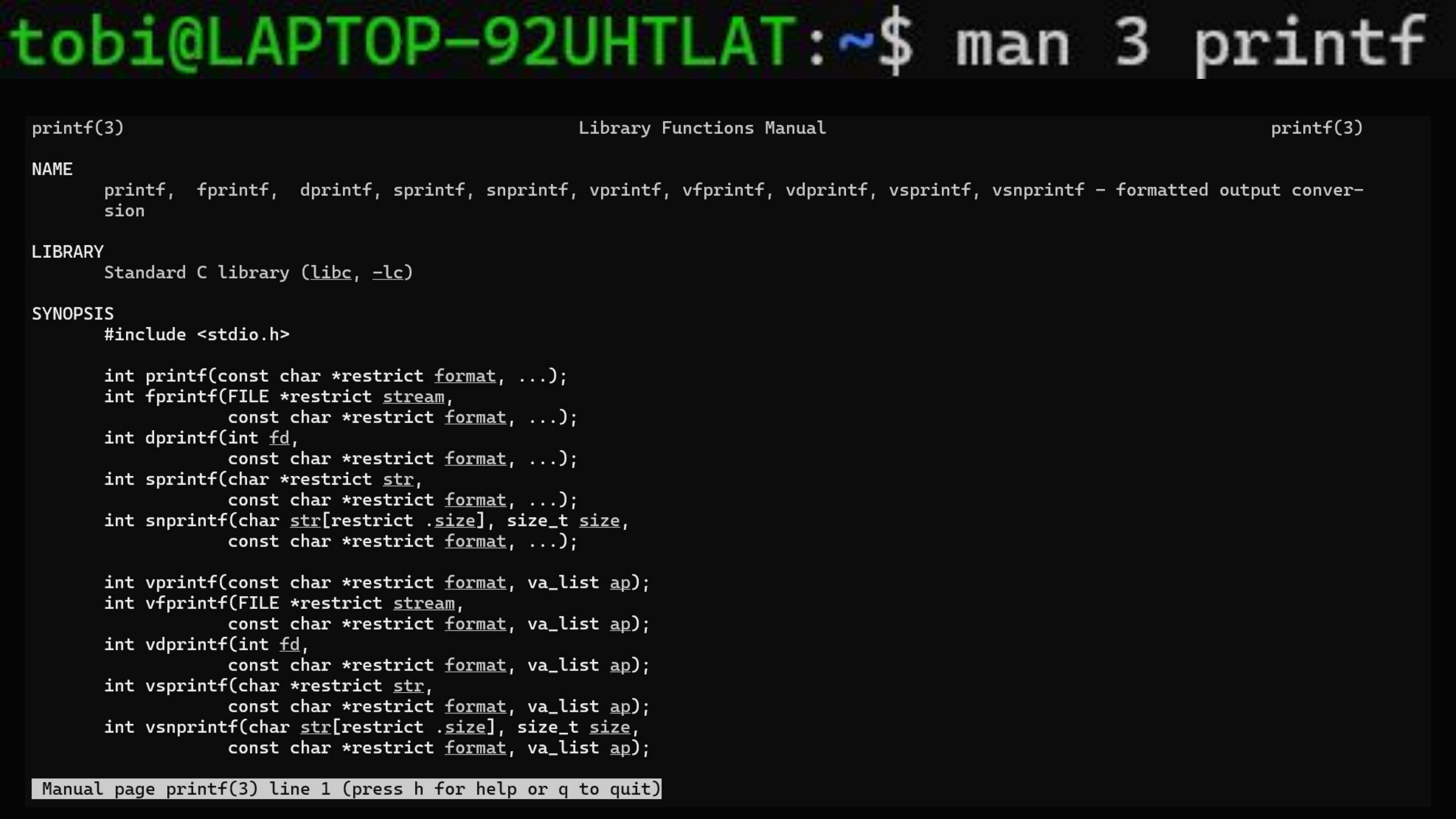
```
tobi@LAPTOP-92UHTLAT:~$ ls -a dev
.  ..  beautiful.txt  ugly.txt
```

The Manual Command:

```
man topic
```



```
tobi@LAPTOP-92UHTLAT:~$ man man
NAME
       man - an interface to the system reference manuals

SYNOPSIS
       man [man options] [[section] page ...] ...
       man -k [apropos options] regexp ...
       man -K [man options] [section] term ...
       man -f [whatis options] page ...
       man -l [man options] file ...
       man -w|-W [man options] page ...

DESCRIPTION
       man is the system's manual pager.  Each page argument given to man is normally the name of a pr
tion.
       The manual page associated with each of these arguments is then found and displayed.  A sectior
irect
       man  to look only in that section of the manual.  The default action is to search in all of the
llow-
       ing a pre-defined order (see DEFAULTS), and to show only the first page found, even if page exi
s.

       The table below shows the section numbers of the manual followed by the types of pages they cor

       1   Executable programs or shell commands
       2   System calls (functions provided by the kernel)
       3   Library calls (functions within program libraries)
       4   Special files (usually found in /dev)
 Manual page man(1) line 1 (press h for help or q to quit)
```

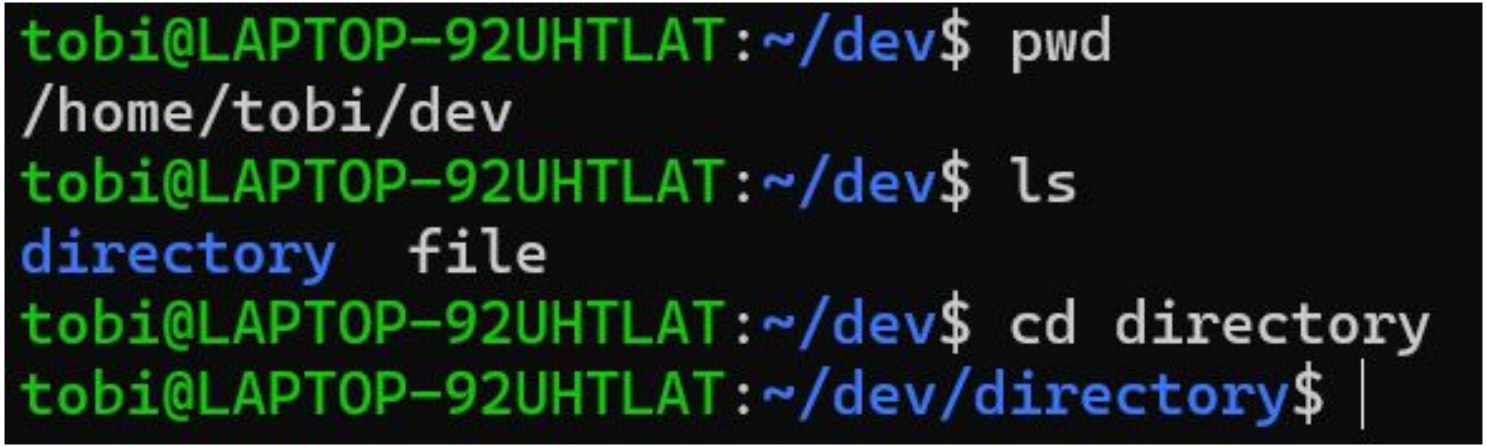


```
tobi@LAPTOP-92UHTLAT:~$ man 3 printf
```

```
printf(3)                        Library Functions Manual                        printf(3)

NAME
       printf,  fprintf,  dprintf, sprintf, snprintf, vprintf, vfprintf, vdprintf, vsprintf, vsnprintf - formatted output conver‐
       sion

LIBRARY
       Standard C library (libc, -lc)

SYNOPSIS
       #include <stdio.h>

       int printf(const char *restrict format, ...);
       int fprintf(FILE *restrict stream,
                   const char *restrict format, ...);
       int dprintf(int fd,
                   const char *restrict format, ...);
       int sprintf(char *restrict str,
                   const char *restrict format, ...);
       int snprintf(char str[restrict .size], size_t size,
                   const char *restrict format, ...);

       int vprintf(const char *restrict format, va_list ap);
       int vfprintf(FILE *restrict stream,
                   const char *restrict format, va_list ap);
       int vdprintf(int fd,
                   const char *restrict format, va_list ap);
       int vsprintf(char *restrict str,
                   const char *restrict format, va_list ap);
       int vsnprintf(char str[restrict .size], size_t size,
                   const char *restrict format, va_list ap);

 Manual page printf(3) line 1 (press h for help or q to quit)
```

Navigating

- **pwd** - writes the path of current directory
- **ls** - Lists content of current directory
- **cd** - Changes directory
    - **cd 'path'** - Go to 'path'
    - **cd ..** - Go to parents directory
    - **cd ~ -** Go to home directory

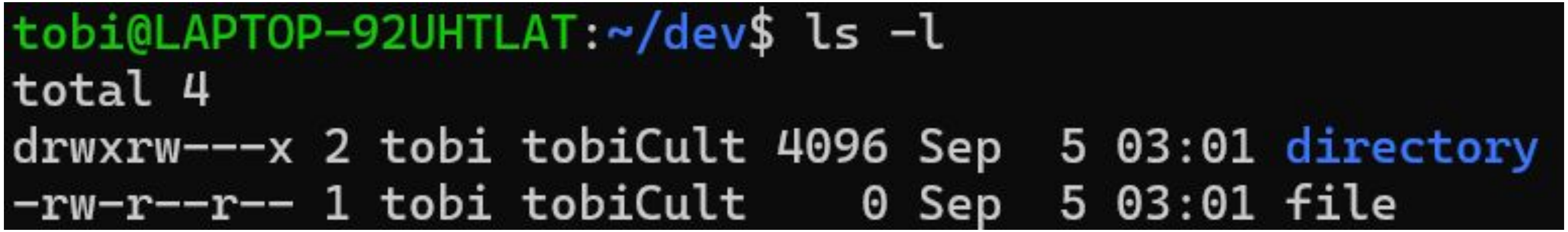


```
tobi@LAPTOP-92UHTLAT:~/dev$ pwd
/home/tobi/dev
tobi@LAPTOP-92UHTLAT:~/dev$ ls
directory  file
tobi@LAPTOP-92UHTLAT:~/dev$ cd directory
tobi@LAPTOP-92UHTLAT:~/dev/directory$
```

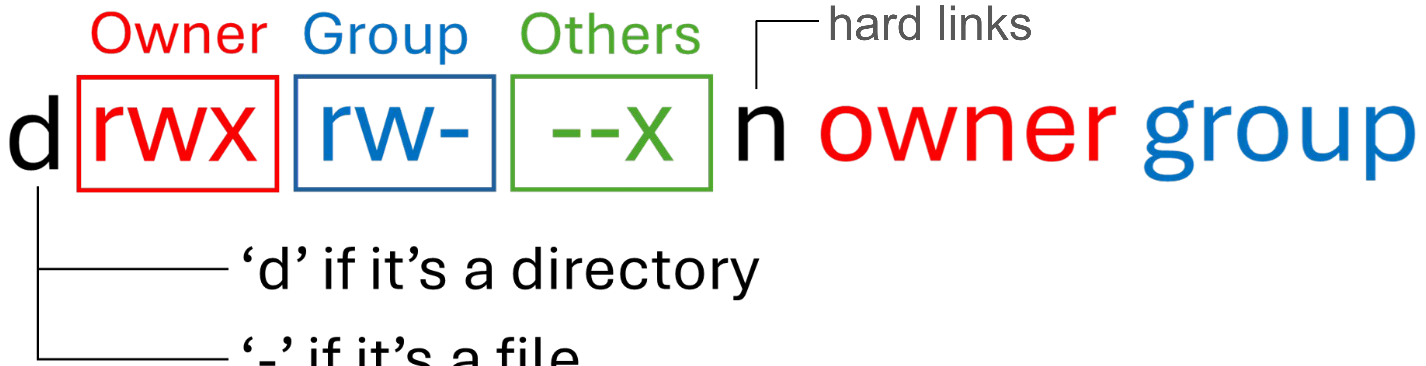
Manipulating files/directories

- **mkdir name** - Creates a directory called “name”
- **touch name** - Creates a file called “name”
- **rmdir name** - Removes an EMPTY directory called “name”
- **rm name** - Removes a file called “name”
- **cp** - Makes copy of file/directory
    - **cp file directory** - Makes a copy of “file” in “directory”
    - **cp -r dir anotherdir** - Recursivly makes a copy of “dir” in “anotherdir”
- **mv** - Moves or renames a file.
    - **mv file path/to/directory** - Moves “file” to “path/to/directory**”**
    - **mv filename anothername** - Renames the file

Permissions



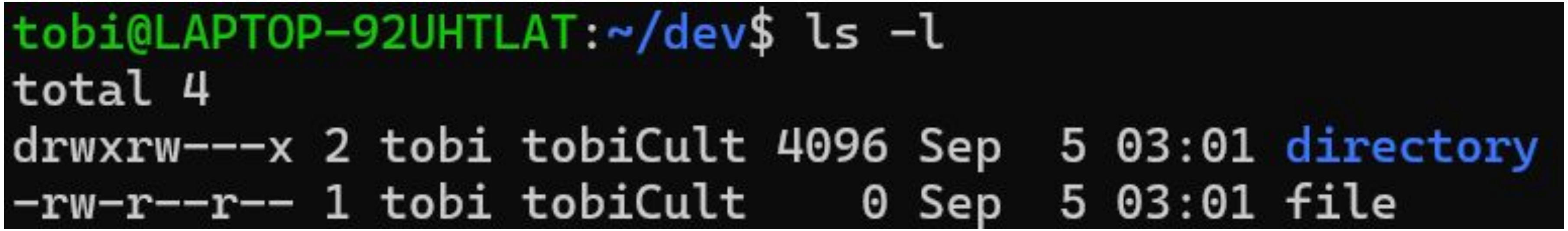
```
tobi@LAPTOP-92UHTLAT:~/dev$ ls -l
total 4
drwxrw---x 2 tobi tobiCult 4096 Sep  5 03:01 directory
-rw-r--r-- 1 tobi tobiCult    0 Sep  5 03:01 file
```



r = read, w = write, x = execute

chown

Changes owner and/or group of file/directory



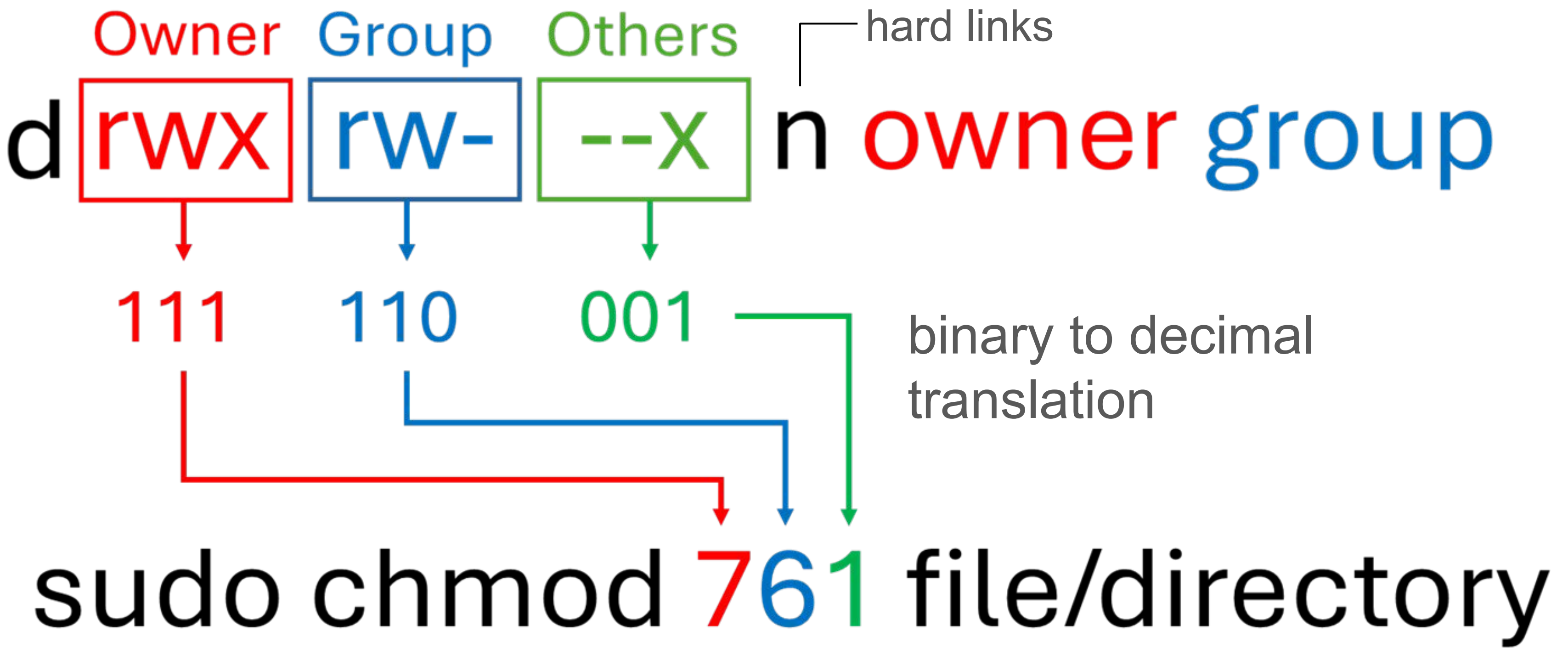
```
tobi@LAPTOP-92UHTLAT:~/dev$ ls -l
total 4
drwxrw---x 2 tobi tobiCult 4096 Sep  5 03:01 directory
-rw-r--r-- 1 tobi tobiCult    0 Sep  5 03:01 file
```

sudo chown owner file/directory

sudo chown owner:group file/directory

chmod

Changes read, write, & execute permission for owner, group, & others



Owner
Group
Others
hard links
d rwx rw- --x n owner group
111
110
001
binary to decimal translation
sudo chmod 761 file/directory

Thank you for listening! :3

Check github for:
Small exercises
A markdown file with commands
This presentation