csce215 — UNIX/Linux Fundamentals
Spring 2022 — Lecture Notes: Basics

This document contains slides from the lecture, formatted to be suitable for printing or individual reading, and with some supplemental explanations added. It is intended as a supplement to, rather than a replacement for, the lectures themselves — you should not expect the notes to be self-contained or complete on their own.

(1.1) The shell

When you give commands in a terminal window, you are interacting with a special program called a **shell**, whose job is to read, interpret, and execute the commands you give, usually by running other programs.

```
$ whoami
jokane
$ uptime -s
2022-03-20 22:03:37
```

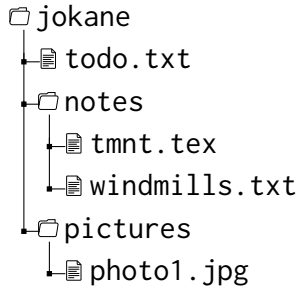
There are several shells that differ in the details of how certain commands work.

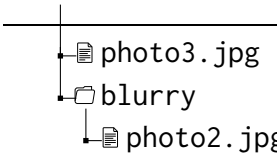
`bash`, `csh`, `fish`, `sh`, `tcsh`, `zsh`, ...

This course focuses on `bash`, which is the most commonly used shell and the default on most systems.

(1.2) Files and directories

Files are organized into a **hierarchy** of directories (a.k.a. folders).



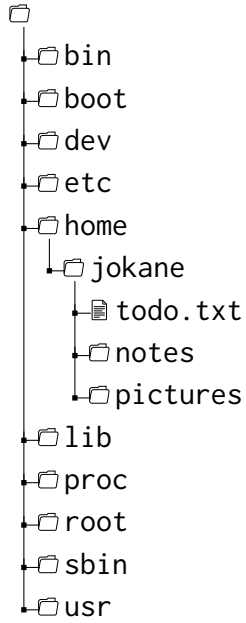


Every user has a **home directory** to store documents, code, etc.

Each file and each directory has a **name**, with is case sensitive, i.e. `Hello` and `hello` are different names.

(1.3) *The root directory*

Every file on a system lives somewhere in one big hierarchy that starts from a **root directory**.



(1.4) *The current directory*

Each program, including the shell, always has a **current directory** (or **working directory**).

Most commands operates on things in the current directory.

pwd 
Show the current directory.

```
$ pwd
/home/jokane
```

You'll want to develop a habit of knowing what the current directory is. Keeping track is **extremely important** for using the shell correctly.

(1.5) *Changing directories:* `cd`

We can change the current directory to move around.

`cd` 
Change the current directory.

```
$ pwd
/home/jokane
$ cd pictures
$ pwd
/home/jokane/pictures
```

Notice the / character for separating directory names.

(1.6) *Listing files:* `ls`

We'll often want to see what files and subdirectories are in the current directory.

`ls` 
Show a list of files and subdirectories. ('ell-ess', not 'one-ess')

```
$ pwd
/home/jokane/pictures
$ ls
blurry
photo1.jpg
photo3.jpg
```

You can also see the files in a specific directory:

```
$ ls blurry
photo2.jpg
```

Or the files that match a certain pattern:

```
$ ls photo*
photo1.jpg
photo3.jpg
```

(1.7) *Getting more detail:* `ls -l`

Often we want to know more than just a list of files.

`ls`

Show a list of files and subdirectories.

`-l` Show details about each file.

('dash ell', not 'minus one')

```
$ ls -l
total 128
drwxrwxr-x 2 jokane jokane  4096 Dec  7  1941 blurry
-rw-rw-r-- 1 jokane jokane 77931 Dec  7  1941 photo1.jpg
-rw-rw-r-- 1 jokane jokane 41772 Dec  7  1941 photo3.jpg
```

The output includes **type**, **permissions**, **owner**, **group**, **size** in bytes, and **modification date**.

(1.8) *Other options for* `ls`

`ls`

Show a list of files and subdirectories.

`-a` Show all files, even if their names start with a period.

`-h` Show sizes in a human-friendly format.
Use with `-l`.

```
$ ls -l -a -h
total 136K
drwxrwxr-x 3 jokane jokane 4.0K Dec  7  1941 .
drwxrwxr-x 4 jokane jokane 4.0K Dec  7  1941 ..
drwxrwxr-x 2 jokane jokane 4.0K Dec  7  1941 blurry
-rw-rw-r-- 1 jokane jokane  77K Dec  7  1941 photo1.jpg
-rw-rw-r-- 1 jokane jokane  41K Dec  7  1941 photo3.jpg
```

(1.9) *Special directory names*

There are several important shortcuts for referring to certain directories.

`.`	current directory
`..`	parent directory
`~`	home directory
`/`	root directory

(1.10) *Absolute and relative paths*

When you mention a file, it's usually interpreted as a **relative path**, i.e. starting from the current directory.

Example relative paths (with `pictures` as the current directory):

```
$ ls blurry
photo2.jpg
```

```
$ ls ..
notes
pictures
todo.txt
```

```
$ ls .
blurry
photo1.jpg
photo3.jpg
```

Start with a / to get an **absolute path**, i.e. starting from the **root directory**.

```
$ ls /
bin
boot
dev
etc
home
lib
proc
root
sbin
usr
```

```
$ ls /home
jokane
```

Or use ~ to get the home directory:

```
$ ls ~
notes
pictures
todo.txt
```

```
$ ls ~/notes
tmnt.tex
windmills.txt
```

(1.11) *Examples with* cd

```
$ pwd
/home/jokane/pictures/blurry
$ cd ..
$ pwd
/home/jokane/pictures
$ cd .
$ pwd
/home/jokane/pictures
$ cd ~
$ pwd
/home/jokane
```

(1.12) *Remember*

Use `ls` to see what's in a directory.

Use `cd` to 'go to' a different directory.

(1.13) *Displaying the contents of a file*

Sometimes we want to see what's in a file.

`cat` 
Show the contents of a file.

```
$ cat ~/notes/windmills.txt
In a village of La Mancha, the name of
which I have no desire to call to mind,
there lived not long since one of those
gentlemen that keep a lance in the
lance-rack, an old buckler, a lean hack,
and a greyhound for coursing. An olla of
rather more beef than mutton, a salad on
most nights, scraps on Saturdays,
lentils on Fridays, and a pigeon or so
extra on Sundays, made away with
     . . .
     . . .
```

(1.14) *Displaying a long file*

Sometimes a file is too long for `cat` to be helpful.

`less`

Show the contents of a file interactively, allowing scrolling and searching.

- `arrow keys` scroll
- `/` search forward from here
- `q` quit
- `n` search again
- `g` go back to the start
- `?` search backward from here

```
$ less ~/notes/windmills.txt
```

(1.15) *Getting more information*

Linux systems have a built-in manual.

`man`

Show the documentation for a command.

```
$ man ls
$ man pwd
$ man man
```

This secretly uses `less`, so any keys that work in `less` also work in `man`.

(1.16) *Sample final exam questions*

1. Which option can be used with the `ls` command to show details including permissions, owner, group, size in bytes, and modification date about each file?

   A. `-h`
   B. `-a`
   C. `-l`
   D. `-v`

2. The purpose of the `less` command is to _____.

   A. reduce the number of devices connected to the computer
   B. show a list of files in a directory, including hidden and deleted files
   C. show a list of running processes in the system, including processes owned by other users
   D. show the contents of a file interactively, allowing scrolling and searching

3. Which of these refers to the home directory?

   A. .
   B. /
   C. . .
   D. ~

4. The purpose of the `man` command is to _____.

   A. show the path to the executable of a command
   B. show the history for a command
   C. show the documentation for a command
   D. show the version for a command

5. The command to show a list of files and subdirectories is _____.

   A. `show-files`
   B. `1s`, as in 'one-ess'
   C. `ls`, as in 'ell-ess'
   D. `file-show`

6. The purpose of the `pwd` command is to _____.

   A. Show the contents the root directory.
   B. Show the name of the home directory.
   C. Show the contents of the current directory.
   D. Show the name of the current directory.

7. Which command will change the current directory to the user's home directory? (That is, which command will 'go to' the home directory?)

A. `pwd ~`
B. `cd ~`
C. `pwd ..`
D. `cd ..`

8. The command `ls ~/..` would show a list of all _____.

A. home directories on the system
B. files in the current directory
C. files in all directories on the system
D. files in user's home directory

9. The purpose of the `cd` command is to _____.

A. Change the current directory.
B. Print the name of the current directory.
C. Change the home directory.
D. Clear the disk.

10. The purpose of the `cat` command is to _____.

A. display the contents of a file
B. edit a file
C. categorize a file
D. move a file

11. Within `less`, which command will search forward through the file for the word 'public'?

A. `:public`
B. `\public`
C. `/public`
D. `public`