ULI101: INTRODUCTION TO UNIX / LINUX AND THE INTERNET

WEEK 2: LESSON 1

UNIX & LINUX FILE MANAGEMENT CONCEPTS
MANAGING DIRECTORIES

LESSON 1 TOPICS

Unix / Linux File Management Concepts

- Purpose of Directories
- Directory Pathnames / Tree Diagrams
- Filename Rules

Managing Directories

- Creating / Viewing Contents of / Manipulating / Removing Directories:
  `mkdir -p`, `rmdir`, `rm -r -i`, `ls -l -d -R` , `tree`, `cp -R`, `mv`
- Demonstration

Homework

- Perform **Tutorial 2: Unix / Linux File Management (Investigation 1)**
  Perform LINUX PRACTICE QUESTIONS (1 – 8)

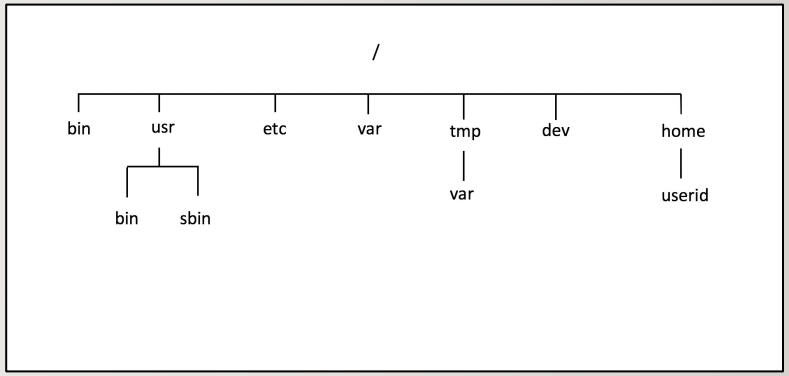
LINUX FILE MANAGEMENT CONCEPTS

Purpose of Unix / Linux Directories

To better **organize** files (eg. text, images, documents, spreadsheets, programs) within your Matrix account, they should be stored in **directories**.

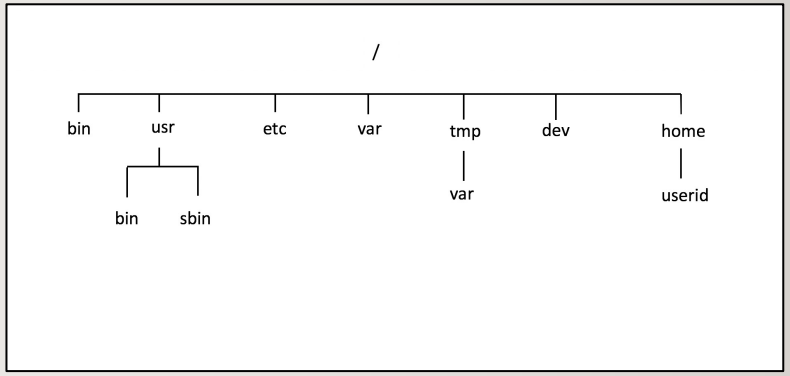
To further organize <u>many</u> files, directories may contain **sub-directories**.

Learning how to issue Linux commands for **navigating** and **manipulating** directory and files within the the Linux filesystem are **essential skills** for Linux users and Linux system administrators (i.e. *sysadmins*).



LINUX FILE MANAGEMENT CONCEPTS

Purpose of Unix / Linux Directories



The Unix/Linux file system is **hierarchical**, like other operating systems such as **Windows**, **macOS**, etc. In Unix / Linux (as opposed to Windows), there are no drive letters such as **C:**, or **D:**

All files and directories appear under a single ancestor directory called the "**root directory**".

In the Linux (Unix) OS, the "**root directory**" / is the starting directory, and other "*child directories*", "**grandchild directories**", etc. can be created as required. The hierarchical structure resembles an "*upside-down tree*". There is actually a command called **tree** that displays a "**directory tree diagram**"!

```
tree linux
linux
└── uli101
    ├── notes
    ├── practice
    └── samples
```

LINUX FILE MANAGEMENT CONCEPTS

Directory Pathnames

*A **pathname** is used to specify the **location** of a file within the file system.*

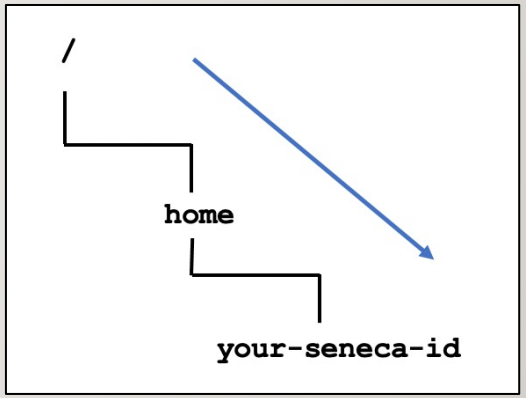
*A pathname **points** to a file system location by **following the directory tree hierarchy** expressed in a string of characters in which path components, separated by a delimiting character, represent each directory.*

*The **delimiting character** is most commonly the slash character ("/").*

Example:

`/home/your-seneca-id`

`without / : relative path`



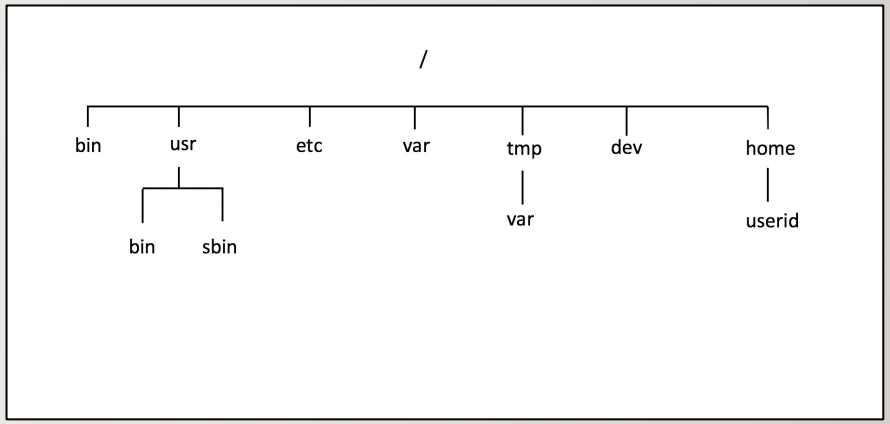
Reference: https://en.wikipedia.org/wiki/Path_(computing)

LINUX FILE MANAGEMENT CONCEPTS

Common Unix / Linux Directories

Below are several common Unix / Linux Directories and their purpose:

Directory Pathname	Purpose
/	Root directory (ancestor to all directories)
/home	Used to store users' home directories
/home/username	A **specific** User's Home Directory
/bin , /usr/bin	Common system binaries (commands)
/usr/sbin	Common utilities for system administration
/etc	System administration files (eg. passwd)
/var variable log	Dynamic files (log and mail files)
/tmp , /var/tmp	Temporary files for programs
/dev	Device driver files (terminals, printers, etc.)



Everything is file

LINUX FILE MANAGEMENT CONCEPTS

Directory File Naming Rules

Before learning to **create** directories, it is important to understand what represents an appropriate directory filename. Here are some **rules**:

Unix / Linux File Naming Rules
✓ Unix/Linux characters are **case sensitive** (e.g. always use lowercase letters) ASCII: Capital letter come before lowercase letter
✓ Adopt a **consistent directory naming scheme** (this will help you to better navigate within your directory structure)
✓ Make your directory names **meaningful** (short but descriptive)
✓ **Avoid using spaces** for directory names (consider **periods**, **hyphens**, and **underscores** instead)
✓ **Avoid non-alphanumeric characters**, as they may have a special meaning to the system that will make your work more difficult when changing to directories, etc.

MANAGING DIRECTORIES

Managing Directories

Below are some common Unix / Linux commands to manage Directories:

Directory Pathname	Purpose
`mkdir -p`	Creates a directory. The **-p** option creates parent directories then directory pathnames specified.
`rmdir`	Removes empty directories.
`rm -r -i`	Removes files, but when used with **-r** option, will remove non-empty directories and their contents. The **-i** option is used to prompt user to confirm deletion of directory contents
`ls -l -d -R , tree`	List directory contents. Useful to verify if directory was created. The **-d** option lists the directory itself (not contents) The **-R** option displays directories and subdirectory contents. The tree command displays diagram of directory structure.
`cp -R`	Copies directory and its contents (recursive) to a different directory
`mv`	Moves directory and its contents to a different directory

-p: if the directory doesn't exist, create for me

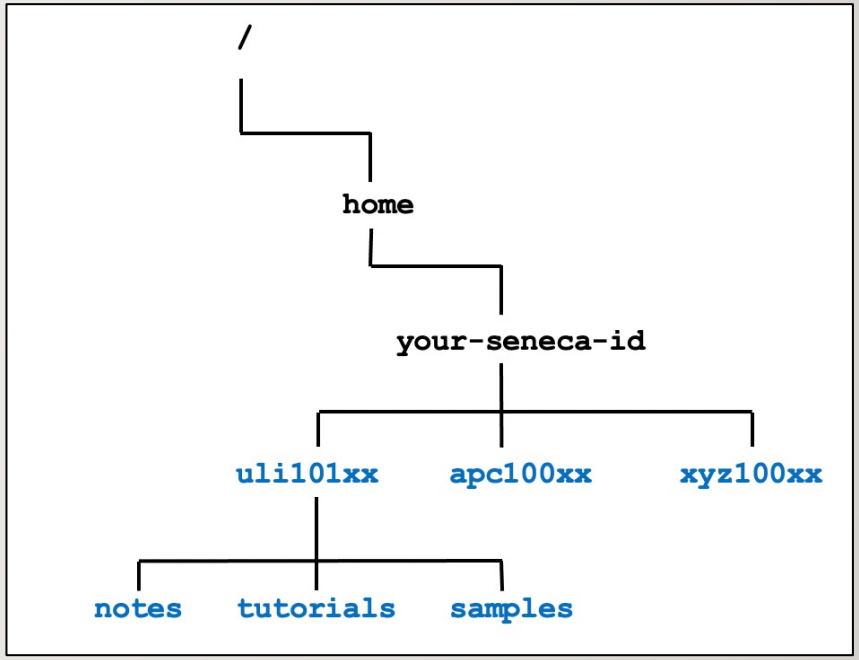
directory doesn't exist, rename
if directory exist, move the directory

MANAGING DIRECTORIES

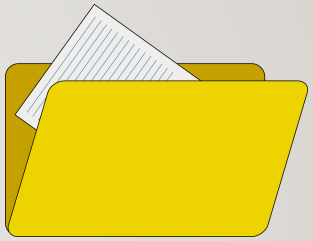
Managing Directories Demonstration

Your instructor will demonstrate how to manage directories by issuing Unix / Linux commands:

- Create directory structure as shown in diagram to the right
- View / Verify created directories
- Copy directories
- Move directories
- Remove empty directories
- Remove non-empty directories



MANAGING DIRECTORIES



Determine Type of File

When issuing the **`ls`** command to view the contents of a directory, the **`-l`** option can be used to help determine file type.

```
ls -l
```

```
drwxr-xr-x 2 murray.saul users 6 Jan 11 09:42 documents
-rw-r--r-- 1 murray.saul users 0 Jan 11 09:42 file.txt
crw-rw-rw- 1 root root 1, 3 Dec  2 07:25 /dev/null
```

The first character on the **left** of the output indicates the type of file:

`d:` directory file

`-:` regular file

`b` or `c:` device file

MANAGING DIRECTORIES

Hidden Files (Don't show by default)

A file is hidden if its name starts with a period "." This can hide both regular files and directory files.

Why make files hidden?

- To clean up directories
- To hide backups
- To protect important files from accidental deletion

If you issued the `ls` command without arguments, hidden files do NOT appear.

The `ls` command with the `-a` option will show all files including hidden and non-hidden. Current and Parent directories (. and ..) are displayed.

. refer to current directory .. refer to parent directory

The `ls` command with the `-A` option will show all files including hidden and non-hidden. Current and Parent directories (. and ..) are NOT displayed.

rm ./f : remove -f in the current directory

HOMEWORK

Getting Practice

Perform the online tutorial **Tutorial 2: Unix / Linux File Management (Due: Friday Week 3 @ midnight for a 2% grade)**:

- INVESTIGATION 1: MANAGING DIRECTORIES
- LINUX PRACTICE QUESTIONS (Questions 1 – 8)