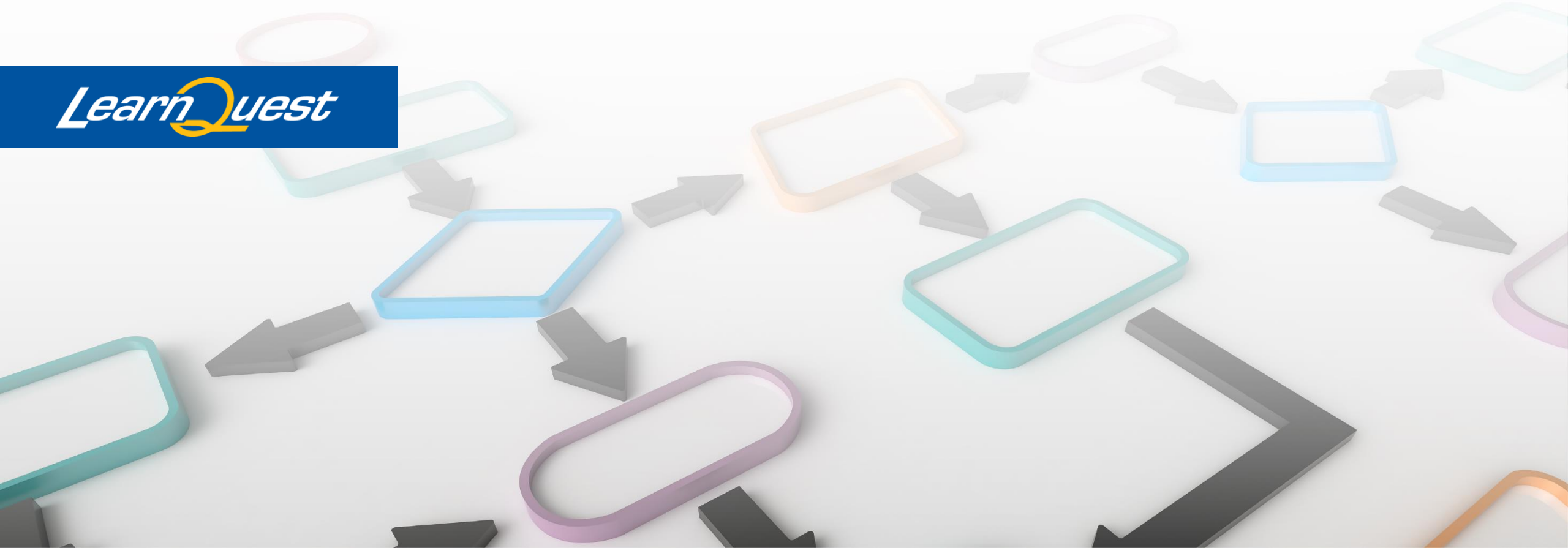
Linux Fundamentals

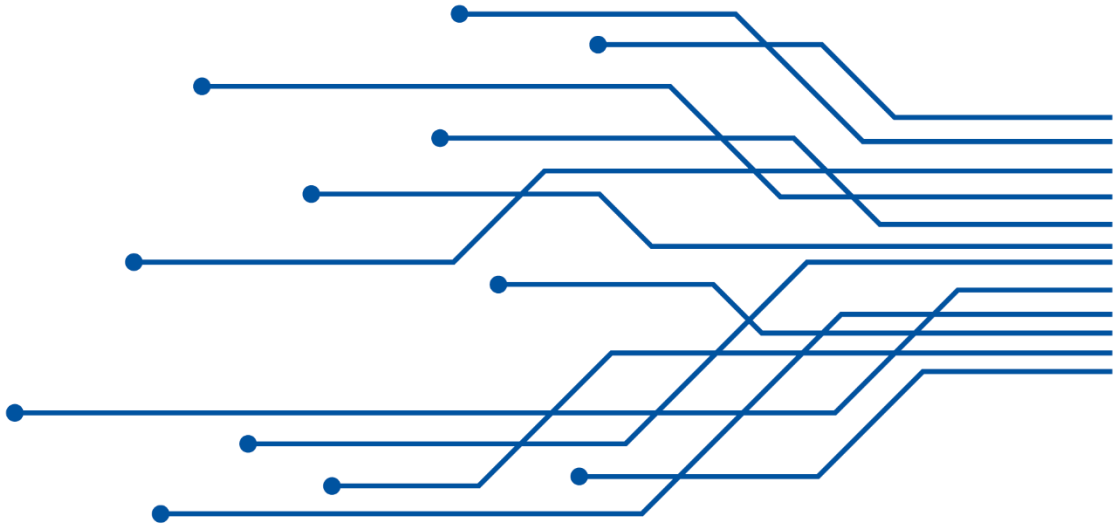
- 1st Course in Linux Foundations Specialization



View, Create, Copy, Move and Remove Files

In the third module of this course, we will learn how to manage files and directories in the Linux operating system.

3



Learning Objectives

View, Create, Copy, Move and Remove Files

Upon completion of this module, learners will be able to:

- View and Create Files
- Copy and Move Files
- Remove Files
- Link Files and Directories
- Read File Portions
- Read Complete Files
- Find File Differences

Lesson 1

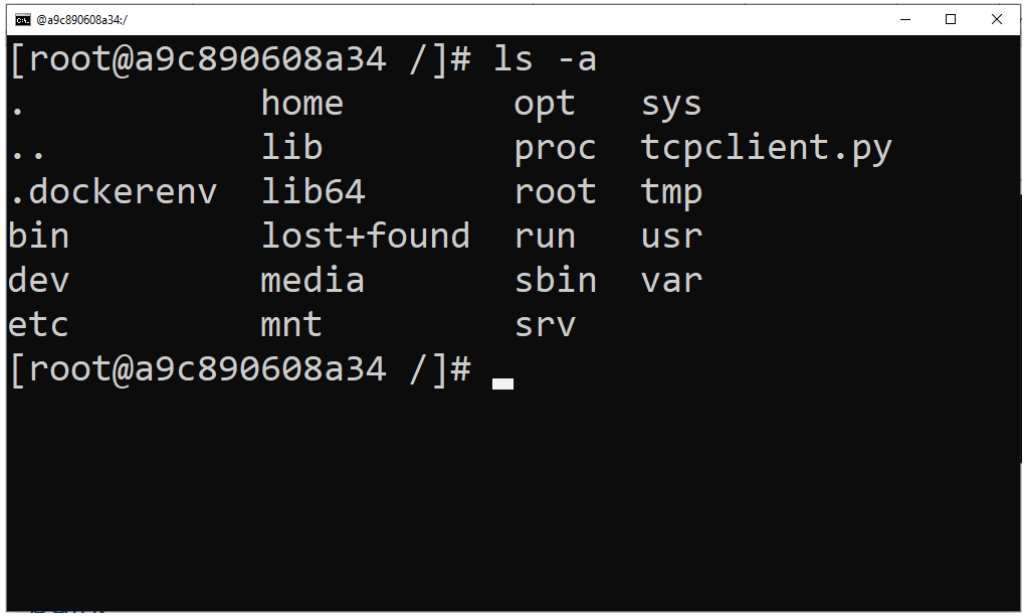
Handle Files and Directories

In this lesson, we look at how to View, Create, Copy, Move and Remove Files

ls Command Deep Dive (part I)

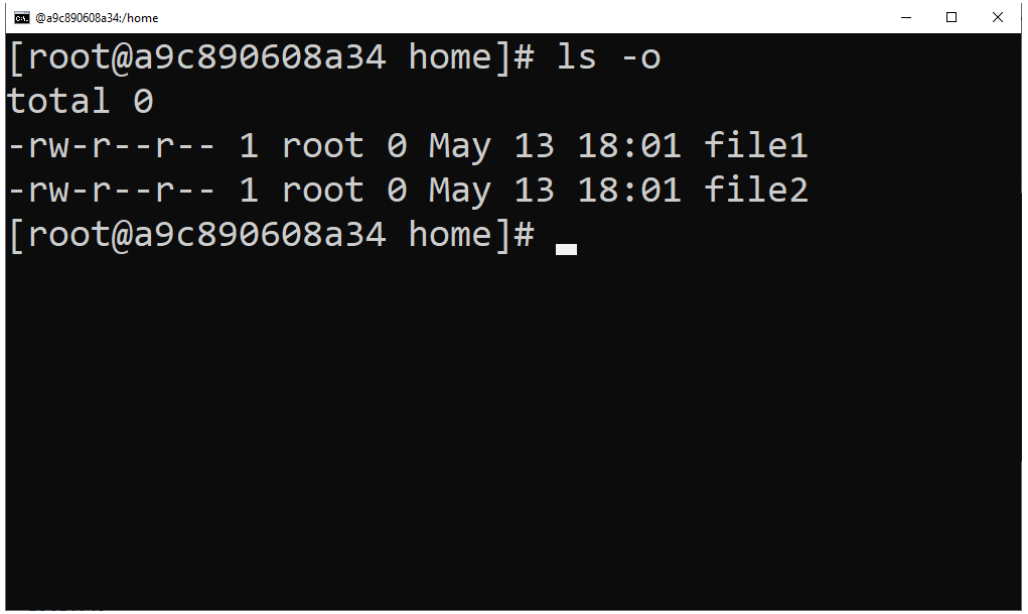
ls [options] [paths]

- -a: list all files including hidden files. These are files that start with ".".
- -A: list all files including hidden files except for "." and ".." – these refer to the entries for the current directory, and for the parent directory.
- -R: list all files recursively, descending down the directory tree from the given path.



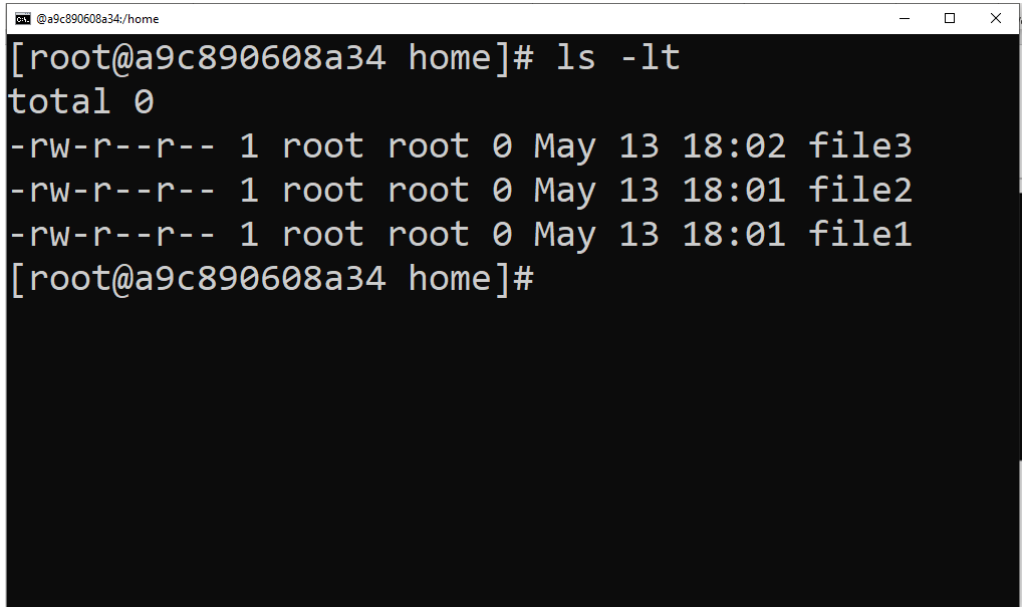
ls Command Deep Dive (part II)

- -l: list the files in long format i.e. with an index number, owner name, group name, size, and permissions.
- –o: list the files in long format but without the group name.
- -g: list the files in long format but without the owner name.
- -i: list the files along with their index number.
- -s: list the files along with their size.



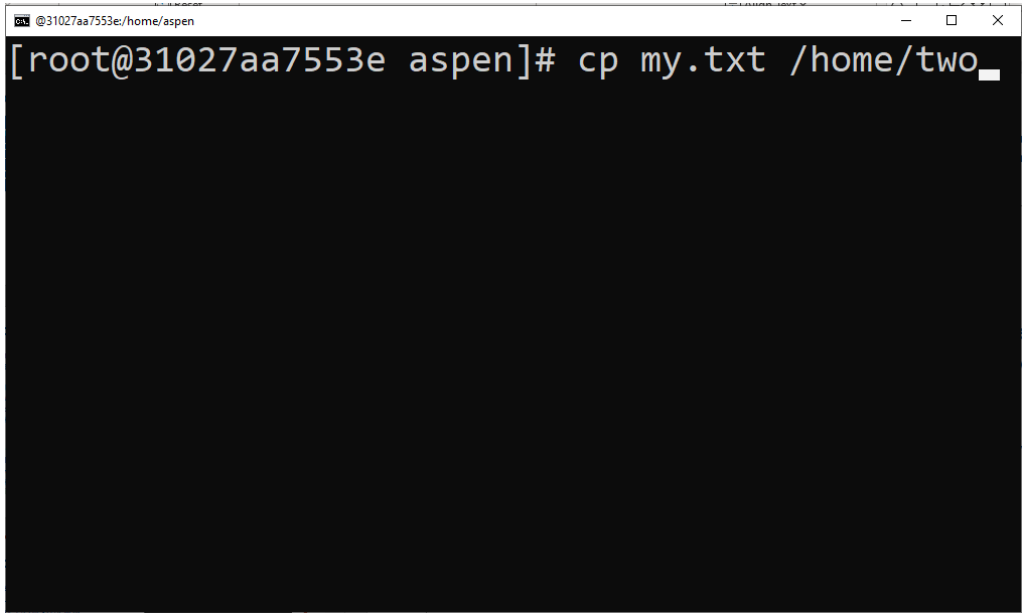
ls Command Deep Dive (part III)

- -t: sort the list by time of modification, with the newest at the top.
- -S: sort the list by size, with the largest at the top.
- -r: reverse the sorting order.



File Commands Review

- cat – list the contents of a file on the standard output
- cp – copy files or directories
- mv – move or rename files or directories
- mkdir – create a new directory in the current directory
- rm – remove file and directories
- Touch – update time and date of file



Lesson 1 Review

Touch command can create new files

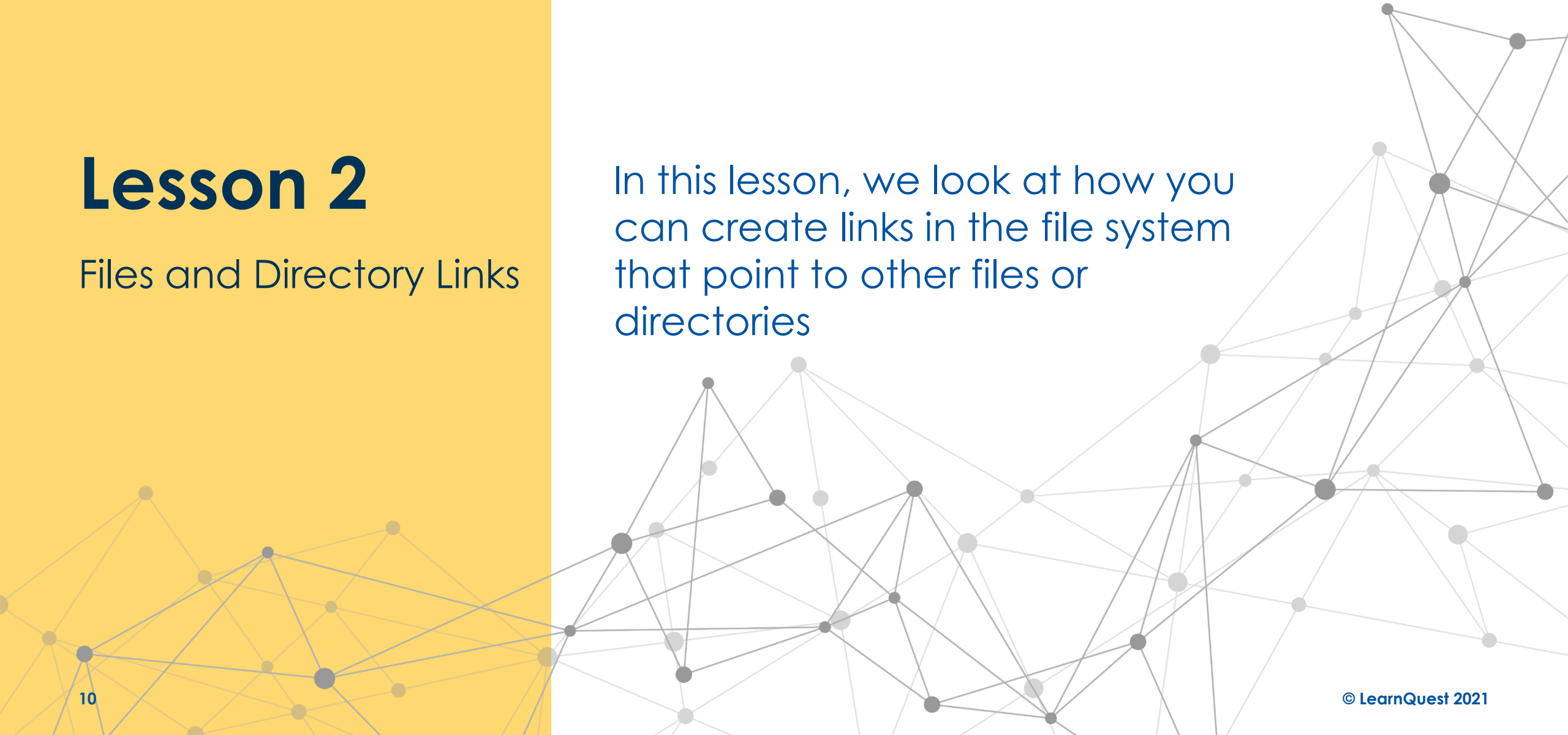
The ls command can show hidden files

Many commands work with both directories and files

Lesson 2

Files and Directory Links

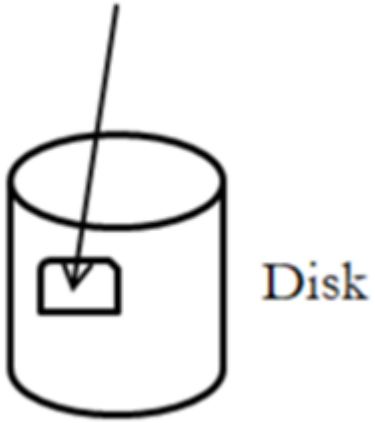
In this lesson, we look at how you can create links in the file system that point to other files or directories

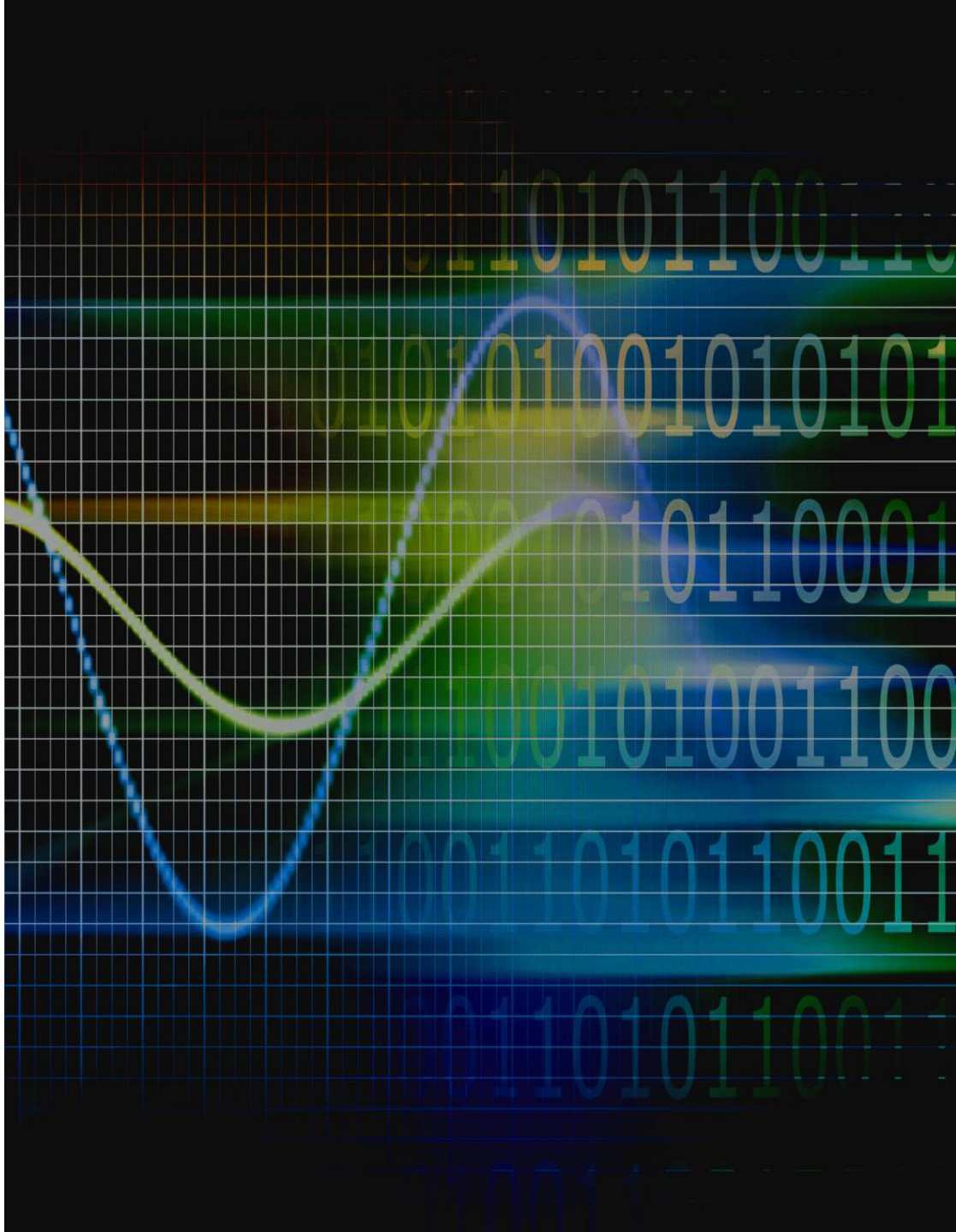


Hard Link

A file (or directory) with one index (inode) number and at least two different file names.

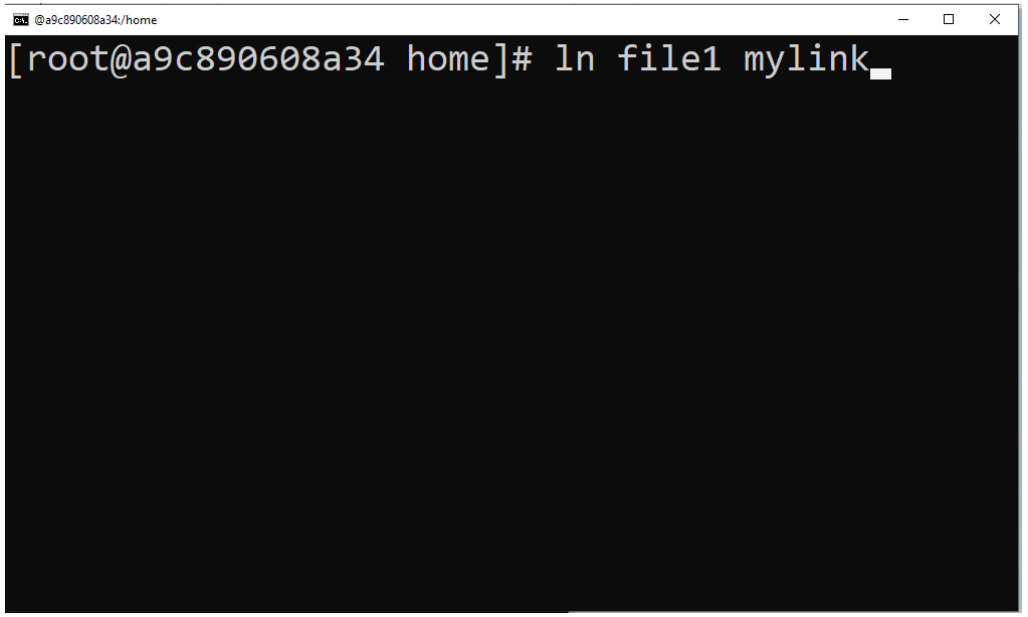
Filename1 -> inode #[] <- Filename2





Creating Hard Link

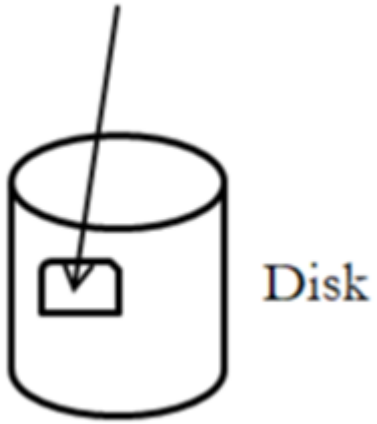
- Original file must exist prior to issuing the command
- Linked file is created when command is issued
- ln [originalfilename] [linkname]

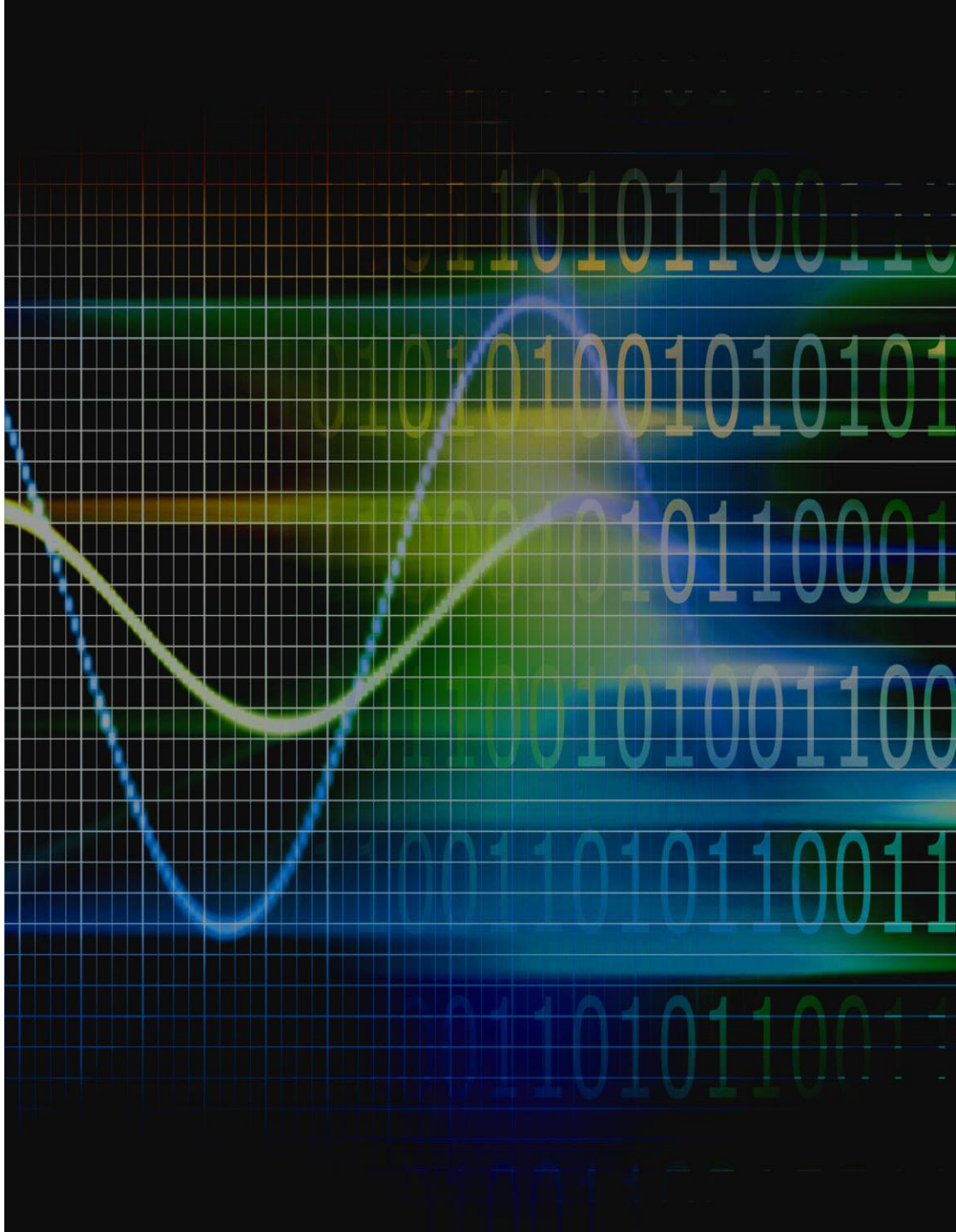


Soft Link

A file with different index (inode) numbers. The soft link file points to other file.

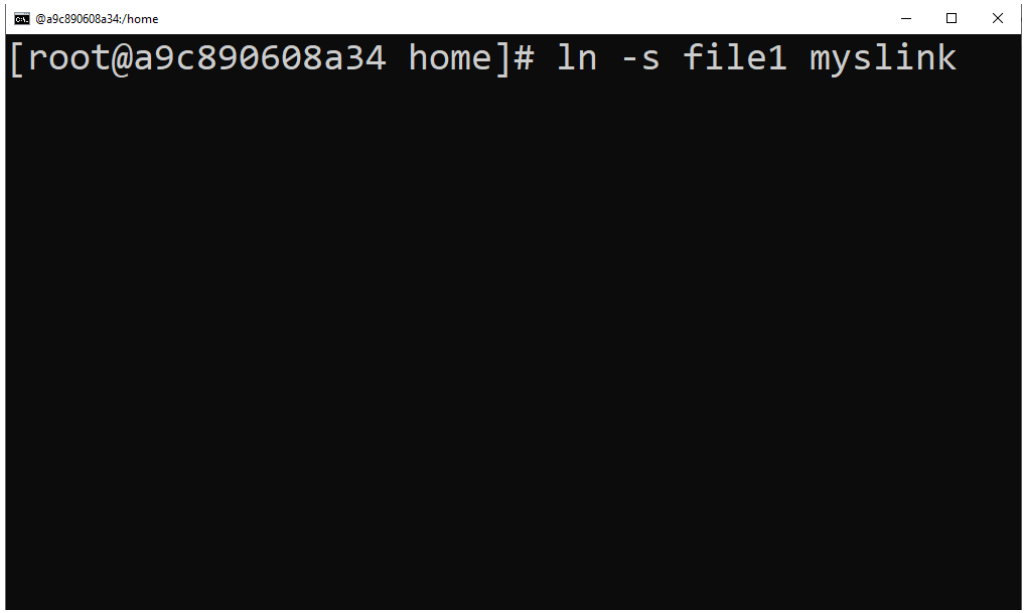
Filename1 inode #[] -> Filename2 inode #[]





Creating Soft Link

- Original file must exist prior to issuing the command
- Linked file is created when command is issued
- ln –s [originalfilename] [linkname]



Lesson 2 Review

If you delete the original file with a hardlink, the link still works

If you delete the original file with a softlink the link will be broken

You can link both files and folders

Lesson 3

Reading Files

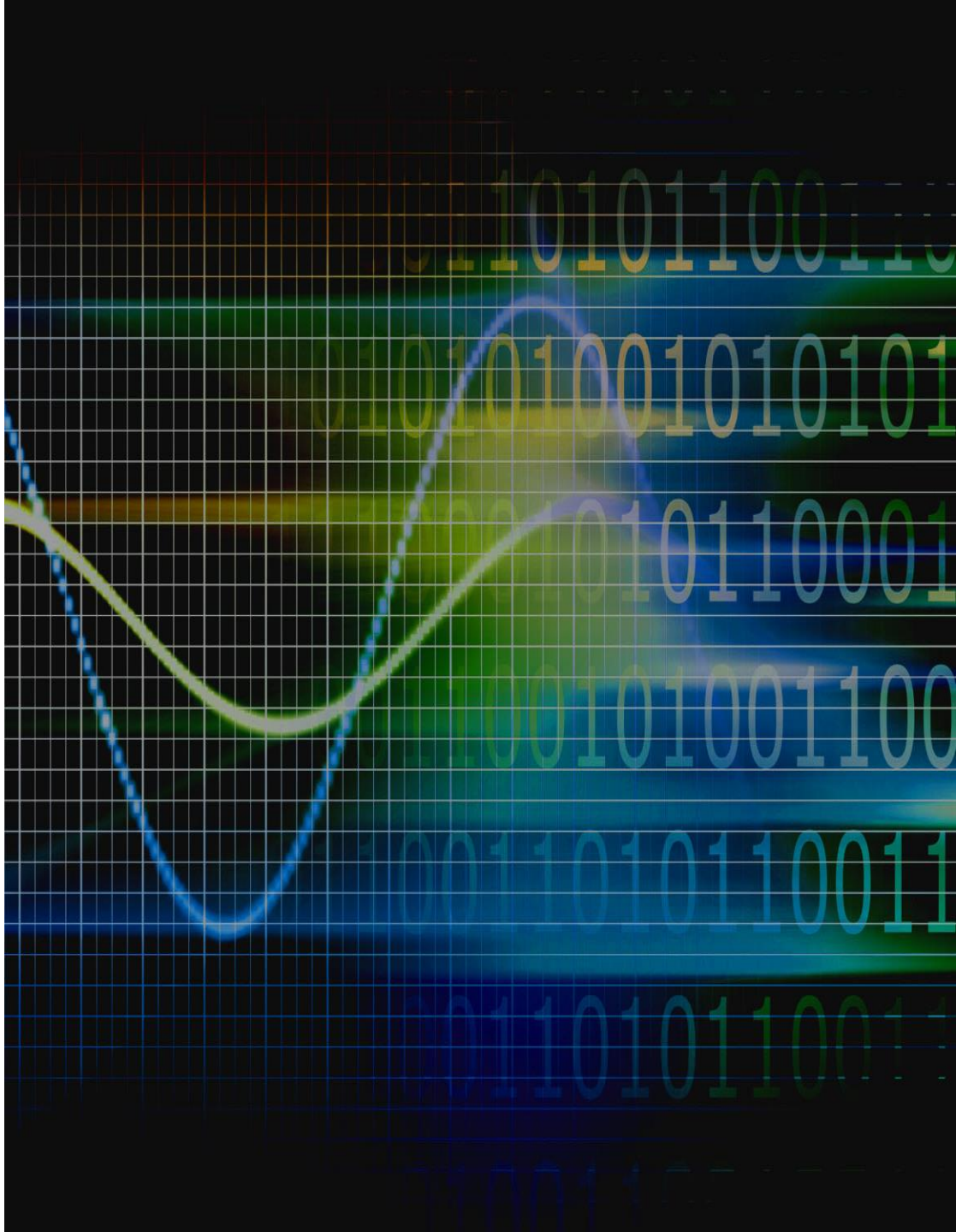
In this lesson, we look at how to read complete and parts of files from the Linux command line.

Linux Text Files

Linux systems contain many text files. They include

- configuration files
- log files
- data files
- plus, others

Understanding how to view these files is an important skill.



Cat Command

List the contents of a file on the standard output

Example Usage:

- cat > filename creates a new file
- cat filename1 filename2>filename3 joins two files and stores the output of them in a new file

Options:

- -n: number the output lines.
- -s: suppress repeated output lines that are empty.

Pr Command

Displays two short text files at the same time. You can quickly view the files side by side.

Example Usage:

- pr –m file1.txt file2.txt

Options:

- -m: print all files in parallel, one in each column.
- -l: set the page length.

Grep Command

Help you find a file line (or lines) that contain certain text strings.

Example Usage: grep -i aspen /etc/passwd

Options: -i: ignore case.

Head Command

Displays the first 10 lines of a text file.

Example Usage:

- head –n 5 /etc/passwd

Options:

- -n: the number of file lines to display.

Tail Command

Displays the last 10 lines of a text file.

Example Usage:

- tail –n 5 /etc/passwd

Options:

- -n: the number of file lines to display.

Pager Commands

One way to read through a large file's text is by using a pager. A pager utility allows you to view one text page at a time and move through the text at your own pace. The two commonly used pagers are the more and less utilities.

Example Usage:

- less /etc/passwd

Options:

- Move forward one line: Down Arrow, Enter, e, or j.
- Move backward one line: Up Arrow, y, or k.
- Move forward one page: Space bar or Page Down.
- Move backward one page: Page Up or b.
- Scroll to the right: Right Arrow.
- Scroll to the left: Left Arrow.
- Jump to the top of the file: Home or g

Lesson 3 Review

Grep can find individual lines in a text file

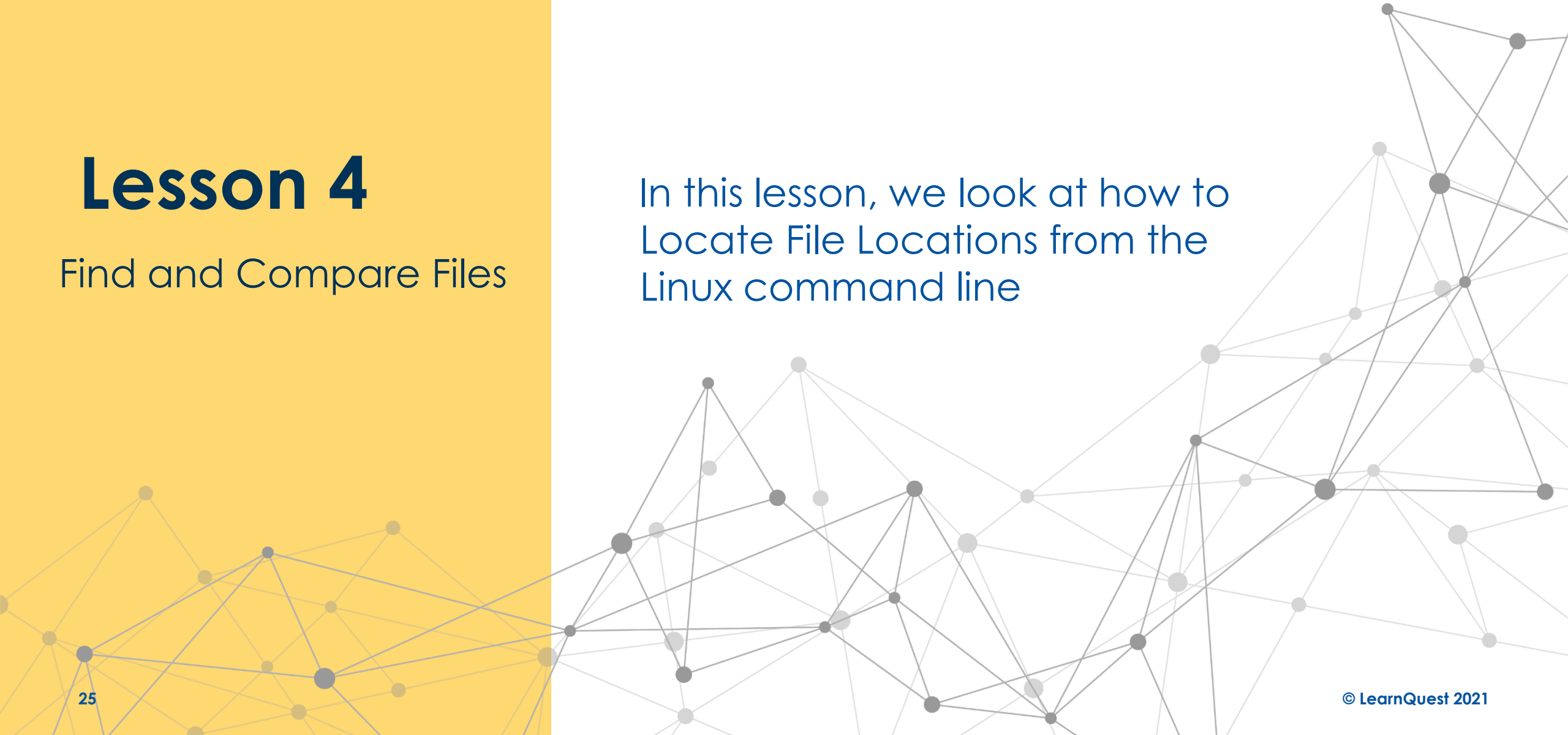
Head can show you n number of rows from the top of a text file

Less is faster than more

Lesson 4

Find and Compare Files

In this lesson, we look at how to Locate File Locations from the Linux command line



Diff Command

Make comparisons between two files, line by line.

Example Usage:

- diff file1.txt file2.txt

Options:

- -q: If files are different, issue a simple message expressing this.
- -r: Compare any subdirectories within the original directory tree, and consecutively compare their contents and the subdirectories as well
- -y: Display output in two columns.

Which Command

Shows you the full path name of a shell command passed as an argument. The which command is also handy for quickly determining if a command is using an alias.

Example Usage:

- which diff
  - /usr/bin/diff

Locate Command

This utility searches a database, mlocate.db, which is located in the /var/lib/mlocate/ directory, to determine if a particular file exists on the local system.

Example Usage:

- locate myProject.txt
  - /home/aspeno/myProject.txt

Options:

- -b: Display only file names that match the pattern and do not include any directory names that match the pattern.
- -i: Ignore case in the pattern for matching file names.
- -q: Do not display any error messages, such as permission denied, when processing.
- -r: Use the regular expression, R, instead of the pattern list to match file names.

Find Command

Allows you to locate files based-on data, such as who owns the file, when the file was last modified, permission set on the file, and so on.

Example Usage:

- find . -name "*.txt"

Options:

- -P: Never follow symbolic links. This is the default.
- -L: Follow symbolic links.
- -name pattern: Returns true if the base of a file name (the path with the leading directories removed) matches shell pattern.
- -gid: Display names of files whose group id is equal to n.
- -perm: Display names of files whose permissions matches mode. Either octal or symbolic modes may be used.
- -size: Display names of files whose size matches n. Suffixes can be used to make the size more human readable, such as G for gigabytes.
- -user: Display names of files whose owner is name.

Lesson 4 Review

Diff command can be used to compare two files

Locate is a simple command to discover the location of a file

The find command has much more control than the simpler pinpoint commands