Undercover Unix

Cybersecurity

Terminal 101 Day 3

Class Objectives

By the end of class today, students will be able to:

- [ ] Create and execute shell scripts using Unix commands.
- [ ] Employ research and documentation to identify new commands for unfamiliar situations
- [ ] Use sed to make substitutions to a file.
- [ ] Use awk to make changes to a file.

Warm-Up Activity

In this activity, you will review the topics covered last class.

Instructions sent via Slack

Suggested Time:
7 Minutes

Your Turn: Warm-Up

Instructions:

You've just been given a zip file that includes a Docs folder filled with folders and files from a user's personal machine.

Use the command line to complete the following tasks:

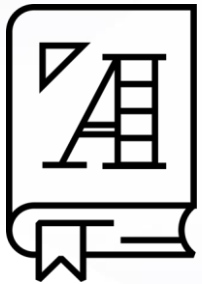
a. Create two subfolders within the Music folder: one called FLAC and the other called MP3.
b. Write a command that finds all .mp3 and .flac files and copies them into their respective subfolders.
c. Write a command to count the number of .mp3 and .flac files in each folder.

7 Minutes

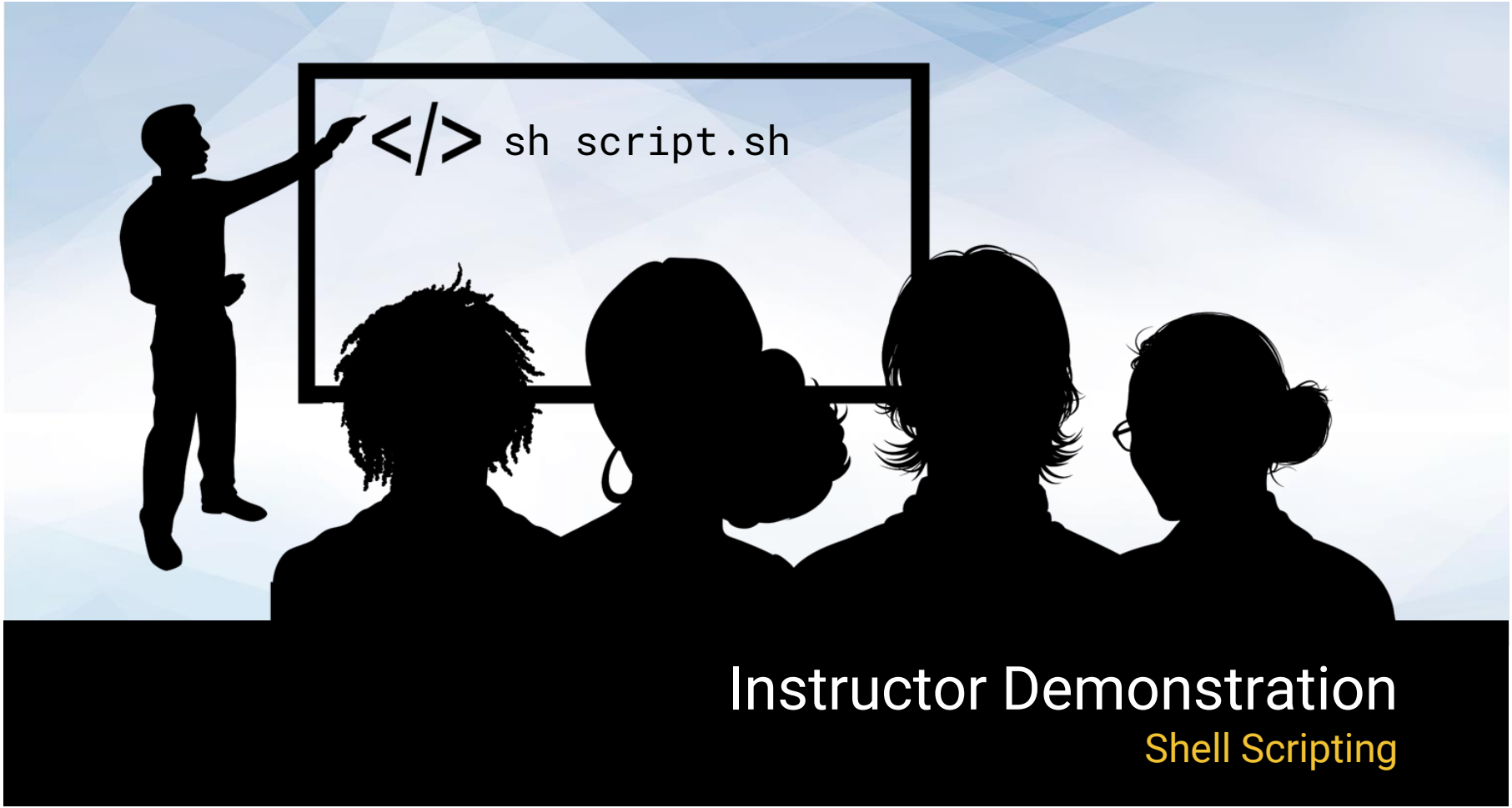
Time’s Up! Let’s Review.

Warm-Up Activity

Introduction to Shell Scripting



A shell script wraps up a series of commands into a file (a script) and runs all those commands when you run the script.



Instructor Demonstration

Shell Scripting

Activity: My First Shell Script

In this activity, you will practice creating shell scripts.

Instructions sent via Slack

Suggested Time:
15 Minutes

Your Turn: My First Shell Script

Instructions

Create a shell script which accomplishes the following actions:

- Creates a folder called Summary.
- Combines all files in the Files folder provided into a single file called `MySummary.txt.`
- Moves the `MySummary.txt` file into the Summary folder.
- Previews the first ten lines of content.

Run the shell script.

Hints:

- To start you will first need to create a file. Use the terminal and type `touch script.sh.`
- Consider testing the individual commands of your shell script in the command line. And then copy and paste into the shell script. Start where you feel comfortable before transitioning to what's new.

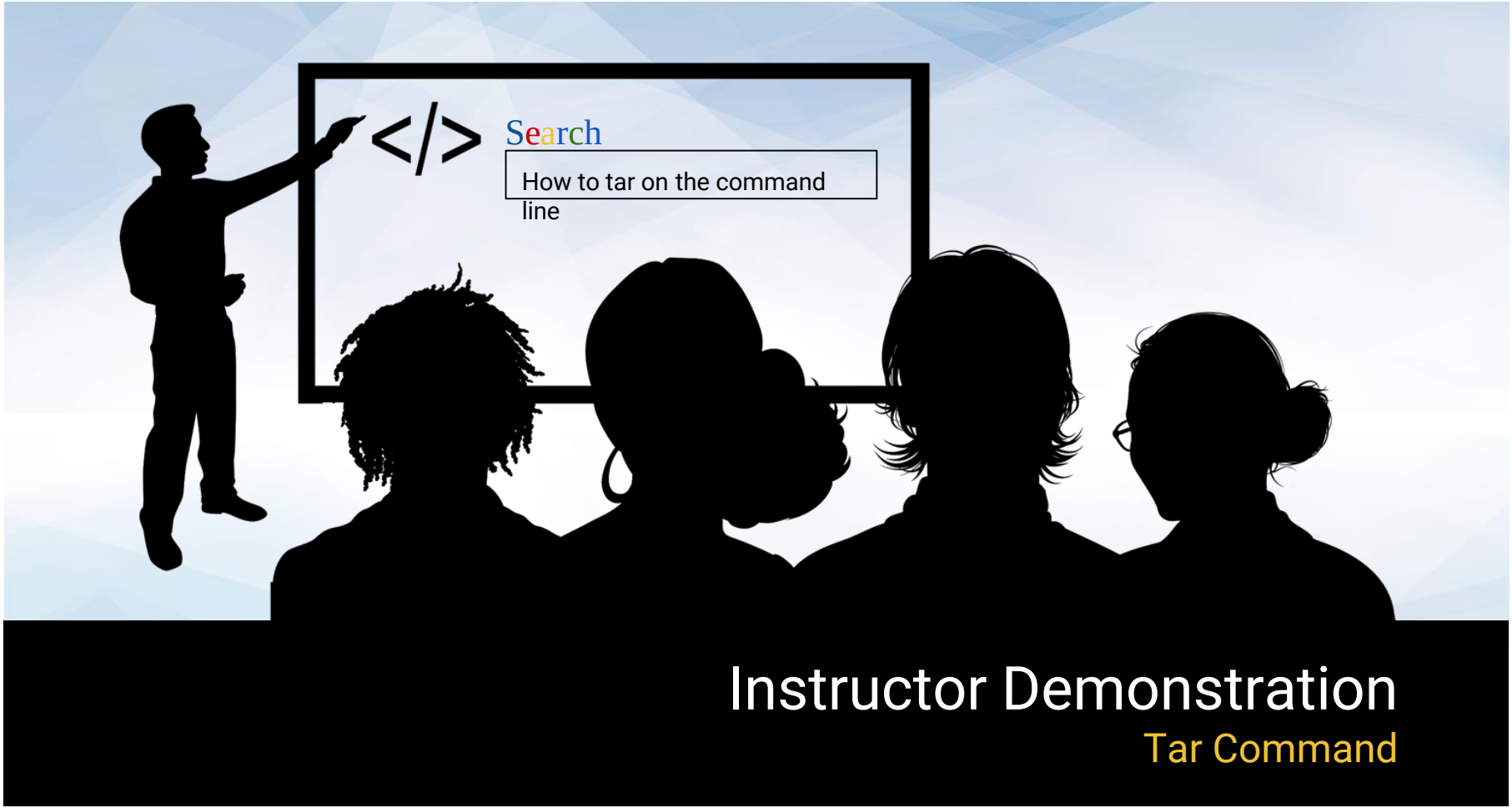
Time’s Up! Let’s Review.

My First Shell Script

Learning New Commands and echo

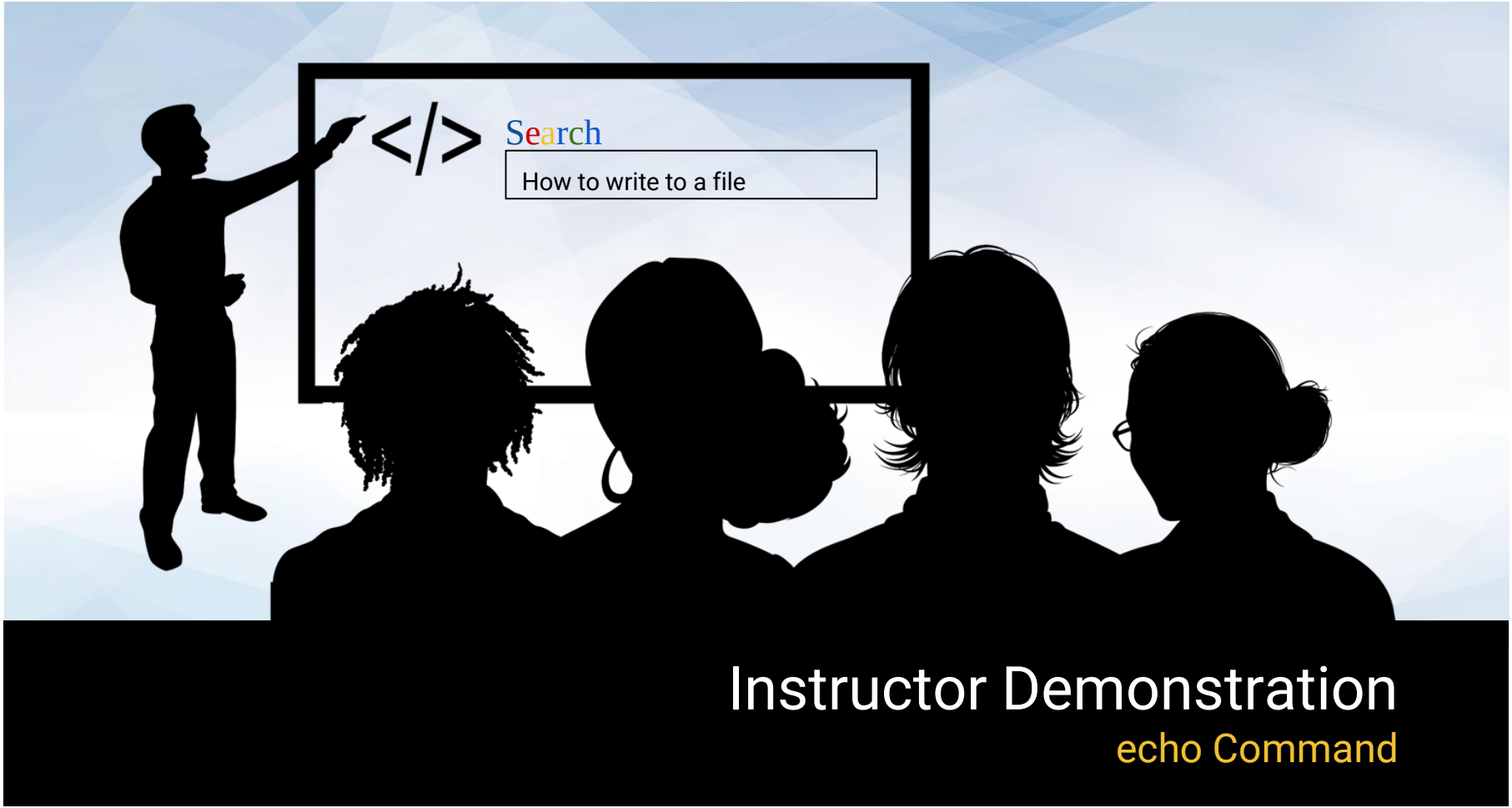
As security professionals, you'll be expected to expand your knowledge set on the fly.

It is important to get comfortable with the process of learning new commands.



Instructor Demonstration

Tar Command



Instructor Demonstration

echo Command

Activity: Writing to a File

In this activity, you will use their research skills to master new Unix commands for writing and appending to files.

Instructions sent via Slack.

Suggested Time:
10 minutes

Your Turn: Writing to a File

Instructions:

Using the Readings folder as a starting place, create a shell script that accomplishes the following:

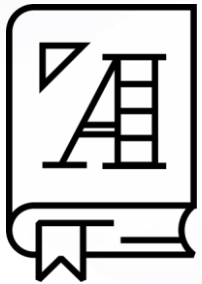
- Create a new file called `MyFile.txt`.
- In the file, add the following line of text: "Hey there! This is my sentence".
- Clear the `Pride.txt` file and insert a line of replacement text.
- Add a line of text to the bottom of the `Alice.txt` file.

10 Minutes

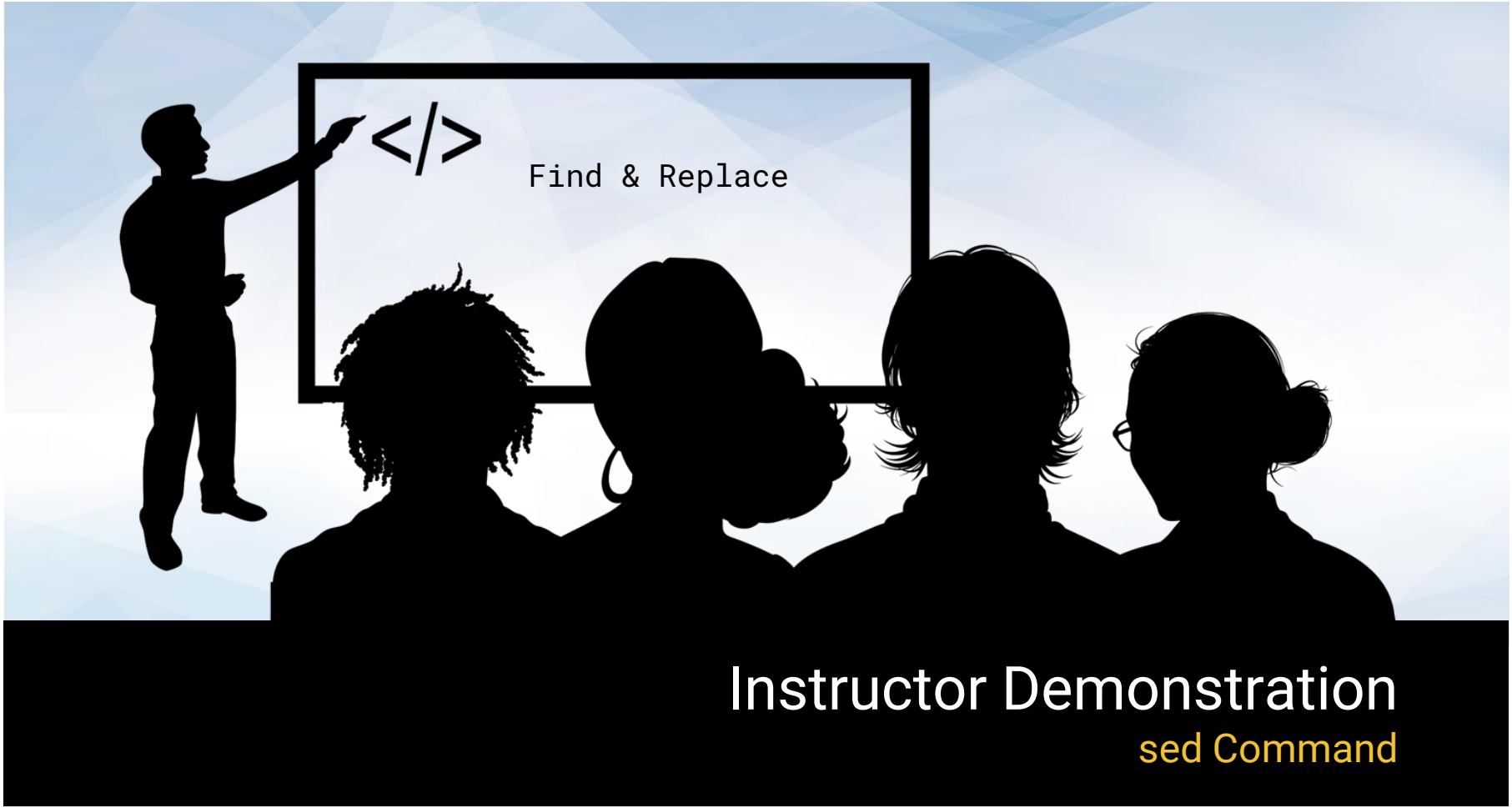
Time’s Up! Let’s Review.

Writing a File

Introduction to sed



sed (Stream Editor) also for quick edits and substitution for text files on the command line.



Instructor Demonstration

sed Command

sed Find and Replace

```
sed 's/clank/clink/g' othello.txt > othello_new.txt
```

sed is the first part that runs the program.

s is 'substitute' option for sed, telling it that specific substitutions are coming next.

/clank/ is the pattern that sed will search for.

/clink/ is the string that will be substituted whenever the first pattern is matched.

g means to apply this substitution 'globally'

Instructor Demo: Regular Expressions

`/abc/` Matches anytime these three letters are shown in this exact order.

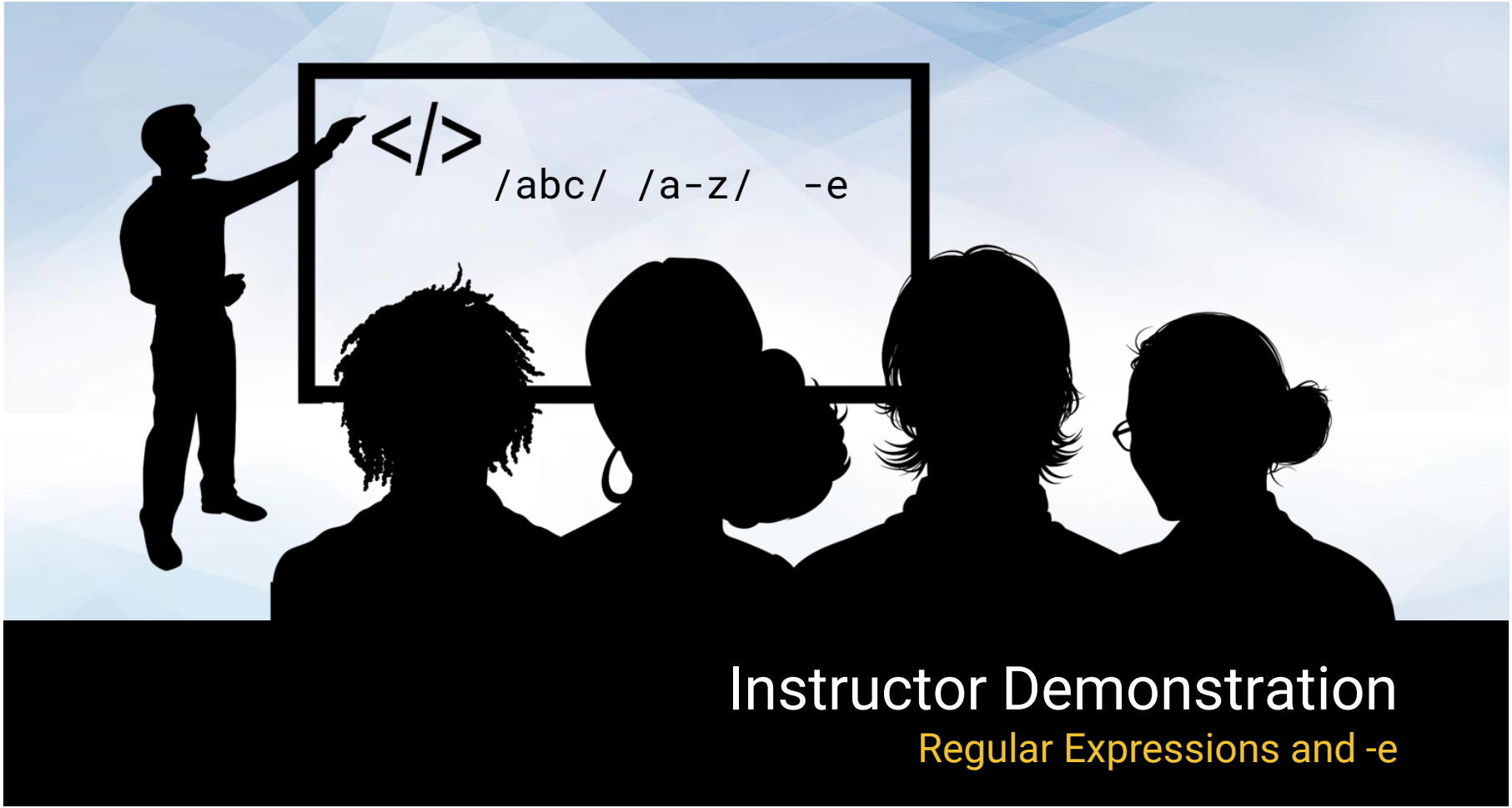
`/a-z/` Matches any string of lowercase letters.

What will each of these commands show?

```
sed 's/[a-z]/Z/g' sed.txt

sed 's/[A-Z]/z/g' sed.txt

sed 's/[0-9]/#/g' sed.txt
```



Instructor Demonstration

Regular Expressions and -e

Activity: He sed, She sed

In this activity, you will use the sed command to alter a text file.

Instructions sent via Slack.

Suggested Time:
10 Minutes

Your Turn: He sed, She sed

Instructions:

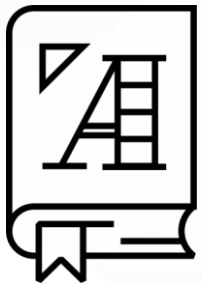
Using the lights.txt file create a sed command that accomplishes the following:

- Changes `dark` to `bright` where it makes sense to do so.
- Changes `bright` to `dark` where it makes sense to do so.
- Changes `but` to `and` in the first sentence.
- Changes on to `off` at the end of the second sentence.

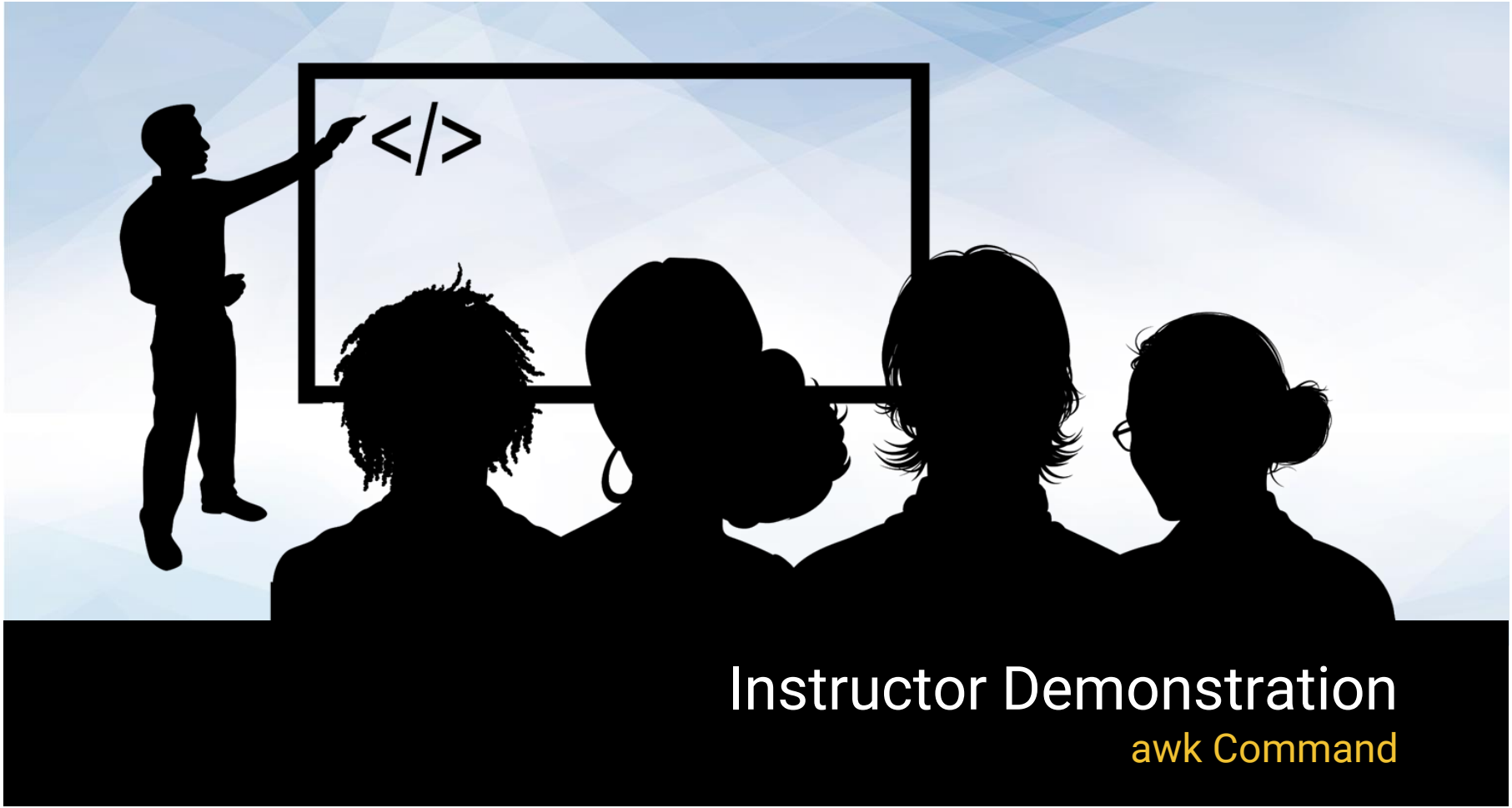
10 Minutes

Take a Break!

awk



awk is a programing language designed specifically for processing text which allows quick and useful tasks in the command line.


Instructor Demonstration
awk Command

Activity: Start gAWK-ing

In this activity, students will make some changes to a file.

Instructions sent via Slack.

Suggested Time:
20 minutes

Time’s Up! Let’s Review.

`Start gAWKing`

Class Objectives

By the end of class today, students will be able to:

- Create and execute shell scripts using Unix commands.
- Employ research and documentation to identify new commands for unfamiliar situations
- Use sed to make substitutions to a file.
- Use awk to make changes to a file.