**Task 5: Writing a Shell Script**

Shell Scripting Basics

* Creating a Bash script using the nano text editor.
* Adding the shebang line (#!/bin/bash) to specify the script interpreter.
* Using comments (#) for documentation within scripts.

Variable Management

* Defining and using variables in Bash scripts.
* Assigning command output to variables using backticks (e.g., `command`).
* Concatenating strings to create dynamic variables (e.g., combining file paths and file names).

Command Execution and Output

* Using echo to display variable values for debugging.
* Creating scripts that include reusable file paths and formatted dates for file naming.

File Permissions

* Using chmod to modify file permissions:
  + Granting full access to the user (chmod u=rwx).
  + Granting read and execute permissions to groups and others (chmod go=rx).
  + Adding or removing specific permissions (e.g., chmod o-r or chmod o+r).

Archiving and Compression

* Using the tar command to:
  + Create archive files (-c option).
  + Specify output file names (-f option).
  + Compress files while archiving using the -z option.
* Understanding the importance of option order in tar commands.

File Paths

* Using the tilde (~) for referencing the home directory.
* Creating variables for commonly used paths and reusing them in the script.

Script Testing

* Running scripts using relative paths (e.g., ./myscript).
* Debugging scripts by verifying outputs and permissions.

Backup Automation

* Creating scripts to back up directories to a single compressed archive file.
* Incorporating timestamps into file names to prevent overwriting.

Error Avoidance

* Commenting out unnecessary or debugging lines (e.g., echo commands) to maintain a clean script.

Understanding Legacy Commands

* Working with tar, including its quirks and historical context.