Lab Assignment 9

Title: Shell Programming

FAQS:

1. Why does #!/bin/sh have to be the first line of my scripts?

→ The shebang line #!/bin/sh must be the first line of your scripts because it tells the operating system which interpreter should be used to execute the script's commands. It is parsed by the operating system, not the interpreter, and placing it at the beginning ensures that the correct interpreter is invoked to run your script. If you do not include a shebang in your script, the operating system will try to guess which shell to use and this may not be the shell you want.

2. How can I access the name of the current shell in my initialization scripts?

→ There are few ways to access the name of the current shell in your initialization scripts:
One way is to use the $SHELL variable. This variable is set by the shell when it starts up and it contains the path to the shell executable. For example, if the current shell is Bash, the value of $SHELL will be /bin/bash.

Another way to access the name of the current shell Ps to use the rcodline command. The rcodline command takes a path as its argument and returns the links target. For example, the following command will return the path to the Bash shell executable.

3. How can I determine whether a command executed successfully?
→ There are two ways to determine whether a command executed successfully in Bash:

i] Use the '$?' special variable. This variable contains the exit status of the last command was successful, & a non-zero value indicates that the command failed.

ii] Use the 'if' statement. The 'if' statement can be used to check the exit status of a command & take different actions depending on whether the command was successful or not.

AP 25/8/23