Bash scripting :- task 2

Why we are using if to check the specific condition

If The statement runs according to the value we assign to the variable.

#!/bin/bash

count=100

if [ $count -eq 100 ]

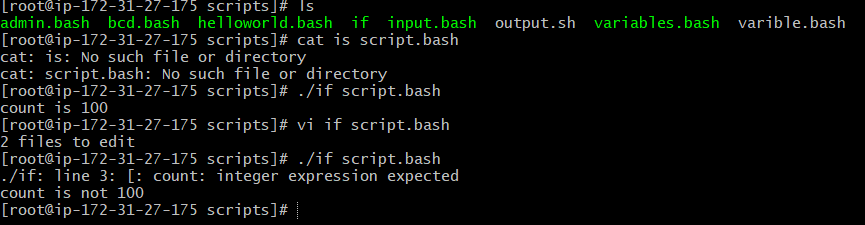
then

echo count is 100

else

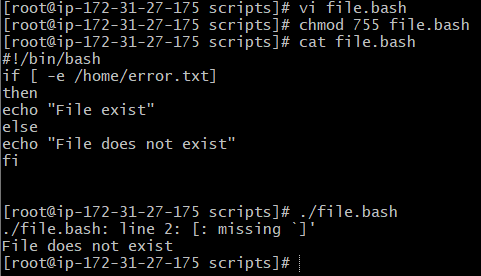
echo count is not 100

fi

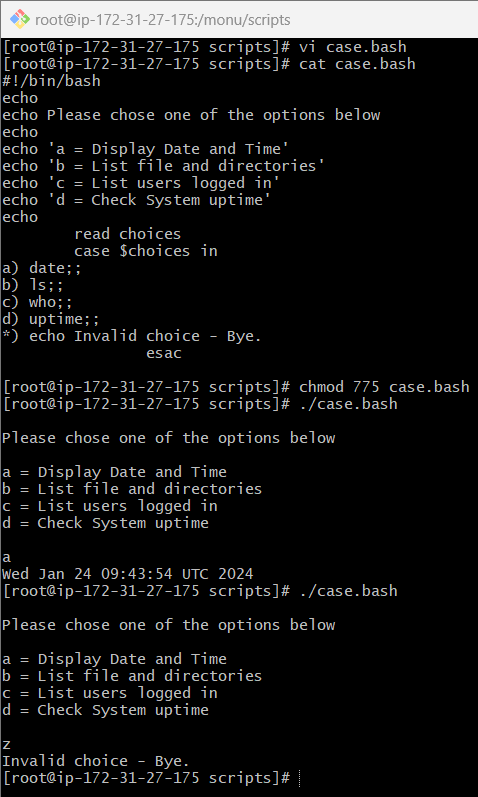


In case to find out if a file is available and there is an option to verify the file exists is -e

syntax: if [ -e directory or file path ]   
 then   
 echo “statement-1”   
 else   
 echo “statement-2”   
 fi







Case :-

This is used in situations where there are multiple options that users will select, and the result will depend on which option they choose.

The script commands below are assigned to variables

#!/bin/bash

echo

echo Please chose one of the options below

echo

echo 'a = Display Date and Time'

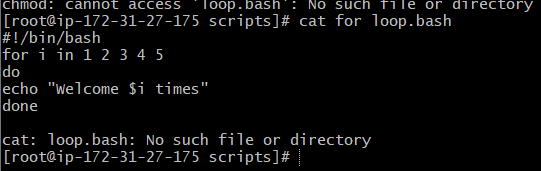
echo 'b = List file and directories'

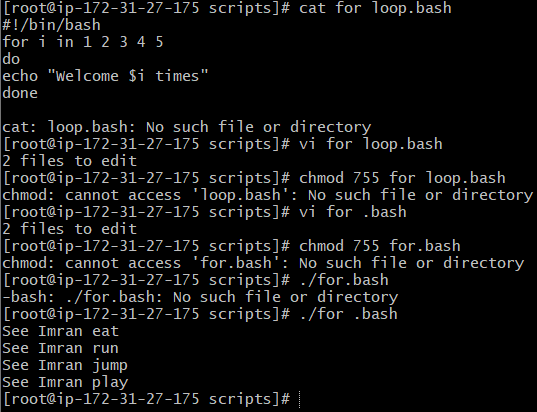
echo 'c = List users logged in'

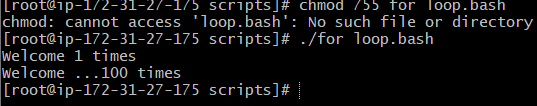
For loop scripts:-

For loop keeps running until specified number of variable

varible = green,blue,red (then run the script 3 times for each color)



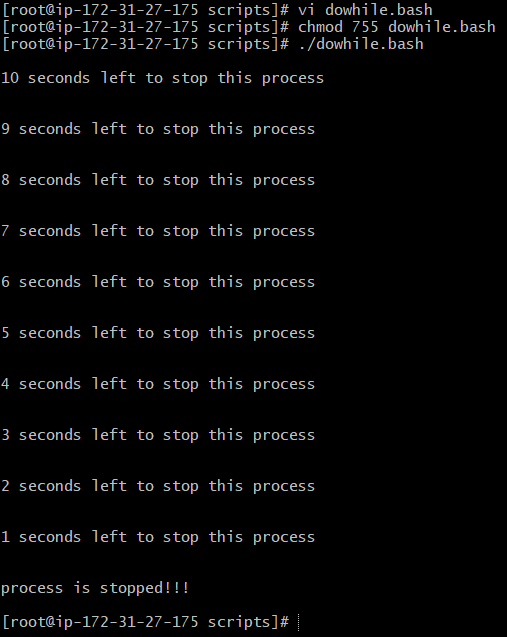




**Do While Loop:**   
• It will execute the statements n number until the given condition   
becomes true   
 -lt is less than

-gt is Greaterthan

-eq id equal



Exit status :-

By using the echo $to know the previously executed command is successful we have to use echo $

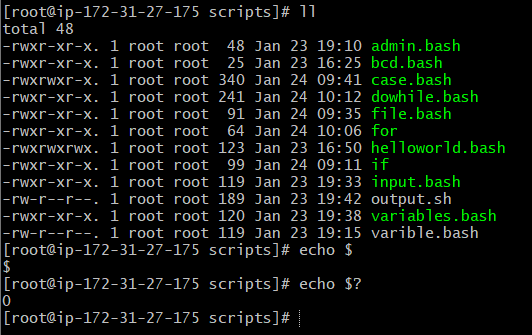
Every time a command is executed in Linux it will have a status is associated with it.

\*) 0 = ok or successful

\*) 1 = Minor problem

\*) 2 = serious problem

\*) 3-255 = Everything else



Cornjobs:-

Cornjob is used to run the script without any manual intervention

