Assessment:  
Write a bash script that writes a directory and changes permissions and edits stuff.

Case/set locale dependent [0-9 a-z A-Z]  
Classes locale independent [[:lower:]]

|  |  |
| --- | --- |
| **File Globbing** |  |
| \* Zero or more characters | ls D\* |
| ? single characters | ls D?esktop |
| **e.g.** | ls | grep [D] lists all files/dirs containing “D |
| Assigning | Log\_File=mylog1 |
| Reading | echo ${Log\_File} |
| Cmd Expan | version=$(uname –r) echo $version |
| retain assignment after closure | export myVar=”hello world” |
| ‘ ’ Single quotes | Stuff is echoed as is |
| “ ” double quotes | Echo replaces some special characters |
| Echo “some data” 1> newFile | overwriting data into a file (1 exists by default) |
| Echo “some data” 1>> newFile | Adds extra data into file without overriding |
| Echo “some data” 2>> newFile | Error logging |
| cp | copy |
| mv | Move (rename) |
| Rm –rf | Delete a file –recursive - force |
| mkdir | Make directory |
| rmdir or rm -r | Delete directory or delete all files and dir |
| export -f | export functions |
| ${VarName} Variable substitution | $(CommandName) Command Substitution |

|  |  |
| --- | --- |
| **Examples** |  |
| $ ls \*.? | All files/dirs that have a period with one letter after it (typically just “.d”) |
| $ more [A-Z]\* | Gives all contents of everything |
| $ ls /etc/[!a-m]\* | Lists all files not beginning with “a-m” |
| $file /usr/bin/\*X\* | Lists all files containing an “X” |
| $ ls [a-z]\*[0-9] | Lists everything containing lowercase letters and then numbers |
| $ ls –l .[!.]\* | Long lists everything that contains one period |
| $ ls –d /etc/\*.d/\* | Lists contents of dirs that are named ending in a “.d” |

|  |  |
| --- | --- |
| **Symbolic/Hard Links** |  |
| Stat filename | Gives more info like iNode #, access rights, last modification date etc. |
| iNode | If iNode # of two files is same (hard link), they are updated together regardless of location  This is a unique ID of file  Contains: read/write/iNodeChange date, hard drive location. |
| ln file1 file2 (file2 doesn’t exist yet) | Links file1 and file2 so that if 1 is updated, 2 is also updated |
| Ln –s file1 file2 | Creates a symbolic link between files (useful on different machines) |
| Symbolic Link | Separate machines, different iNodes for files, but one file contains the iNode of the other file |

|  |  |
| --- | --- |
| **History Commands** |  |
| !n | nth line in history |
| !-n | nth prev line |
| !! | last line |
| !prefix | most recent line with specified prefix |
| !^xx^zz | only last line with string xx replaced by zz |
| !n:s/xx/zz/ | nth line with string xx replaced by zz |
| !\* | all args of last command |
| !$ | last arg of last command |
| !^ | first arg of last command |
| !:n | nth arg of last command |
|  |  |

|  |  |
| --- | --- |
| **Shell Variables** |  |
| set | displays all variables |
| env | displays exported variables |
| unset | delete variables |
| echo $PATH | gives search path (not inc. current dir) |
| alias la=”/bin/la –lart” | creates alias called la |
| set –o/+o optionName | switch on/off a specific option |
| export PATH=”${PATH}:filePath | add paths to startup |
|  |  |

|  |  |
| --- | --- |
| **File System Access** | **r-4 w-2 x-1** |
| chmod change mode | change access rights (user/group/other) |
| chown change owner | change main owner name |
| chgrp change group | change main group name |
| sticky bit | allows only the owner/root to delete or rename a file |
| stat –c %a fileName | stat(file inf) –c%a(format octal nums) of this file |

|  |  |
| --- | --- |
| **Other** |  |
| | <- Pipe Symbol | stdout becomes stdin |
| noclobber | disallows the ability to overwrite files  can be looked over by using >| instead of > |
| wc -L | word count (line count) |
| who|tee fileName|wc –L | adds output of “who” to a file and outputs linecount |
| ls –L|grep –v “D” | all files not with “D” in name |
| ls –L|grep –c file | amount of occurrences of “file” |
| ls –l|grep –i documents | ignores case and still searches for Documents |
| ls –l|grep –n fileName | gives line number of occurrence |
| cat fileName | cut –d”,” –f3,4,6 | display file contents delimiting “,” (split data after each “,”) and showing only columns 3,4,6 |
| cat fileName | sort -ru | sorts lines of text backwards without duplicates |
| cat fileName | uniq -u | removes same consecutive entries |
| cat fileName | sort | uniq -u | …So we use this instead  -d shows which ones were duplicates  -c shows number of occurrences of the word |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **Shell Scripts** |  |
| #! /bin/bash | Tells Linux to run the script in a bash shell |
| if [ input ] then “stuff” fi | must have “spaces” between [ ] and end in fi |
| -gt | greater than |
| -lt | less than |
| -ge | greater or equal to |
| -le | less or equal to |
| -eq | equal to |
| -ne | not equal to |
| read userInput | read allows us to assign a user input to a var |
| read value  case $value in  “a”) echo “letter;;   1. echo “number”;;   \*} echo “default”  esac | works very similarly to a java “switch” but must have ;; after each case except last |
| declare -a arrayVar=(“Adam” “Bob” “Chester”)  counter=1  for i in “{arrayVar[@]}”  do  “Person ${counter} is ${i}  ((counter++))  done | this is how you declare an array (“-a”)  use the “@” symbol to loop through the array |
| $compgen –c –a –k –b –A unset | all commands, aliases, keywords, built-ins, function, unset something |
|  |  |
|  |  |