Assignment Man Pages

Name:

It is important to research topics found in this assignment.
Download this document, complete the assignment and save as a .pdf. After you complete this assignment, transfer the file to the Linux web server, make it visible and send the URL to your instructor. Email your instructor if you have any questions judy.pino@sfcc.edu.

Student Learning Objectives:
After successful completion of this exercise a student should be able to:

- Use Linux *man* command to browse the Linux documentation
- Research the Linux documentation to learn about a command
- Explain how the shell uses the PATH variable to locate commands
- Explain the shell and its relationship with the kernel

Resources:
https://en.wikipedia.org/wiki/Man_page
UNIX tutorial http://www.ee.surrey.ac.uk/Teaching/Unix/
http://www.linfo.org/path_env_var.html

The shell is a command interpreter; it starts the program call from the command line and passes instructions (arguments) to the program. The shell is also a programming language. It can run files of commands when issued a command to the shell.

A utility program sometimes referred to as a command performs a task that is related to the operations system. Linux distributions include many utilities. Examples of utilities are *cp, ssh* and *df.*

UNIX commands are generally implemented as files representing executable programs. They are mainly written in C. When a command is run it is loaded into memory and the CPU starts executing the instructions contained in the program. UNIS is case sensitive. Unlike Windows, UNIX doesn't require command names to have an extension (.exe, etc.).
The shell command is invoked the moment you log in and runs until you log out.   There are external and internal commands on a UNIX system. External commands exist on the disk as separate files, but there is also a separate set of commands that are built into the shell executable. The command cd is an example of a built in executable. You can use the command *type* to check if a command is built in. For example, try the command *type echo.* What was the output?

The shell maintains a variable named PATH (all caps, not to confuse with *path*) in its own environment (known as environment variable). PATH is set of a list of colon-delimited directories. When a command is entered the shell looks in each of these directories to locate the file with that name. The executable is located and executed.   The command *whereis* can be used to find the location of the executable file for a command.

The *which* command tells the directory that contains the command. For example, run the command *which grep*. Where is the command *grep* located? You can also try the command *whereis grep*.

Another command is *info*. At the UNIX command prompt type, *info grep*. This information is organized in nodes. You can use the commands, *n, p, u* and *l* to navigate the info documentation.

There are three commands that provide information about other commands – *which, whereis* and *type*.

5 points each (50 points):

Command	**Explain, observe**
man man	Brings up a manual page for the man command.
man –k cron	Brings up all commands that have manual pages that contain the string “cron”
man grep	Brings up a manual page for the grep command.
info grep	Brings up an alternative manual page for the grep command.
whatis which	Brings up all commands that have whatis pages that contain the string which.
whatis whereis	Brings up all commands that have manual pages that contain the string whereis.
whereis ls	Returns a path to ls.
which ls	Returns a path to ls.
echo $PATH	Prints the PATH environmental variable.
which echo	Returns a path to echo.

(20 points)
Find out whether these commands are internal or external: *echo, date, pwd, bash, kill* and *ls*.

echo: internal
date: external
pwd: internal
bash: external
kill: internal
ls: external

(30 points)
Research the UNIX shell and kernel. How does the shell work with the kernel? Why is Linux popular and often used in academia?

The shell receives commands from the user which tell it which programs to run, which then tell the kernel to interact with the hardware in some way, such as retrieve a list of the files in a directory from the disc if the command entered was ls. Linux is popular in academia because it is designed to be easily customizable.