Introduction to UNIX/LINUX
Part 2

Moving around the file system

Command	Description
mkdir	make a directory
cd *directory*	change to named directory
cd	change current directory to your HOME directory
cd ~	change current directory to your HOME directory
cd ..	in Unix, (**..**) means the parent of the current directory, so typing cd .. will take you one directory up the hierarchy
cd ~bob	change the current directory to the user bob's home directory (if you have permission)
cd .	in UNIX, (**.**) means the current directory, so typing cd . means stay where you are
pwd	display the full pathname of the current directory

Viewing and editing files

Command	Description
cat	display the content of a file
cat –b	display the line number with the content of a file
more	progressively dump a file to the screen: ENTER = one line down SPACEBAR = page down q=quit
less	display a file a page at a time
head	show the first 10 lines of a file
head -n	show the first n lines of a file
tail	show the last 10 lines of a file
tail -n	show the last n lines of a file
vi	edit a file using the vi editor

Cat Command

- *cat* is one of the most frequently used commands on Unix-like operating systems.
- It has three related functions with regard to text files: displaying them, combining copies of them and creating new ones.
- cat's general syntax is

```
cat [options] [filenames] [-] [filenames]
```

The square brackets indicate that the enclosed items are optional.

Cat Command (continue)

- Reading Files

  The most common use of cat is to read the contents of files,   and cat is often the most convenient program for this purpose.

  ```
  cat file1
  ```

- Reading large file that doesn’t fit to the monitor screen

  ```
  cat file1 | less
  ```

- Redirecting the content from one file to another

  ```
  cat file1 > file2
  ```

Cat Command (continue)

- Concatenation (stringing together)

  cat file1 file2 file3 - the contents of each file will be displayed on the monitor screen

  cat file1 file2 file3 > file4 – the contents of each file will be redirected to the new file

  cat file1 file2 file3 | sort > file4 – the output of all files is filtered (sorted) and redirected to the new file

Cat Command (continue)

- File creation

  The third use for cat is file creation. For small files this is often easier than using vi, gedit, or other text editors. It is accomplished by typing cat followed by the output redirection operator and the name of the file to be created, then pressing ENTER and finally simultaneously pressing the CONTROL and d keys.

  ```
  cat > file1
  ```

Moving, renaming, and copying files

Command	Description
cp *file1 file2*	copy file1 and call it file2
mv *file1 file2*	move or rename file1 to file2
rm *file*	remove or delete a file
rm –r *directory*	recursively remove a directory and its contents
rmdir *directory*	remove an empty directory
rm –d *directory*	delete an empty directory
wc –l [-w], [-c] *file*	count number of lines [words], [characters] in file

Redirection Input and Output

In UNIX, we can redirect both the input and the output of commands

- the symbol > redirects the standard output of the command to the file, where it is written and saved, or to a device (such as a printer, where it is printed).
- the symbol < redirects the standard input of the command from a file

Exercise

Create the file called **list2** containing the following fruit: orange, plum, mango, grapefruit. Read the contents of **list2**

Use the cat command to join (concatenate) **list1** and **list2** into a new file called **biglist**.

Redirecting the Input

- We use the **<** symbol to redirect the input of a command from a file.

Exercise

- Use the command **sort** (alphabetically or numerically sorts a list) to sort fruits from the file **biglist** using the symbol **<** . The sorted list will be output to the screen.

Redirecting the Output, Input. Pipes

Command	Description
command > file	redirect standard output to a file
command >> file	append standard output to a file
command < file	redirect standard input from a file
command1 | command2	pipe the output of command1 to the input of command2
cat ***file1 file2 > file3***	concatenate file1 and file2 to file3
sort	sort data

Pipes

- The "|" (pipe) operator sends the standard output of one command to another command as standard input. It allows commands to be combined in a sequential order
- any number of commands can be connected in sequence, forming a pipeline
- all programs in a pipeline execute at the same time
- complex operations are easily performed by piping commands

Exercise

Using pipes, print all lines of **list1** and **list2** containing the letter 'p', sort the result, and write the result to a file**pwords.txt**

Listing Directory Contents

- **$ ls –l**

```
drwxr-xr-x    4   janna     user       1024   Jun 18 09:40 UNIX_Class
-rw-r--r--      1   janna     user      767392 Jun  6 14:28 scanlib.tar.gz
^ ^  ^   ^       ^       ^           ^           ^        ^           ^         ^
| |  |    |        |       |           |           |        |           |          |
| |  |    |        |      owner    group      size   date       time      name
| |  |    |          number of links to file or directory contents
| |  |      permissions for world
| |  permissions for members of group
|  permissions for owner of file: r = read, w = write, x = execute -=no permission
type of file: - = normal file, d=directory
```

- **ls -a**   List the current directory including hidden files. Hidden files start with ".“
- **ls -l**   List all the file and directory names in the current directory using long format