Quick Guide to Unix Commands #1--Draft

(or .1% of unix)

You are the 006266 visitor to this page

Introduction

Your Free Student Account can be used for much more than just email. It can be used to read newsgroups, to make your own homepage (for those familiar with Mosaic and Netscape a.k.a..the World Wide Web), to talk to others on the internet, and to transfer files froom one computer to another. However, before you try these tasks, it is necessary to know the very basics of the unix operating system.
This guide is intended as a **VERY VERY BASIC** reference sheet for complete beginners in unix. It does not even begin to go into the functionality of unix, but rather will give you enough commands to move around in unix, create and remove files, and organize your files into directories. After you have mastered all of these skills, further unix commands will be within reach.
Remember that all of these commands are case sensitive.
IF THEY ARE IN LOWER CASE LETTERS, THAT MEANS THEY MUST BE IN LOWER CASE LETTERS.

Making and Editing a File

The first thing that you need to know about unix is that it deals with files and these files come in many different forms. These files could be pieces of email which you have saved, articles which you have saved, or just files that you created with an editor (which is like creating a document with a word processor). Other files are created by certain programs in unix, such as elm, pine, and nn, and are used to configure those programs correctly. Finally, there are some files that are needed for your account to function and should **never** be removed.
As an example, there are a couple of different ways to save a piece of email as a file and the method of how to do it depends on the mailer you are using. To find out how to do it in the various mailing systems, such as elm or pine or mail, refer to the related documentation.
To make a file on your own, you need to use an editor. Two common editors are vi and pico and each of these editors have some documentation on them. vi is more powerful and has more documentation, but it is more complicated. Pico is much simpler but it is far less powerful and less documented. The only documentation (that I know of) is **http://ux1.cso.uiuc.edu/~kundert/josh/docs/picodoc.html**This documentation is unofficial. For beginners, pico is recommended because the added functionality of vi isn't needed until you become more unix experienced.

First Needed File Commands

These are the first commands that you need to know in order to look at your files, both as a list and the contents of each, and how to remove them.

ls

This command lists the contents of your working directory. There are several "add ons" to this command which will change either the contents of the listing or the format of the listing. Typing: **ls -a**, for example , will list all of your files including the "dot" files, and it might look like this:

```
ux1>ls -a
.               .newsrc         94              boston.ng
..              .newsrc.bak     BlueData        chiphish
.cshrc          .nn             Mail            chiwaukee
.elm            .oldnewsrc      News            ftphish
.gopherrc       .rnlast         acappella       license.txt
.login          .rnsoft         bosngdir        mbox

ux1>
```

There are a number of files that start with a "." that won't normally show up with when you type ls. An example of such a file is your .newsrc file. A second add on is -s. Typing: **ls -s** will give you a listing of your files along with their size in kilobytes. Another add on is -l. Typing: **ls -l** will give you a "long listing," which will look something like this:

```
ux1> ls -l
total 62
-rw-r-----  1 kundert     29928 Sep 20  1994 94
-rw-r-----  1 kundert       820 Nov 22  1993 BlueData
drwx------  2 kundert       512 Oct  3  1993 Mail
drwxr-x---  3 kundert       512 Sep 28 11:46 News
-rw-r-----  1 kundert         0 Oct  7 15:23 acappella
-rw-r-----  1 kundert      1070 Sep 26 22:03 bosngdir
-rw-r-----  1 kundert      1226 Sep 26 22:03 boston.ng
-rw-r-----  1 kundert       801 Sep 19  1994 chiphish
-rw-r-----  1 kundert      1040 Sep 19  1994 chiwaukee
-rw-r-----  1 kundert      4375 Sep 19  1994 ftphish
-rw-r-----  1 kundert      7480 Nov 10  1993 license.txt
-rw-------  1 kundert      4351 Jan 28 15:25 mbox
drwxr-xr-x  7 kundert       512 Aug 31  1994 public_html
-rw-r-----  1 kundert      1325 Mar 23 16:37 robin
-rw-r-----  1 kundert        31 Oct  1 18:45 sch
```

The only advantage of this form of listing is that it gives you the file access modes (these are important for when you want to set up a homepage, but don't worry about this right now).
All of these add on's can be combined, so one might type:
ls -as
to list all of your files and find their size.

more

This command displays the contents of a ascii file (a plain text file--not a binary program). For example, if you had a file named test, and you wanted to see its contents, the command would look something like this:

```
ux1> more test
```

The contents of the file will be displayed one screen at a time. To go onto the next screen hit either the spacebar or the return button. To quit in the middle of a file hit the letter: "q".

rm

The rm command stands for remove and it is the means to permanently delete your files. To remove a file you type:

```
ux1> rm filename
```

If you are in a directory and want to remove all the files,type:

```
ux1>  rm *
```

and it will delete everything, but be very careful WHICH directory you are in when you do this command or you may delete files that you wanted. To be safe while doing this procedure, you could type:

```
rm -i *
```

and the system will prompt you before each file is deleted.

Directories and Moving/Renaming Files

Directories are the way to organize your files. They are your computer "filing system." You can create directories within directories. These commands will allow you to move back and forth within directories, to move files into and out of directories, and to create/remove directories.

cd

This command will change the working directory. For example, to change into the News/alt/history directory from your home directory (your home directory is the one you are automatically put in when you log onto your account), you would type:

```
ux1>cd News/alt/history
```

There are some additions that you should also know with reference to the command cd.

cd ~

This will take you to your home directory as will plain cd. You can use this to change from a deep subdirectory to a completely different directory (i.e.--from a directory within another directory to a totally different directory). For example, you could change from News/alt/history directory to the public_html directory by typing:

```
ux1>cd ~/public_html
```

from within your News/alt/history directory.

cd ..

This will take you to the directory just above the one you are in. For example, if you were in the News/alt/history subdirectory, typing:

```
ux1>cd ..
```

would put you in the News/alt subdirectory.

cp

This command is used to copy files to other filenames or to copy them from one directory to another directory. To copy a file to another filename (the file test to test2, for example), you type:

```
ux1>cp test test2
```

To copy a file from one directory to another (for example, the file test from your home directory into the News/alt/history directory) you would type:

```
ux1>cp test News/alt/history/
```

If the directory you want to copy to isn't a subdirectory of your working directory, then you will have to go back throught the home directory using the ~/directory/subdirectory/ method. (like in the cd command.) Another useful point to know in copying files to or from directories is that you can copy all the files at one time using a * formation with the command. For example to copy all the files from one directory(1) to another directory(2) (for example, copying the entire News/alt/history directory to the News/soc/history directory), you would type:

```
ux1>cp * ~/News/soc/history
```

from the News/alt/history directory.

mkdir and rmdir

The mkdir command allows you to make a directory while the rmdir removes a directory. To make a directory (for example, a public_html directory), you would type:

```
ux1>mkdir public_html
```

To remove a directory, it must be empty of all files. After all the files have been removed, you return to the home directory or if it is a subdirectory, the directory right above the directory to be deleted. You would then type (for example, deleting the News/alt/history directory):

```
ux1>rmdir history
```

from the News/alt directory.

mv

This command stands for "move" and it is used to move files within your directory structure or to rename your files. To rename your files, (for example, letter1 to oldletter) you would type:

```
ux1> mv letter1 oldletter
```

The moving of files within directories is exactly the same as in the copy (cp) command.

Further "Help"

For some further help, there are always the man pages. It must be mentioned, however, that most of the man pages were written in the "expert-friendly" mode and may be confusing to new users. To use the man pages on a command (mv for example), you would type:

```
ux1> man mv
```

and it would give you what was in the online manuel. You can always go to these pages for more information.

Go directly to .2% of Unix

Josh Zone/Josh/ History Pages/Tax Survey

---

last updated 11/02/95