

Frequently used UNIX commands

<u>Command</u>	<u>Result</u>
<code>ls</code>	lists files in current working directory
<code>ls *.out</code>	lists all files in current directory that end in <i>.out</i>
<code>ls -l</code>	lists files giving details including file size
<code>pwd</code>	displays full-path name of your current working directory on screen (stands for "present working directory")
<code>cd dirname</code>	changes directory to <i>dirname</i>
<code>cd ..</code>	changes directory to one above the current directory
<code>cd</code>	with no argument, takes you to your home directory
<code>mkdir dirname</code>	creates new directory <i>dirname</i>
<code>rmdir dirname</code>	removes empty directory <i>dirname</i>
<code>cp filename newname</code>	makes a copy of <i>filename</i> with the name <i>newname</i>
<code>cp ../filename .</code>	copies <i>filename</i> in directory one tier above to current directory giving it the same name
<code>mv filename newname</code>	renames <i>filename</i> to <i>newname</i> (mv stands for "move")
<code>cat filename</code>	displays contents of <i>filename</i> on screen
<code>more filename</code>	displays <i>filename</i> contents one screen at a time (Enter key scrolls through file by line; space bar scrolls through by screen)
<code>head filename</code>	displays first 10 lines of <i>filename</i> on screen
<code>tail filename</code>	displays last 10 lines of <i>filename</i> on screen
<code>rm filename</code>	deletes <i>filename</i> without double check (rm stands for "remove")
<code>grep string filename</code>	displays lines from <i>filename</i> containing <i>string</i> on screen
<code>./ filename &</code>	runs job <i>filename</i> in the background
<code>qsub filename</code>	sends job <i>filename</i> to queue (where qsub is a submission script file in your bin directory)
<code>qstat</code>	lists jobs running on queue
<code>qstat -a</code>	lists only your jobs
<code>qdel job#</code>	deletes job with number <i>job#</i> from queue
<code>ps</code>	lists processes you have running
<code>ps -ef</code>	lists all processes
<code>kill pid#</code>	kills process with ID number <i>pid#</i>
<code>kill -9 pid#</code>	kills (with the "sure kill" signal) process with ID number <i>pid#</i>
<code>kill -kill 0</code>	kills all processes you have running and logs you off
<code>man commandname</code>	displays manual page for command <i>commandname</i>

<code>man -k keyword</code>	lists manual pages for commands related to <i>keyword</i>												
<code>lp filename</code>	prints <i>filename</i> on printer in room 4241												
<code>chmod ### filename</code> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>owner</th> <th>group</th> <th>all</th> </tr> </thead> <tbody> <tr> <td>rwX</td> <td>rwX</td> <td>rwX</td> </tr> <tr> <td>4 2 1</td> <td>4 2 1</td> <td>4 2 1</td> </tr> <tr> <td colspan="3">sum numbers for each of the three categories of user</td> </tr> </tbody> </table>	owner	group	all	rwX	rwX	rwX	4 2 1	4 2 1	4 2 1	sum numbers for each of the three categories of user			<p>changes “read, write, execute” mode of <i>filename</i></p> <p>Example: to set the privileges for <i>filename</i> so that the owner has the ability to read, overwrite and execute the file, the group has the ability to read and execute the file, and everyone else has no access to the file, use the command <code>chmod 750 filename</code>.</p>
owner	group	all											
rwX	rwX	rwX											
4 2 1	4 2 1	4 2 1											
sum numbers for each of the three categories of user													
<code>Ctrl+c</code>	kills current operation												
<code>Ctrl+h</code>	delete (always works even when delete key doesn't)												
<code>Ctrl+d</code>	logout												

Frequently used Vi Editor commands

<u>Command</u>	<u>Result</u>
<code>vi filename</code>	opens file (vi with no argument opens new file)
Two modes in vi editor: insert and command (default is command mode when file is opened)	
<code>i</code>	change to insert mode - in this mode you can add text
<code>Esc</code>	return to command mode
in command mode →	
<code>:q</code>	quit file
<code>:q!</code>	quit file without saving any changes
<code>:w</code>	save file (write)
<code>:wq</code>	save and quit file (write and quit)
<code>x</code>	delete character
<code>/string</code>	search forward for <i>string</i> (n key finds next)
<code>?string</code>	search backwards for <i>string</i> (n key finds previous)
<code>:g/string/s//string2/g</code>	substitute <i>string</i> with <i>string2</i> globally
<code>:line#</code>	jump to <i>line#</i> in file
<code>:\$</code>	jump to last line of file
<code>..,\$d</code>	delete from current line to end of file
<code>..,+20d</code>	delete from current line through next 20