The essential linux commands *terminal (macos) or powershell (windows10)*

```
cd #change directory
cd /usr/local/bin #change to /usr/local/bin
cd .. #moves up one directory
cd - #moves to the previous directory
cd ~ #moves to the home folder
cd ./direc/ #moves to a subdirectory called direc
cd ../direct/ #moves to an upper-level directory called direct
ls #lists files in the directory
ls /usr/local/bin #lists all files in /usr/local/bin
ls –lh /usr/local/bin  #lists all of the files with the file size printed
ls *.jpg  #* is a wildcard, so this call lists all files with .jpg extension
```

The essential linux commands

```
mkdir  #make directory
mkdir ~/fun-stuff  #makes a folder in the home directory called fun-”stuff”
rmdir #remove directory ---should be done very cautiously
rmdir ~/fun-stuff #removes fun-stuff
pwd  #print working directory (tells you where you currently are)
cp  #copy file—syntax# cp file newfile ##there is a space separating file and newfile
cp ~/test.txt test2.txt #copies test.txt into a file called test2.txt in the same directory
cp ~/test.txt ./direc/test2.txt # copies to a file (test2.txt) in a subdirectory (direc)
cp –r ~/direc2/ ./direc/2 #r is a recursive flag and must be used to copy/move directories
mv #move or rename file—syntax is the same as cp
mv ~/test.txt ~/test2.txt #renames test.txt to test2.txt in the same directory
mv ~/test.txt ~/direc/test.txt #moves test.txt to the direc folder
```

The essential linux commands

OKAY this is a useful but dangerous command. It can ruin the whole operating system if not used cautiously. Never use this unless you are absolutely sure about the file you are removing. You cannot recover the files once they are gone.

```
rm #removes file
rm –rf #removes directory (the same as rmdir)
```

The essential linux commands

```
echo #move data (text) into a file or a new file
echo Quigley Down Under >> ~/new.txt #writes to new.txt
echo is a great movie >>~/new.txt #writes a new line to the file
cat # read contents of file
cat ~/new.txt

ssh # opens a secure shell –fancy way of saying connects to an outside server
##you will be prompted for a password
ssh ethaler@eclogite.geo.umass.edu #connect to eclogite
scp # secure copy –move files between a remote and local or two remote machines
scp ~/new.txt ethaler@eclogite.geo.umass.edu:~/ #moves new.txt to the home directory on eclogite
scp –r ~/direc/ ethaler@eclogite.geo.umass.edu:~/ #r is the recursive flag and must be used to copy a directory
scp –r ethaler@eclogite.geo.umass.edu:~/big-things/ ~/bad-thing/ #moves a folder from the server to the local machine
```

Text editors

Once inside a linux shell (via ssh to eclogite), we have multiple options for text editors.

Two most common are:

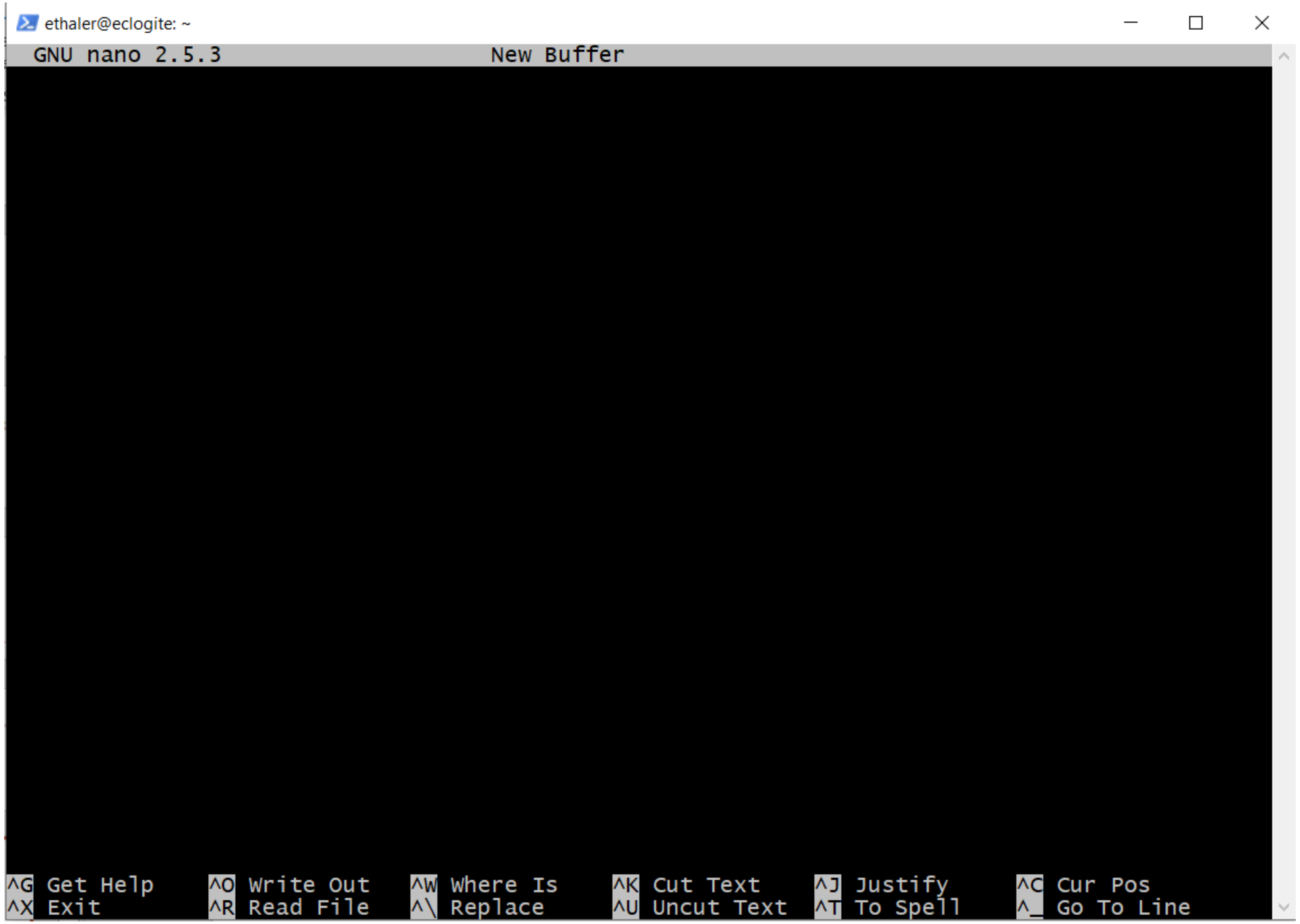
nano

emacs

You can enter either editor by typing their name and pressing enter.

nano is more user-friendly, so let's use it for now.

nano



```
  GNU nano 2.5.3                  New Buffer













^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line
```

The menu at the bottom of the screen will help you navigate.

^ means the ctrl key

^o is save (write out)

^x exits nano

Public keys

- Public keys allow you to connect to remote servers or transfer files to remote servers without having to type your password all the time.

```
#On local machine type:
ssh-keygen

#use default directory
#use no passcode

#copy keygen file to ssh directory on remote server (eclogite.geo.umass.edu in
this case)
scp .ssh/id_rsa.pub user@eclogite.geo.umass.edu:~/.ssh

#Login to remote server
ssh user@eclogite.geo.umass.edu
#Type:
cat .ssh/id_rsa.pub >> .ssh/authorized_keys

##On local machine ssh to remote server
ssh user@eclogite.geo.umass.edu
##You shouldn't be prompted for a password
```

Hostnames

- Hostnames are aliases that you give to certain servers or ip addresses. To save time and mental memory, we can supply hostnames for elogite, geomorph4, and geomorph5. Below is an example for setting a hostname for eclogite

```
#on local machine
nano ~/.ssh/config

Host eclo #alias_to_use_for_ip is whatever you want it to be
HostName eclogite.geo.umass.edu
User ethaler #whatever your username is for the server

#exit nano
ssh eclo #should now connect you to eclogite
```

Hostnames

- Here’s an example for setting geomorph 4.
- Note: we will login to eclogite first and then set the hostname

```
#login to eclogite
ssh eclo
#on eclogite
nano ~/.ssh/config

Host geomorph4 #alias_to_use_for_ip is whatever you want it to be
HostName 172.30.20.24
#I don’t set a username for geomorph 4 or 5 because I login as different users

#exit nano
#try to login to the cosmo user where the LSDtopotools are
ssh cosmo@geomorph4
#password is cosmotools
```

Hostnames

- Here’s an example for setting geomorph 5.
- Note: we will login to eclogite first and then set the hostname

```
#login to eclogite
ssh eclo
#on eclogite
nano ~/.ssh/config

Host geomorph5 #alias_to_use_for_ip is whatever you want it to be
HostName 172.30.23.204

#I don’t set a username for geomorph 4 or 5 because I login as different users
#exit nano
```