

National Institute of Allergy and Infectious Diseases

# Introduction to the Command Line for Biologists

NIAID



National Institute of  
Allergy and  
Infectious Diseases

**R. Burke Squires**  
Computational Genomics Specialist  
BCBB / OCICB / NIAID / NIH

# But...but...but...

- Why do I have to learn about the command line?
- What IS the command line?
- I like my...
  - Mac OS
  - Windows OS
  - GUI (Graphical user interface)
  - ...
- Relax!

# Objectives

- Explain one or more advantages to using the command line for research
- Explain one or more advantages to using the command line for programming
- Be able to get to the command line on your computer
- Understand how to navigate a linux operating system
- Know one or more command to examine and manipulate large files (1 gigbit+)

# Outline

- Motivation
- A brief history of Linux
- Defining some terms
- Command line
  - Getting in, getting out
- Just enough...
  - Navigation
  - Working with Files
  - Grep
  - Pipelining
- Where you can learn more

# Motivation

# Motivation: Why Learn to Use the Command Line

- Enable you to write successful python programs
  - Programs that require you to know where your data file IS, and where you want your output to GO!

# Motivation: Why Learn to Use the Command Line

- Enable you to run the many common bioinformatics software programs
  - Open source bioinformatics software are available for Linux and can be run on a laptop, desktop, or computing cluster (aka super computer)

# Motivation: Why Learn to Use the Command Line

- Enable you to perform analyses on computer clusters
  - Analyze data that is too large to analyze on a laptop or desktop

# Motivation:

## Why Learn to Use the Command Line

- Enable you to use (some of the many) linux programs to do one thing very well and fast
  - Working with text files is especially fast and easy, even for very large files

# Motivation:

## Why Learn to Use the Command Line

- Enable you to compete with most biologists who know how to use the command line and give you a competitive advantage over those who do not

# Motivation: Why Learn to Use the Command Line

- Use the best tool for the job...

# Motivation:

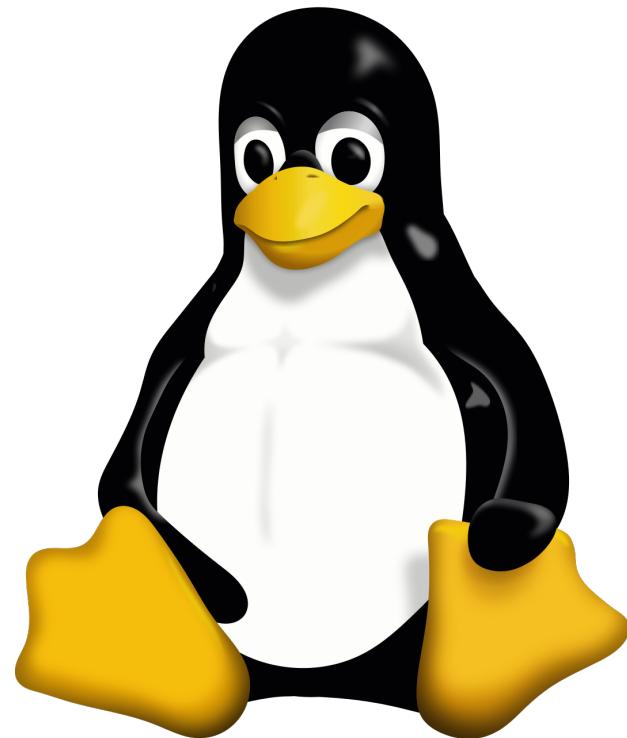
## Why Learn to Use the Command Line

- Enable you to take one step closer to becoming a reproducible scientist
  - How might embracing the command line (and python programming) enable your research to be more reproducible?

# A Brief History of Linux

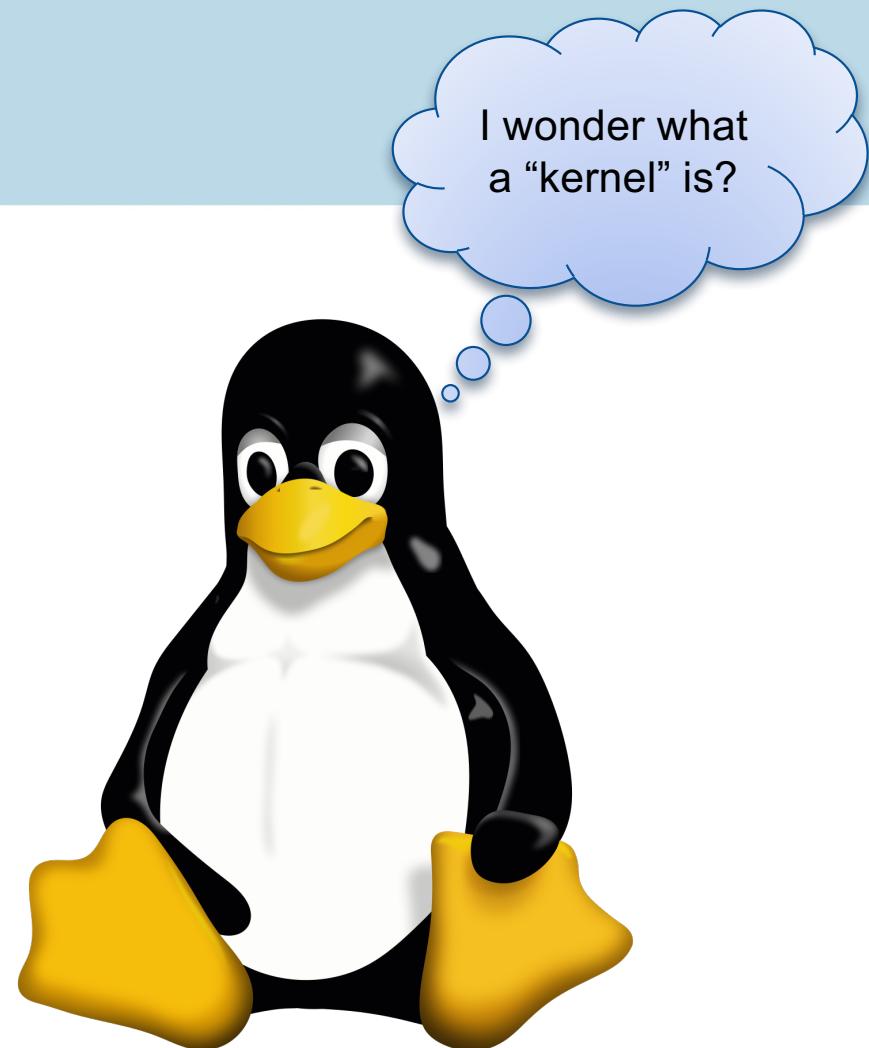
# What is Linux?

- Version 1:
  - An free and open source operating system (OS) that interacts with, and controls, computer hardware; similar to macOS, Windows



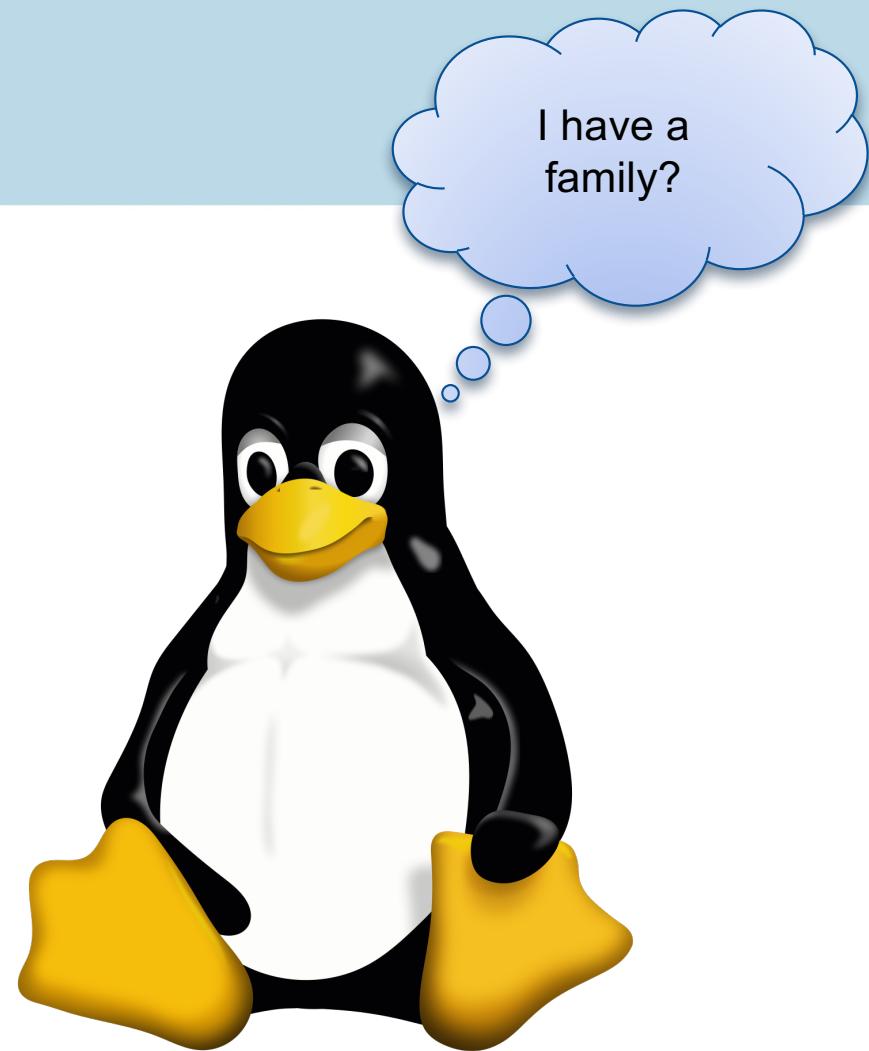
# What is Linux?

- Version 2:
  - A free and open source operating system (OS), **built using the linux kernel**, that interacts with, and controls, computer hardware; similar to macOS, Windows



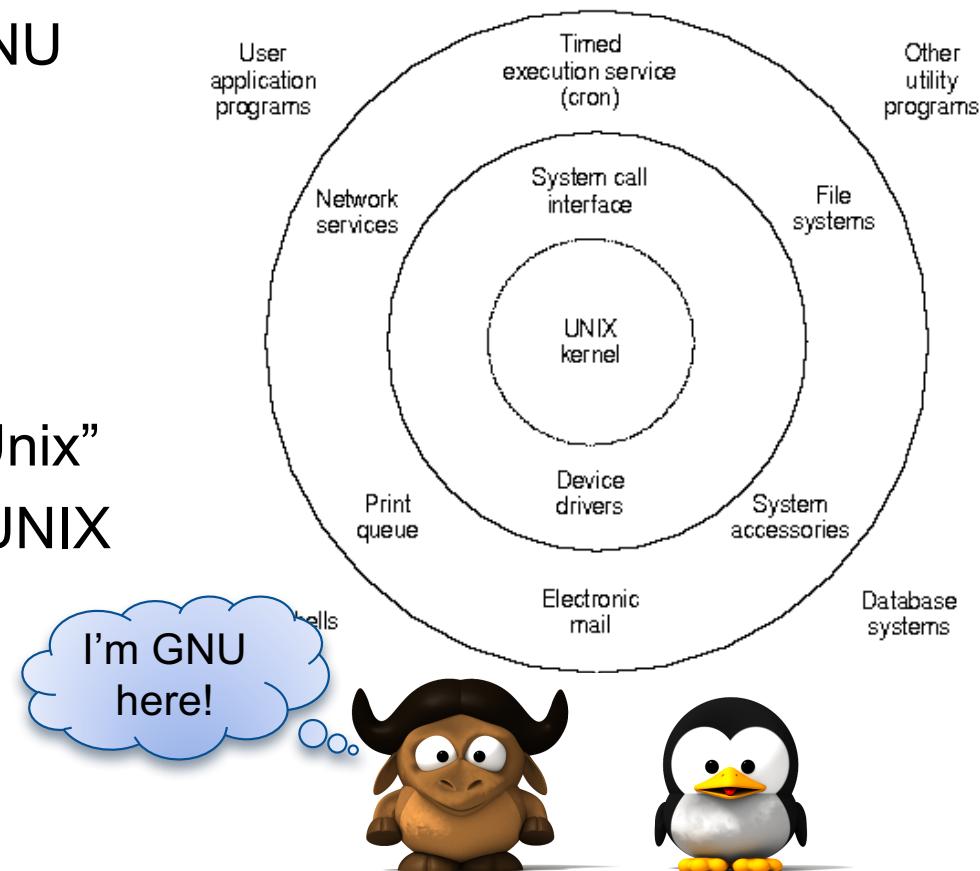
# What is Linux?

- Version 3:
  - “A family of free and open-source software operating system distributions built around the Linux kernel”



# What is Linux?

- What is the rest of that stuff?
  - Programs written by GNU project programmers
  - Shell, etc
- GNU/Linux
  - “GNU” = “GNU is Not Unix”
  - Linux = GNU/Linux = UNIX  
(=> POSIX)



# Why Recreate UNIX?

## UNIX

- Developed at AT&T Bell Labs in 1969
- Commercial
- Capabilities / Philosophy:
  - Scalability
  - Modularity
  - Portability

## Linux (kernel)

- Developed by Linus Torvalds in 1991
- **Open Source**
- Capabilities / Philosophy:
  - Scalability
  - Modularity
  - Portability
  - Free

# Introduction: UNIX vs. Linux vs. macOS vs Windows

	UNIX	Linux	macOS	Windows <10	Windows 10
Started	1969	1991	1987	1993 (1981)	1993 (1981)
Source code	Closed	Open	Mixed	Closed	Closed
Kernel	UNIX	Linux	Linux	Windows	Windows
Interface	CLI (GUI)	CLI, GUI	GUI (CLI)	GUI (DOS)	GUI (CLI, DOS)
64 bit	Yes	Yes	Yes	Yes	Yes
Runs Linux commands	Yes	Yes	Yes	No	Yes (ubuntu on windows)
Protected memory	Yes (1969)	Yes (1991)	Yes (1987)	Yes (1993)	Yes (1993)
MS Office	No	No	Yes	Yes	Yes

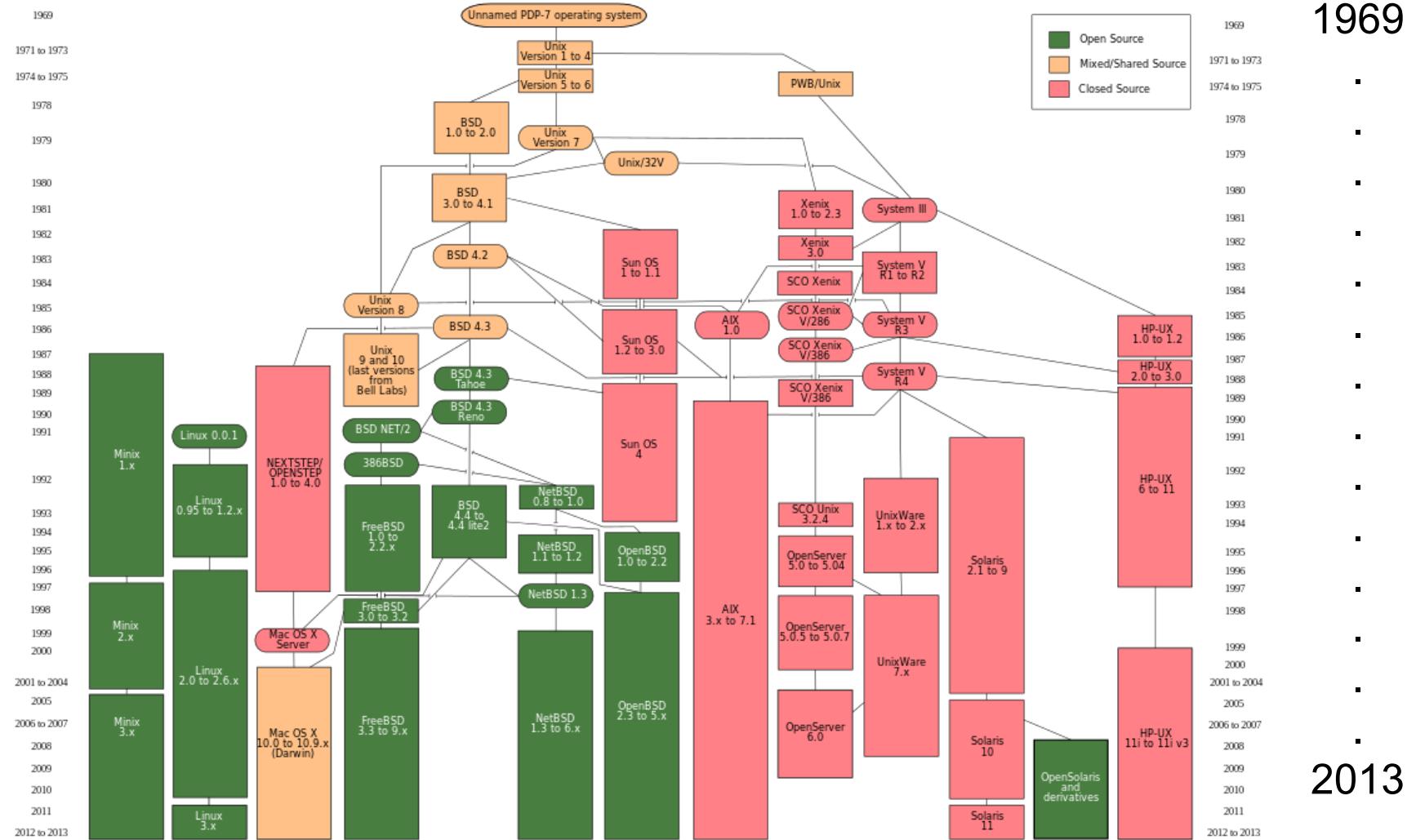


National Institute of  
Allergy and  
Infectious Diseases

For more info: [https://en.wikipedia.org/wiki/Comparison\\_of\\_operating\\_systems](https://en.wikipedia.org/wiki/Comparison_of_operating_systems)

NIAD

# Introduction: Linux / macOS Timeline

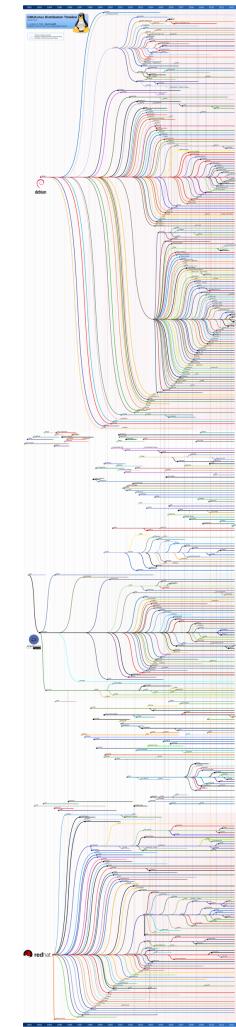
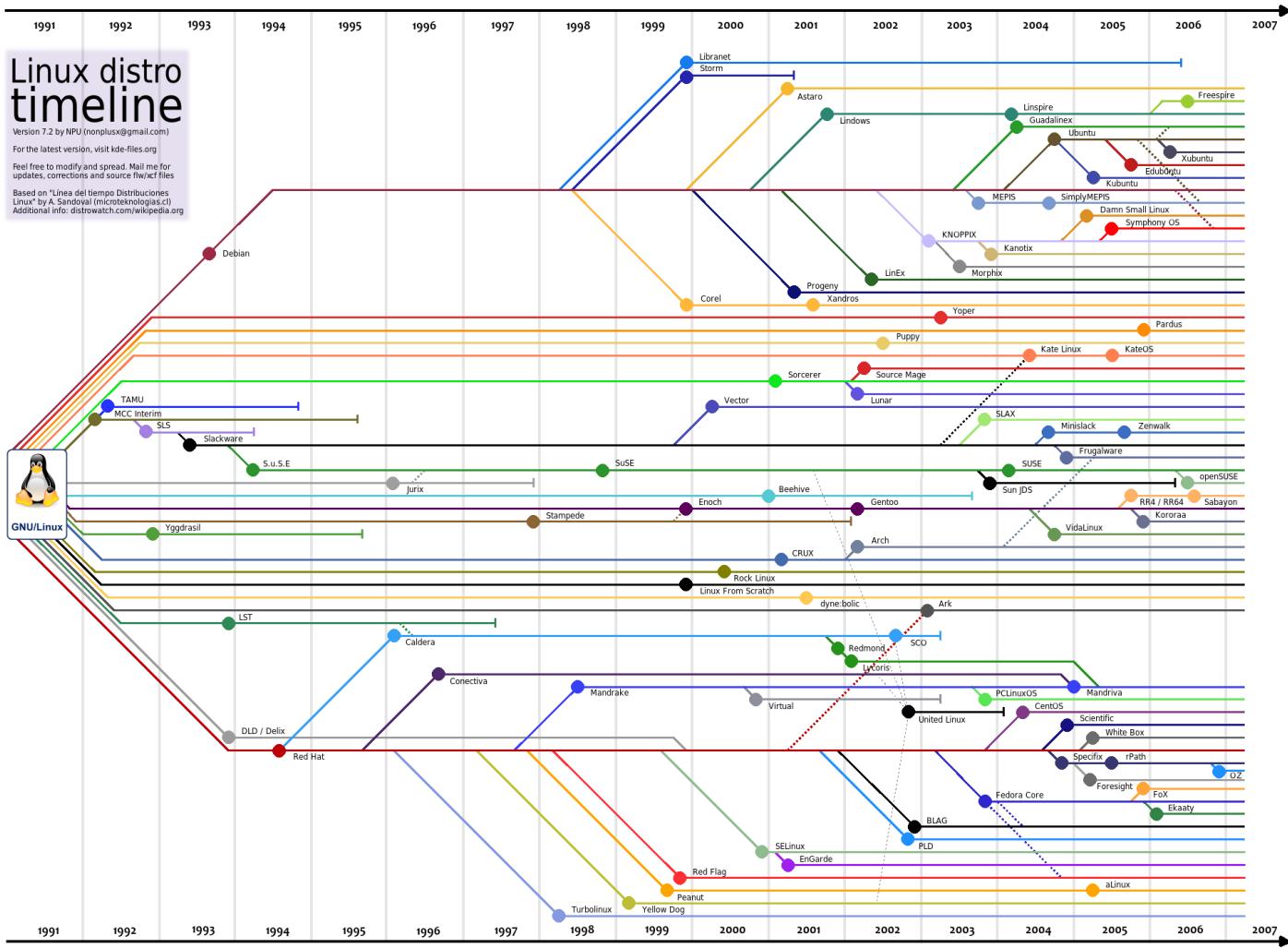


1969

2013

NIAID

# Introduction: Linux Distributions

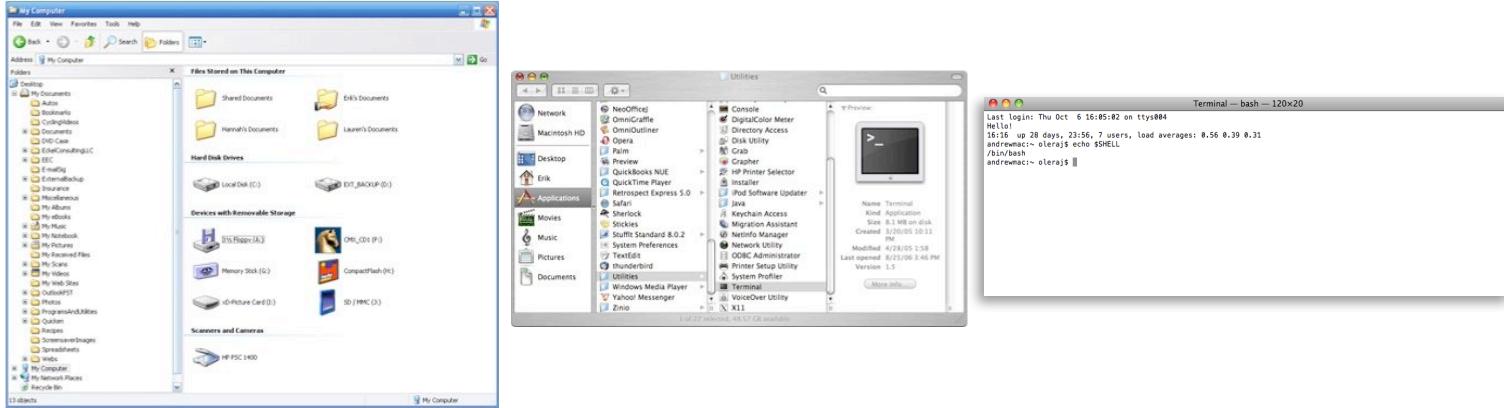


VIAD

# Interacting with Linux

(How do you get to the command line?)

# Interacting with Linux: File Manager / Browser by Operating System



OS: Windows

File Manager: Windows Explorer

macOS

Finder

Linux

Shell

# Interacting with Linux: Accessing the Shell

## Terminal (then)



## Terminal emulator (now)

```
Uname: ls
video
[User@host ~]$ cd ..
[User@host ..]$ cd etc
[User@host etc]$ ls
0.0.10.in-addr.arp  csh.cshrc  gshadow  logrotate.d  osconfig.ini  rmt
address.conf       csh.login   host.conf  logrotate.conf  openoffice  rpc
adttive           csh.logout  hostname  mailcap        opt          screenrc
aliases           db.cache   hosts     mailcap.order  pam.conf   security
alternatives      debconf.conf  hosts.allow  mainname    pam.d      security
apm               debian.version  hosts.deny  mailname    passwd    services
apt              default    hotplug   mailname    passwd-  shadow
asterisk          deflow    hotplug.d  mailname    passwd-  shadow-
at.deny           deluser.conf  hotplug.d  mailname    passwd-  shells
bak.ipkungfu      dhclient.conf identd.conf  mediamr  printcap  skel
bash,bashrc       dhclient-script identd.keyd  mime.types  profile  squid
base-completion   discover.conf  init.d    modules    protocols  ssh
base-completion.d discover.conf-2.6  inittab  modules.conf  python2.3
bind              discover.d   inputrc  modules.conf.old  rc0.d    sudoers
blkid,tar         dkpg      ipkungfu  modules.conf.old  rc1.d    syslog.conf
blkid,tar.old     emc2      issue    modules-utils  rc2.d    terminfo
bzcat            emacs      issue.net  modules      rc3.d    times
chatscripts      emacs22l  issue.net  modules      rc4.d    ucf.conf
chrootkit.conf    email-addresses  kernel-img.conf  mtab    updatedb.conf
complete,tcsh    environment  ldd      mtools.conf  rc5.d    vidario.net.hosts
console          exim4      ld.so.cache  Nutro    rc6.d    wdm
console-tools    fdisk.conf  ld.so.cache  network   modules      rc7.d
cron,v           fonts      locale.alias  network   reportbug.conf  wdm
cron,daily       fstab      locale.gen   networks  resolv.conf  wdm
cron,hourly      groff      localtime  networks  resolv.conf  wdm
cron,monthly     group      logcheck   networks  resolv.conf  X11
cronat           group      login.defs  rsswitch.conf  resolv.conf  xpilot
cronat,weekly    gshadow    logrotate.conf  00BIBdataSources  resolv.conf  resolv.conf.pppd-backup
[User@host etc]$
```

“Terminal” (macOS; a particular terminal emulator)



# Interacting with Linux: Accessing the Shell (with a Terminal Emulator)

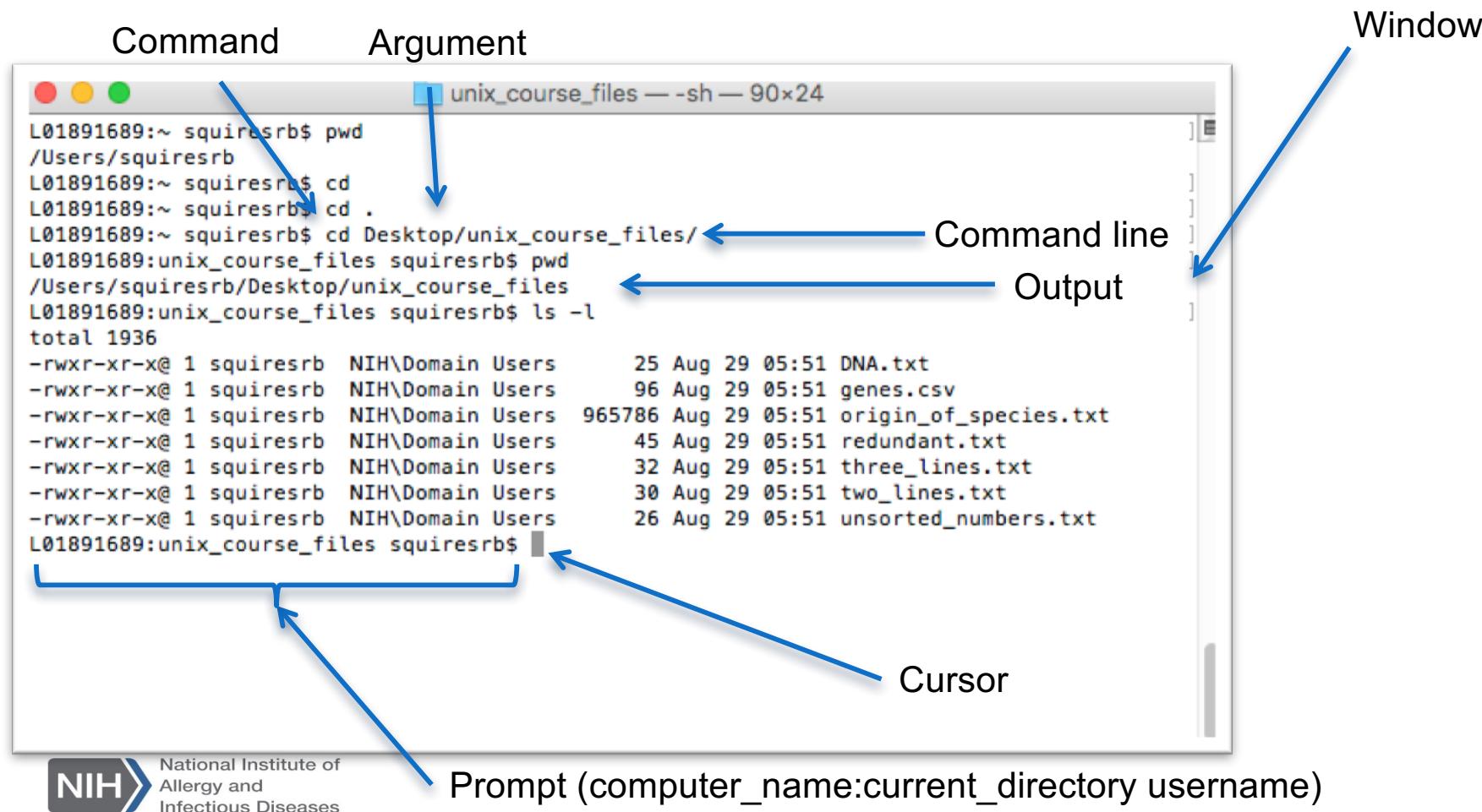
## Mac / Linux

- A Terminal emulator is already installed!
- Mac:
  - Open the Terminal
    - Applications => Utilities => Terminal
- Linux
  - Open the Terminal

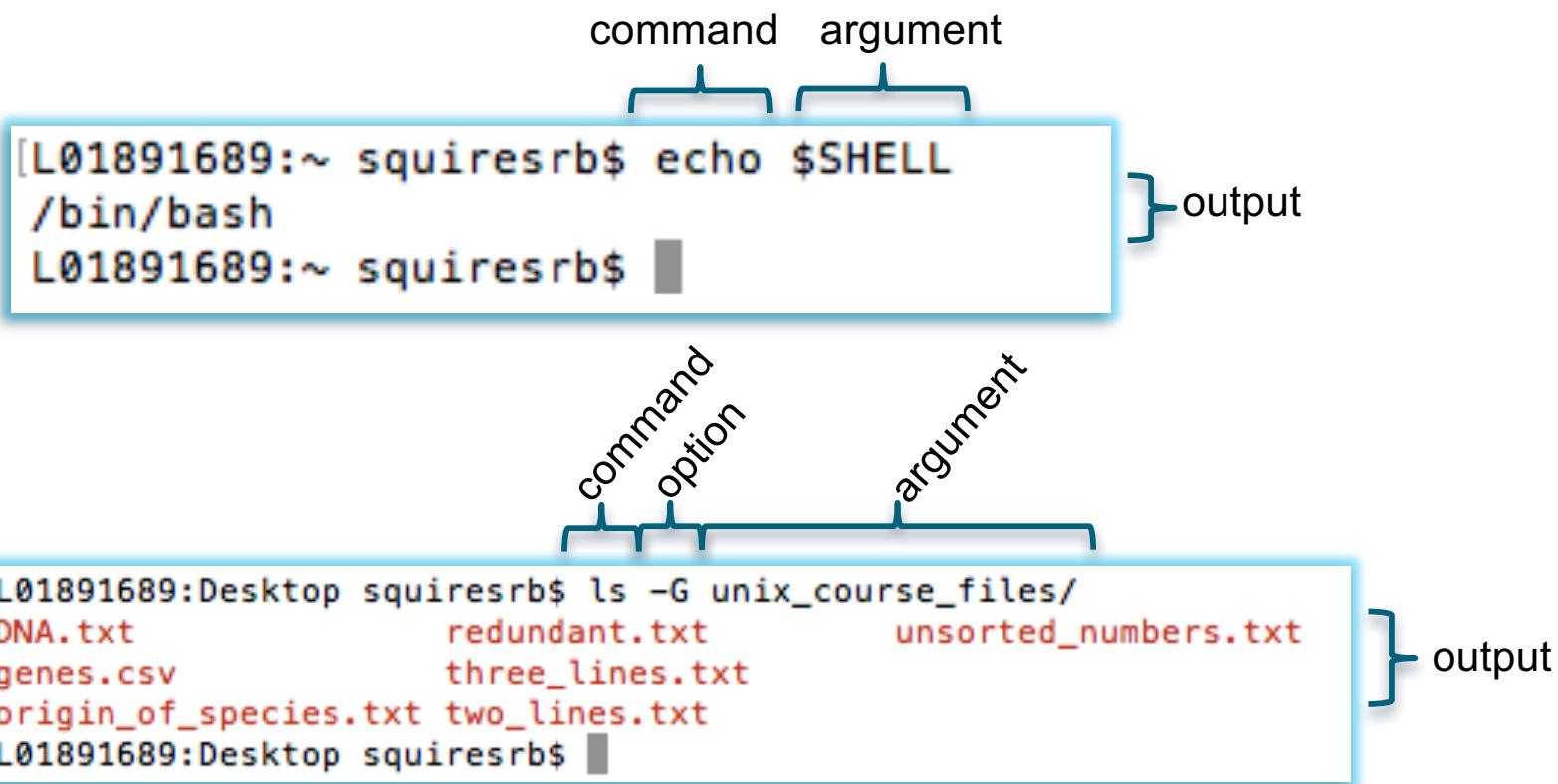
## Windows

- You need to install a version of linux
- Windows 7, 8
  - Install git bash for windows
    - <https://git-for-windows.github.io>
- Windows 10
  - Install Ubuntu through the Windows Store
    - <https://www.microsoft.com/en-us/store/p/ubuntu/9nblggh4msv6>

# Interacting with Linux: Anatomy of the Terminal, “Command Line”, or “Shell”



# How to execute a command



# Help!

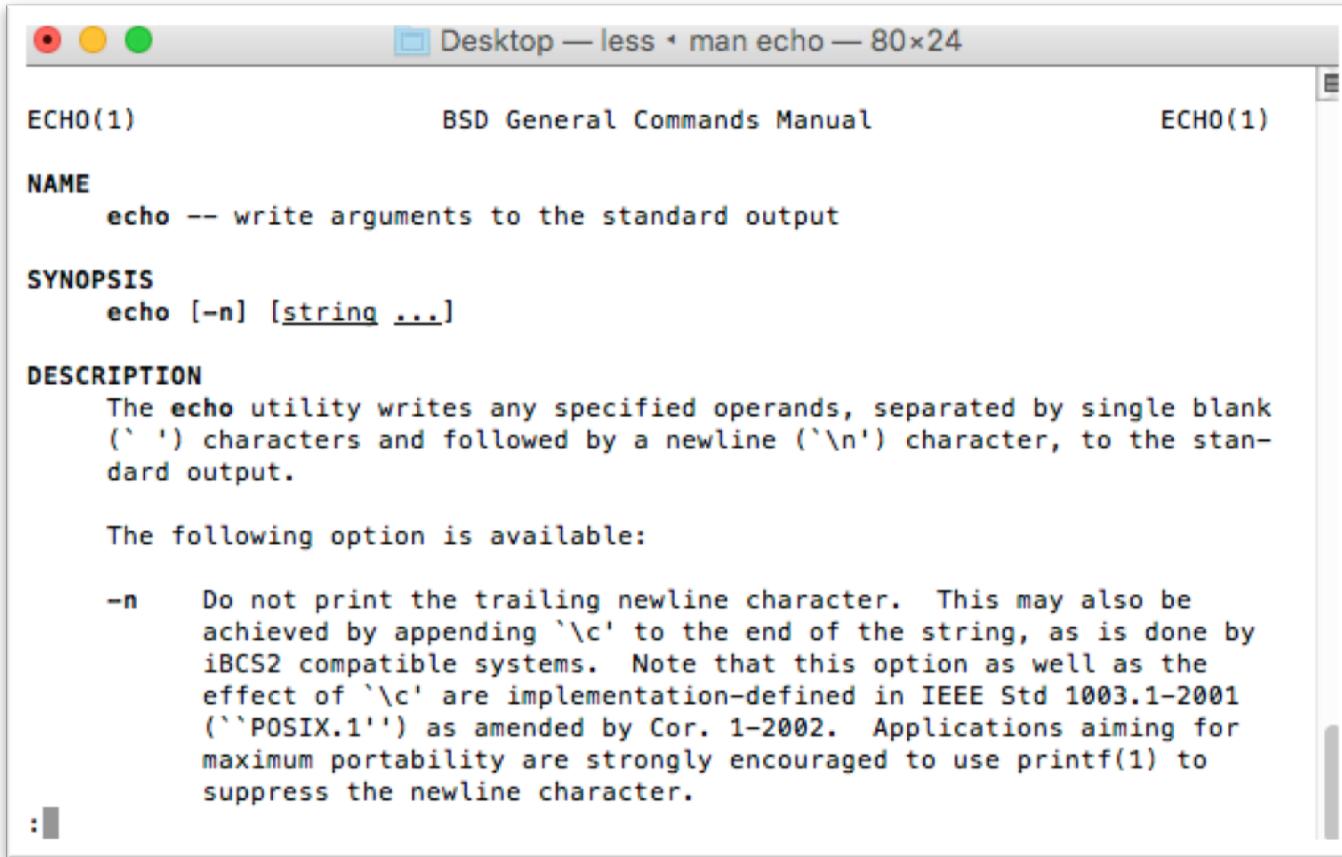
- Whenever you need help with a command type “man” and the command name

# Help!



```
L01891689:Desktop squiresrb$ man
What manual page do you want?
L01891689:Desktop squiresrb$ man echo
```

# Help!



Desktop — less + man echo — 80x24

ECHO(1) BSD General Commands Manual ECHO(1)

**NAME**  
echo -- write arguments to the standard output

**SYNOPSIS**  
echo [-n] [string ...]

**DESCRIPTION**  
The echo utility writes any specified operands, separated by single blank (' ') characters and followed by a newline ('\n') character, to the standard output.

The following option is available:

**-n** Do not print the trailing newline character. This may also be achieved by appending '\c' to the end of the string, as is done by iBCS2 compatible systems. Note that this option as well as the effect of '\c' are implementation-defined in IEEE Std 1003.1-2001 (''POSIX.1'') as amended by Cor. 1-2002. Applications aiming for maximum portability are strongly encouraged to use printf(1) to suppress the newline character.

# What is the “shell”?

- After logging in, Linux starts another program called the shell
- The shell interprets commands the user types and manages their execution
  - The shell communicates with the internal part of the operating system called the kernel
  - The most popular shells are: tcsh, csh, korn, and bash
  - The differences are most times subtle
  - For this tutorial, we are using bash
- Shell commands are **CASE SENSITIVE!**

# Tips To Make Life Easier!

## Speed

- Tab completion: hit Tab to make computer guess your filename.
  - type: ls unix[Tab]
  - result: ls unix\_course\_files
- If nothing happens on the first tab, press tab again...
- Aliases (put in ~/.bashrc file)
  - alias ls='ls -AFG'
  - alias ll='ls -lrhT'

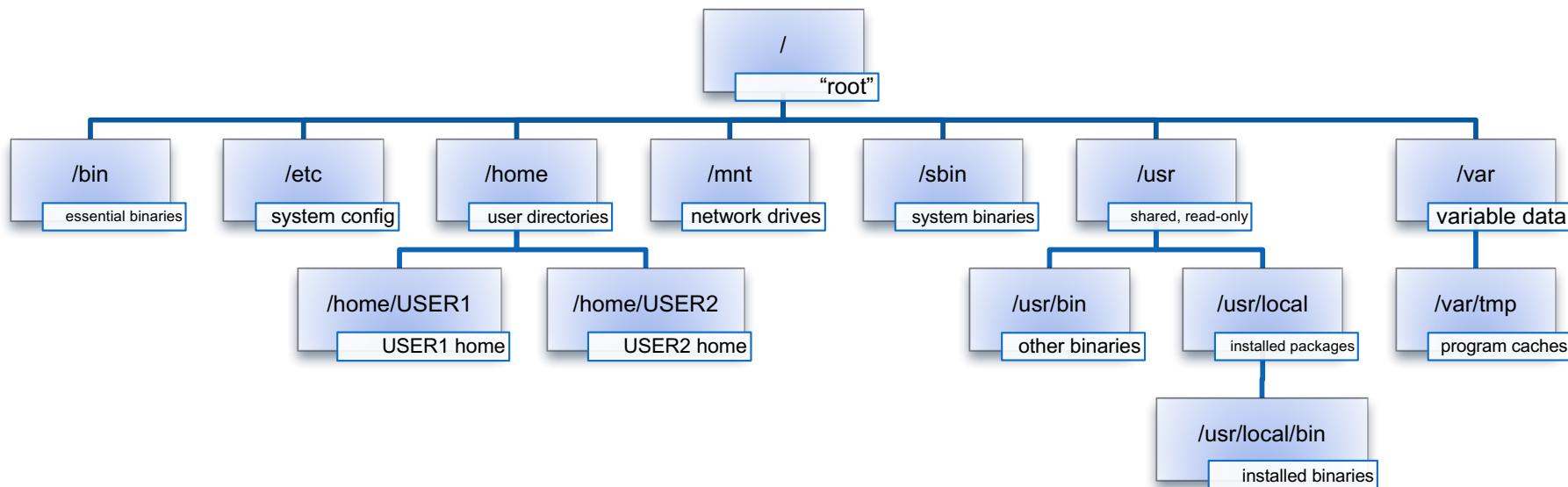
## Movement

- Up Arrow
  - Recalls previous command(s)
- history
  - show every command issued during the session
- !ls
  - repeat the previous "ls" command
- !!
  - repeat the previous command
- Ctrl+a
  - go to beginning of line
- Ctrl+e
  - go to end of line
- Ctrl+c
  - kill current running process in terminal

# Navigating with the Shell

# Navigation: Typical Linux Directory Structure

**pwd** “print working directory”; tells where you are



## Command: pwd

To find your “print working directory” use “pwd”

Note: we are “printing” the information to the screen ☺



A screenshot of a terminal window titled "Desktop — bash — 80x24". The window shows the command "pwd" being run and its output "/Users/squiresrb/Desktop". The window has standard OS X-style window controls (red, yellow, green) and scroll bars.

```
L01891689:Desktop squiresrb$ pwd
/Users/squiresrb/Desktop
L01891689:Desktop squiresrb$
```

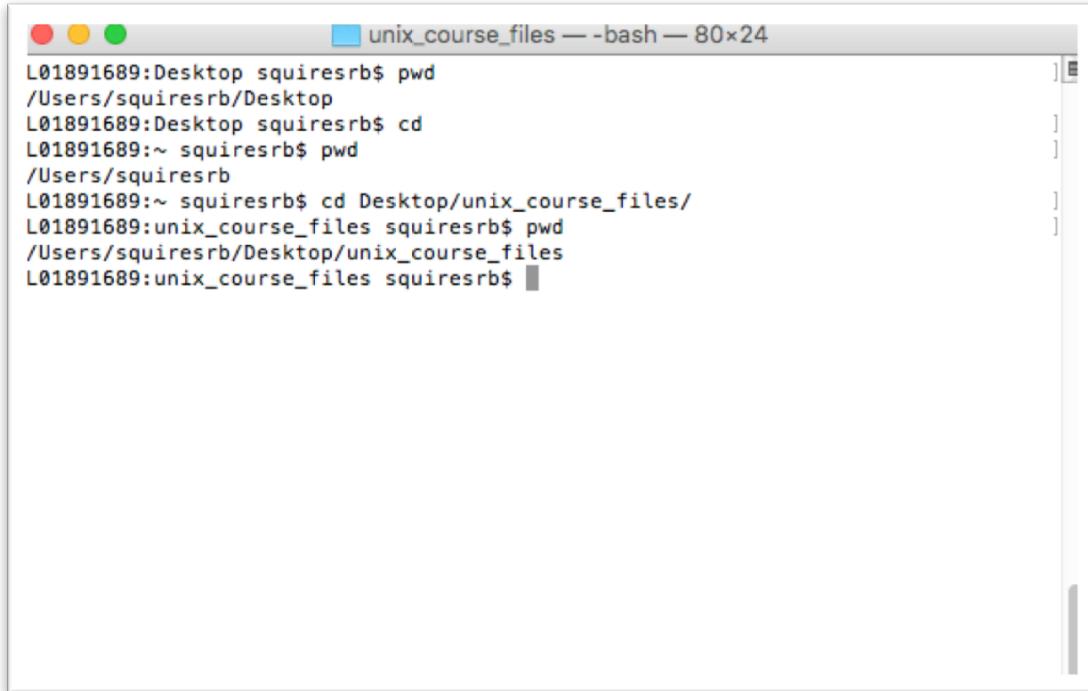
## Command: `cd`

To “change directory” type:  
“`cd`”

To go to home directory:  
“`cd ~`”  
“`cd`”

To go to parent directory:  
“`cd ..`”

Note: “..” is a shortcut to the  
parent directory



```
L01891689:Desktop squiresrb$ pwd
/Users/squiresrb/Desktop
L01891689:Desktop squiresrb$ cd
L01891689:~ squiresrb$ pwd
/Users/squiresrb
L01891689:~ squiresrb$ cd Desktop/unix_course_files/
L01891689:unix_course_files squiresrb$ pwd
/Users/squiresrb/Desktop/unix_course_files
L01891689:unix_course_files squiresrb$
```

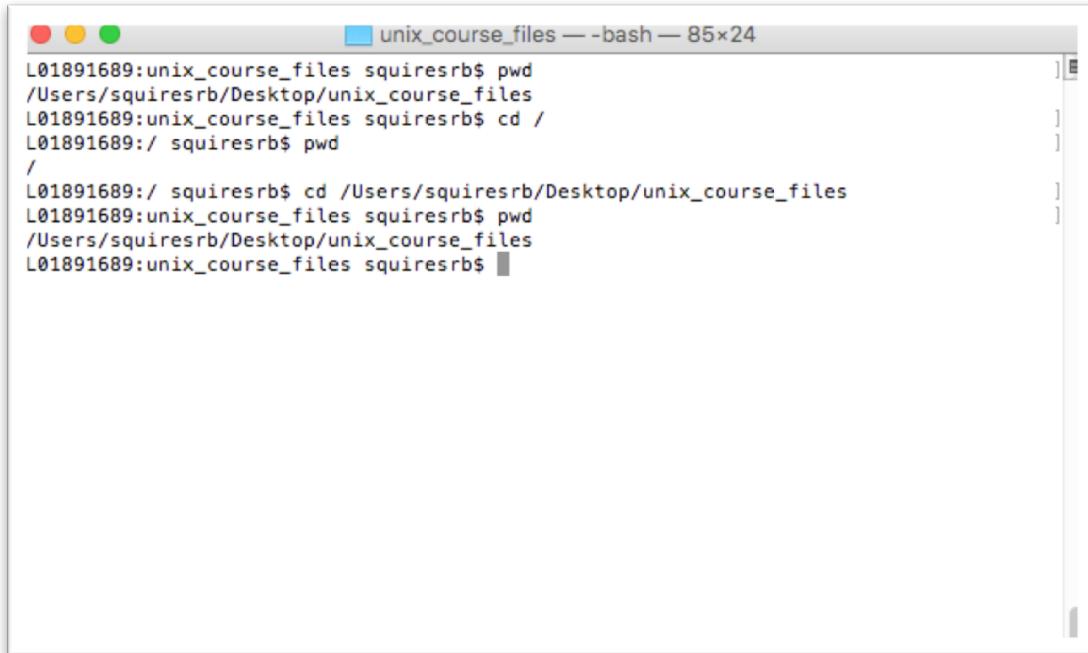
## Paths: Relative vs Absolute

Relative paths:

- “.” current directory
- “..” parent directory
- “./unix\_course\_files”  
directory “output” within  
current directory

To go to specific directory:

- “cd /users/squiresrb/”
- “cd  
/users/squiresrb/Desktop/uni  
x\_course\_files/”



```
L01891689:unix_course_files squiresrb$ pwd
/Users/squiresrb/Desktop/unix_course_files
L01891689:unix_course_files squiresrb$ cd /
L01891689:/ squiresrb$ pwd
/
L01891689:/ squiresrb$ cd /Users/squiresrb/Desktop/unix_course_files
L01891689:unix_course_files squiresrb$ pwd
/Users/squiresrb/Desktop/unix_course_files
L01891689:unix_course_files squiresrb$
```

# Navigation in Summary

- pwd
  - print working directory
  
- cd
  - change directory

# Working with Directories

## Command: ls

To “list” the files in the current directory use

“ls”

ls has many options

- l long list (displays lots of info)
- t sort by modification time
- S sort by size
- h list file sizes in human readable format
- r reverse the order

“man ls” for more options

Options can be combined: “ls -ltr”

```
L01891689:unix_course_files squiresrb$ ls
DNA.txt          redundant.txt      unsorted_numbers.txt
genes.csv        three_lines.txt
origin_of_species.txt  two_lines.txt
L01891689:unix_course_files squiresrb$ ls -l
total 1936
-rwxr-xr-x@ 1 squiresrb  NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb  NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb  NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
-rwxr-xr-x@ 1 squiresrb  NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb  NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb  NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb  NIH\Domain Users    26 Aug 29 05:51 unsorted_numbers.txt
L01891689:unix_course_files squiresrb$ ls -ltr
total 1936
-rw xr-xr-x@ 1 squiresrb  NIH\Domain Users    26 Aug 29 05:51 unsorted_numbers.txt
-rw xr-xr-x@ 1 squiresrb  NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rw xr-xr-x@ 1 squiresrb  NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rw xr-xr-x@ 1 squiresrb  NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rw xr-xr-x@ 1 squiresrb  NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
-rw xr-xr-x@ 1 squiresrb  NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rw xr-xr-x@ 1 squiresrb  NIH\Domain Users    25 Aug 29 05:51 DNA.txt
L01891689:unix_course_files squiresrb$
```

## Listed Columns

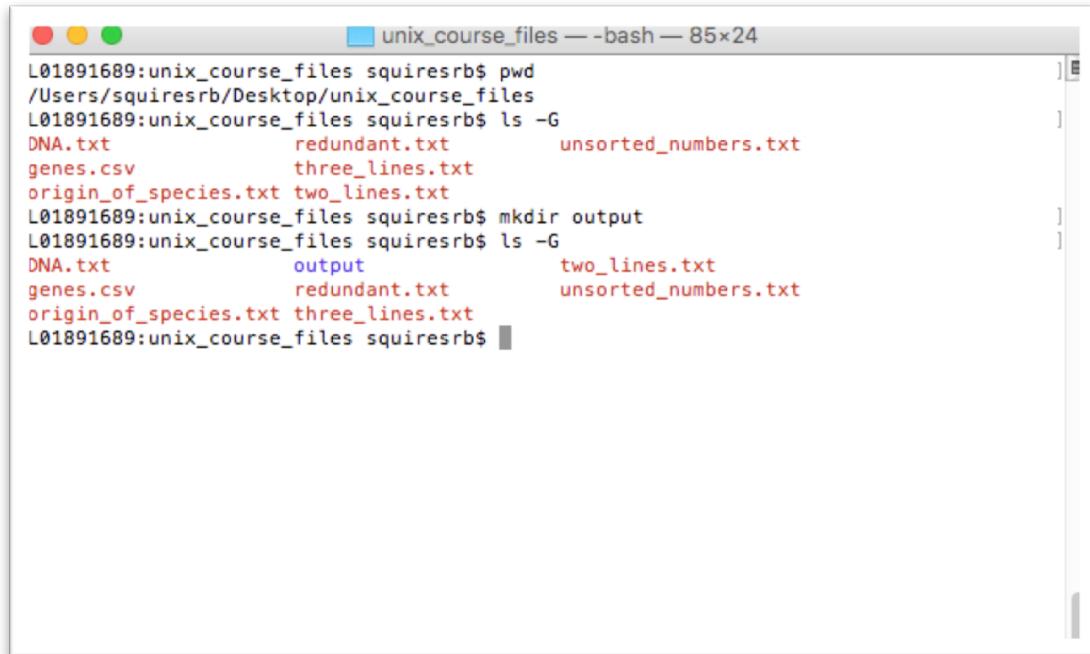
1. File type (d[irectory], - [file], l[ink]) and the permission given on the file
2. Number of memory blocks taken by the file or directory.
3. Owner of the file. This is the Unix user who created this file.
4. Group of the owner. Every Unix user will have an associated group.
5. File size in bytes.
6. Date and the time when this file was created or modified for the last time.
7. File or the directory name.

```
L01891689:unix_course_files squiresrb$ ls -laG
total 1936
drwxr-xr-x@ 10 squiresrb NIH\Domain Users    340 Oct 12 11:57 .
drwx-----+ 39 squiresrb NIH\Domain Users   1326 Oct 12 11:03 ..
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  2 squiresrb NIH\Domain Users     68 Oct 12 11:57 output
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users   26 Aug 29 05:51 unsorted_numbers.txt
L01891689:unix_course_files squiresrb$
```

## Command: **mkdir**

To create a new directory or  
"make directory" use "mkdir"

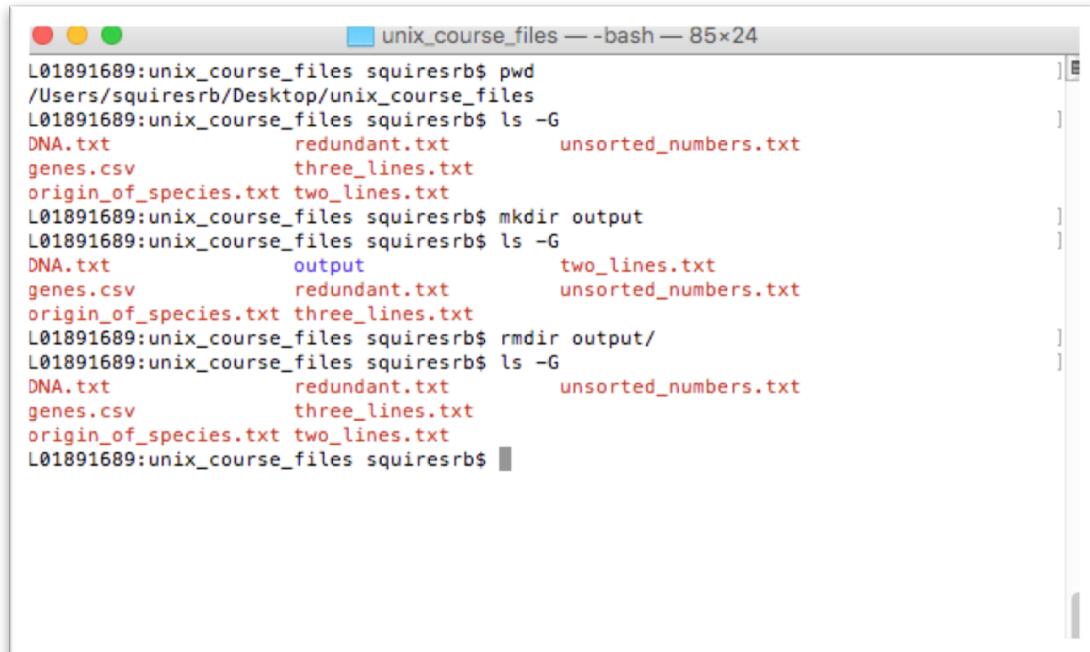
(Notice we use the –G  
option to "ls" to show  
directories in a different  
color (blue))



```
unix_course_files — bash — 85x24
L01891689:unix_course_files squiresrb$ pwd
/Users/squiresrb/Desktop/unix_course_files
L01891689:unix_course_files squiresrb$ ls -G
DNA.txt      redundant.txt      unsorted_numbers.txt
genes.csv    three_lines.txt
origin_of_species.txt two_lines.txt
L01891689:unix_course_files squiresrb$ mkdir output
L01891689:unix_course_files squiresrb$ ls -G
DNA.txt      output      two_lines.txt
genes.csv    redundant.txt      unsorted_numbers.txt
origin_of_species.txt three_lines.txt
L01891689:unix_course_files squiresrb$
```

## Command: **rmdir**

To “remove directory”  
and empty directory use  
“rmdir”

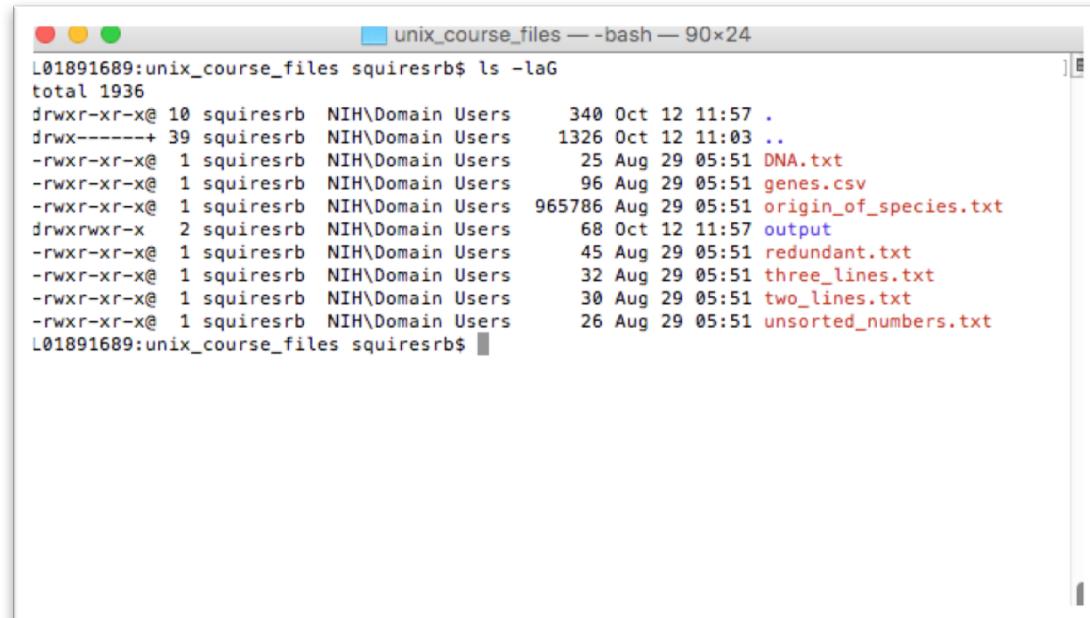


```
L01891689:unix_course_files squiresrb$ pwd
/Users/squiresrb/Desktop/unix_course_files
L01891689:unix_course_files squiresrb$ ls -G
DNA.txt          redundant.txt      unsorted_numbers.txt
genes.csv        three_lines.txt
origin_of_species.txt two_lines.txt
L01891689:unix_course_files squiresrb$ mkdir output
L01891689:unix_course_files squiresrb$ ls -G
DNA.txt          output          two_lines.txt
genes.csv        redundant.txt    unsorted_numbers.txt
origin_of_species.txt three_lines.txt
L01891689:unix_course_files squiresrb$ rmdir output/
L01891689:unix_course_files squiresrb$ ls -G
DNA.txt          redundant.txt      unsorted_numbers.txt
genes.csv        three_lines.txt
origin_of_species.txt two_lines.txt
L01891689:unix_course_files squiresrb$
```

# Working with Files

## Command: ls

We already saw how we can list or “ls” the files in a directory.



The screenshot shows a terminal window titled "unix\_course\_files — bash — 90x24". The command entered is "ls -laG". The output lists 1936 files in a long-form listing format, showing permissions, file size, modification date, and name. The files include "DNA.txt", "genes.csv", "origin\_of\_species.txt", "output", "redundant.txt", "three\_lines.txt", "two\_lines.txt", and "unsorted\_numbers.txt". The terminal window has a standard OS X look with red, yellow, and green close buttons at the top left.

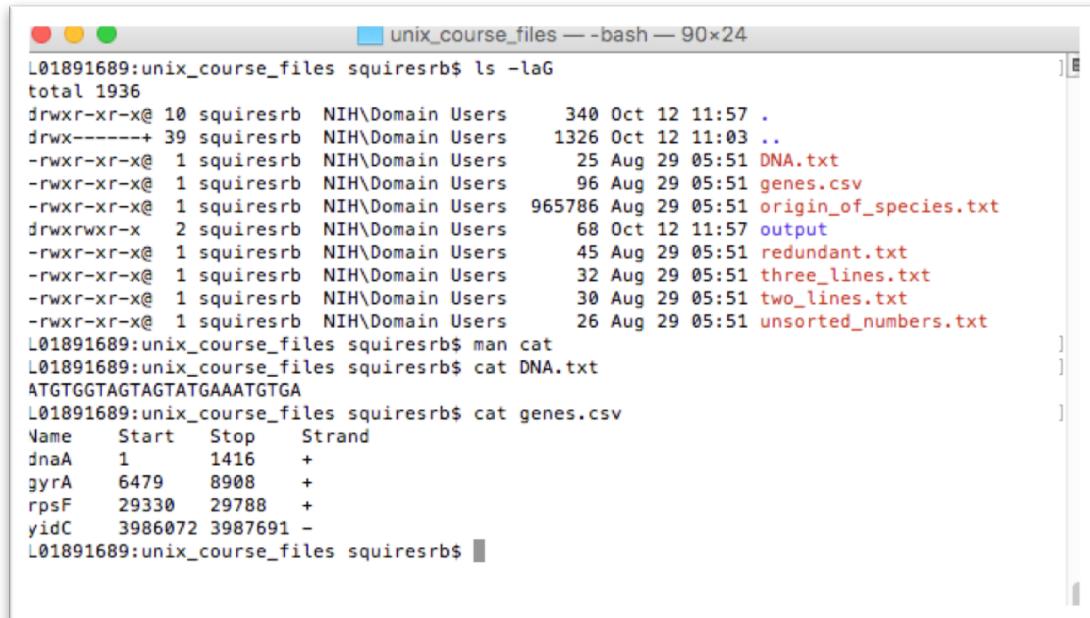
```
L01891689:unix_course_files squiresrb$ ls -laG
total 1936
drwxr-xr-x@ 10 squiresrb NIH\Domain Users    340 Oct 12 11:57 .
drwx-----+ 39 squiresrb NIH\Domain Users   1326 Oct 12 11:03 ..
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  2 squiresrb NIH\Domain Users     68 Oct 12 11:57 output
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rw xr-xr-x@ 1 squiresrb NIH\Domain Users   26 Aug 29 05:51 unsorted_numbers.txt
L01891689:unix_course_files squiresrb$
```

## Command: cat

To view or “concatenate and print” data use cat.

Dumps an entire file to standard output

Good for displaying short, simple files



The screenshot shows a terminal window titled "unix\_course\_files — bash — 90x24". The user has run several commands to demonstrate the use of the "cat" command:

- "ls -laG" lists all files in the current directory with detailed permissions and sizes.
- "cat DNA.txt" displays the contents of the "DNA.txt" file, which contains the DNA sequence ATGTGGTAGTAGTATGAAATGTGA.
- "cat genes.csv" displays the contents of the "genes.csv" file, which contains genomic data for four genes: dnaA, gyrA, rpsF, and yidC, with columns for Name, Start, Stop, and Strand.

## Command: less

Shows “less” then the entire file that “cat” does, allowing forward / backward movement within it

- “return” scrolls forward one line at a time
- “space” advances one page at a time
- “y” scrolls back one line, b one page
- use “/” to search for a string
- Press “q” to quit

```
unix_course_files — bash — 90x24
L01891689:unix_course_files squiresrb$ ls -l
total 1936
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users 25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users 96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users 965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x 2 squiresrb NIH\Domain Users 68 Oct 12 11:57 output
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users 45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users 32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users 30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users 26 Aug 29 05:51 unsorted_numbers.txt
L01891689:unix_course_files squiresrb$ less origin_of_species.txt
L01891689:unix_course_files squiresrb$
```

```
unix_course_files — less origin_of_species.txt — 90x24
E0+FEFF>The Project Gutenberg EBook of On the Origin of Species by Means of Natural Selection, by Charles Darwin

This eBook is for the use of anyone anywhere at no cost and with almost no restrictions whatsoever. You may copy it, give it away or re-use it under the terms of the Project Gutenberg License included with this eBook or online at www.gutenberg.org

Title: On the Origin of Species by Means of Natural Selection or the Preservation of Favoured Races in the Struggle for Life. (2nd edition)

Author: Charles Darwin

Release Date: September 25, 2007 [EBook #22764]

Language: English

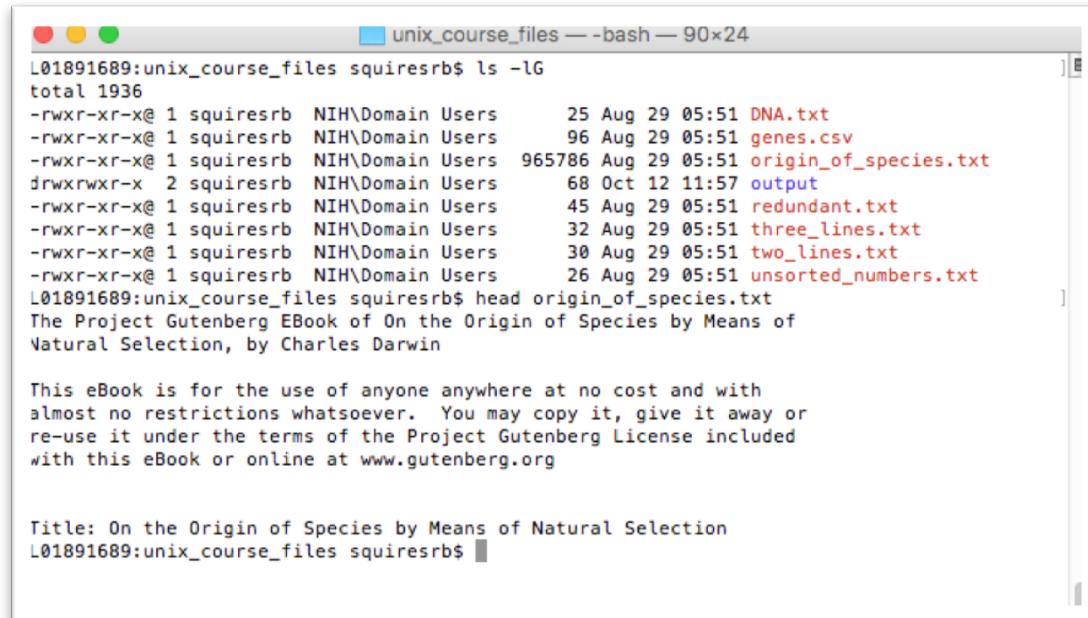
*** START OF THIS PROJECT GUTENBERG EBOOK ORIGIN OF SPECIES ***

origin_of_species.txt
```

## Command: head

“head” displays the top part of a file

- By default it shows the first 10 lines
- “n” option allows you to change the number of lines; “head –n 50 file.txt” displays the first 50 lines of file.txt



```
L01891689:unix_course_files squiresrb$ ls -lg
total 1936
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users      25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users      96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  2 squiresrb NIH\Domain Users      68 Oct 12 11:57 output
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users      45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users      32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users      30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users      26 Aug 29 05:51 unsorted_numbers.txt
L01891689:unix_course_files squiresrb$ head origin_of_species.txt
The Project Gutenberg EBook of On the Origin of Species by Means of
Natural Selection, by Charles Darwin

This eBook is for the use of anyone anywhere at no cost and with
almost no restrictions whatsoever. You may copy it, give it away or
re-use it under the terms of the Project Gutenberg License included
with this eBook or online at www.gutenberg.org

Title: On the Origin of Species by Means of Natural Selection
L01891689:unix_course_files squiresrb$
```

## Command: tail

“tail” displays the top part of a file

- By default it shows the last 10 lines
- “n” option allows you to change the number of lines; “tail –n 50 file.txt” displays the last 50 lines of file.txt

```
L01891689:unix_course_files squiresrb$ ls -lg
total 1936
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users      25 Aug 29 05:51 DNA.txt
-rw-r--r--@ 1 squiresrb NIH\Domain Users      96 Aug 29 05:51 genes.csv
-rw-r--r--@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x@ 2 squiresrb NIH\Domain Users      68 Oct 12 11:57 output
-rw-r--r-x@ 1 squiresrb NIH\Domain Users      45 Aug 29 05:51 redundant.txt
-rw-r--r--@ 1 squiresrb NIH\Domain Users      32 Aug 29 05:51 three_lines.txt
-rw-r--r--@ 1 squiresrb NIH\Domain Users      30 Aug 29 05:51 two_lines.txt
-rw-r--r--@ 1 squiresrb NIH\Domain Users      26 Aug 29 05:51 unsorted_numbers.txt
L01891689:unix_course_files squiresrb$ tail origin_of_species.txt

Most people start at our Web site which has the main PG search facility:
http://www.gutenberg.org

This Web site includes information about Project Gutenberg-tm,
including how to make donations to the Project Gutenberg Literary
Archive Foundation, how to help produce our new eBooks, and how to
subscribe to our email newsletter to hear about new eBooks.
L01891689:unix_course_files squiresrb$
```

## Command: cp

To copy a file use “cp”

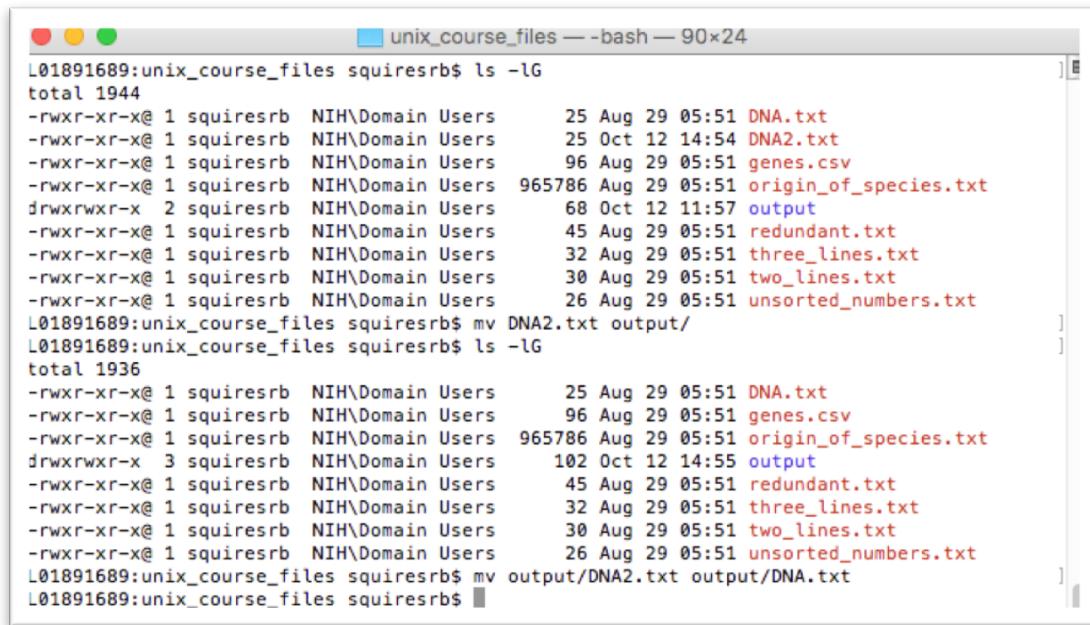
```
unix_course_files — bash — 90x24
L01891689:unix_course_files squiresrb$ ls -lg
total 1936
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  2 squiresrb NIH\Domain Users    68 Oct 12 11:57 output
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    26 Aug 29 05:51 unsorted_numbers.txt
L01891689:unix_course_files squiresrb$ cp DNA.txt DNA2.txt
L01891689:unix_course_files squiresrb$ ls -lg
total 1944
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    25 Oct 12 14:54 DNA2.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  2 squiresrb NIH\Domain Users    68 Oct 12 11:57 output
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    26 Aug 29 05:51 unsorted_numbers.txt
L01891689:unix_course_files squiresrb$
```

## Command: mv

To move a file use “mv”

“mv” can also be used to rename a file:

- mv DNA2.txt DNA3.txt



The screenshot shows a terminal window titled "unix\_course\_files — bash — 90x24". The user has run the command "ls -lg" to list files. The output shows several files including DNA.txt, DNA2.txt, genes.csv, origin\_of\_species.txt, output, redundant.txt, three\_lines.txt, two\_lines.txt, and unsorted\_numbers.txt. The user then runs "mv DNA2.txt output". After this command, the user runs "ls -lg" again, and the output shows DNA.txt, genes.csv, origin\_of\_species.txt, output, redundant.txt, three\_lines.txt, two\_lines.txt, and unsorted\_numbers.txt. DNA2.txt is no longer listed, indicating it has been moved.

```
L01891689:unix_course_files squiresrb$ ls -lg
total 1944
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    25 Oct 12 14:54 DNA2.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  2 squiresrb NIH\Domain Users    68 Oct 12 11:57 output
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    26 Aug 29 05:51 unsorted_numbers.txt
L01891689:unix_course_files squiresrb$ mv DNA2.txt output
L01891689:unix_course_files squiresrb$ ls -lg
total 1936
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  3 squiresrb NIH\Domain Users   102 Oct 12 14:55 output
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    26 Aug 29 05:51 unsorted_numbers.txt
L01891689:unix_course_files squiresrb$ mv output/DNA2.txt output/DNA.txt
L01891689:unix_course_files squiresrb$
```

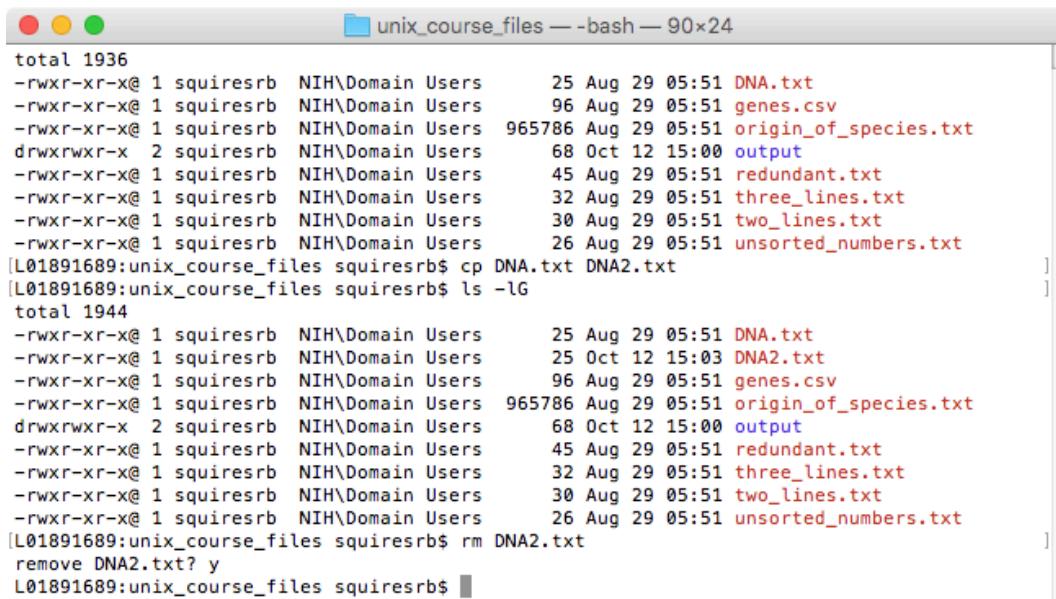
## Command: rm

To remove a file use “rm”

To remove a file  
“recursively”: rm –r

Used to remove all files and  
directories

Be very careful, deletions  
are permanent in Linux



The screenshot shows a terminal window titled "unix\_course\_files — -bash — 90x24". The window displays a list of files in a directory. The user runs the command "cp DNA.txt DNA2.txt" followed by "ls -lg" to show the current state. Then, the user runs "rm DNA2.txt" and is prompted to confirm the deletion with "remove DNA2.txt? y". The terminal window has a standard OS X style with red, yellow, and green close buttons.

```
total 1936
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  2 squiresrb NIH\Domain Users    68 Oct 12 15:00 output
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    26 Aug 29 05:51 unsorted_numbers.txt
[L01891689:unix_course_files squiresrb$ cp DNA.txt DNA2.txt
[L01891689:unix_course_files squiresrb$ ls -lg
total 1944
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    25 Oct 12 15:03 DNA2.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  2 squiresrb NIH\Domain Users    68 Oct 12 15:00 output
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    26 Aug 29 05:51 unsorted_numbers.txt
[L01891689:unix_course_files squiresrb$ rm DNA2.txt
remove DNA2.txt? y
L01891689:unix_course_files squiresrb$ ]
```

# Working with Files: Creating Files in Linux

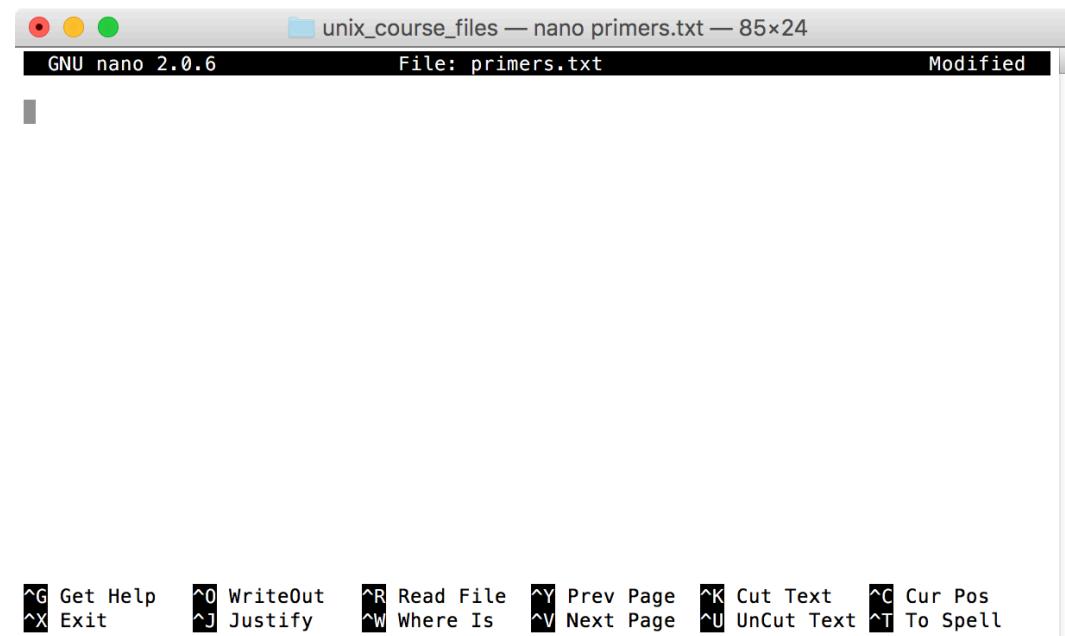
- Requires the use of an Editor
  - Do not use:
  - Microsoft Word
  - WordPad
  - TextEdit (unless you set to plaintext)
- Various Editors:
  - nano / pico
  - vi
  - emacs

## Editing A File Using pico or nano

Type “pico” or “nano” at the prompt

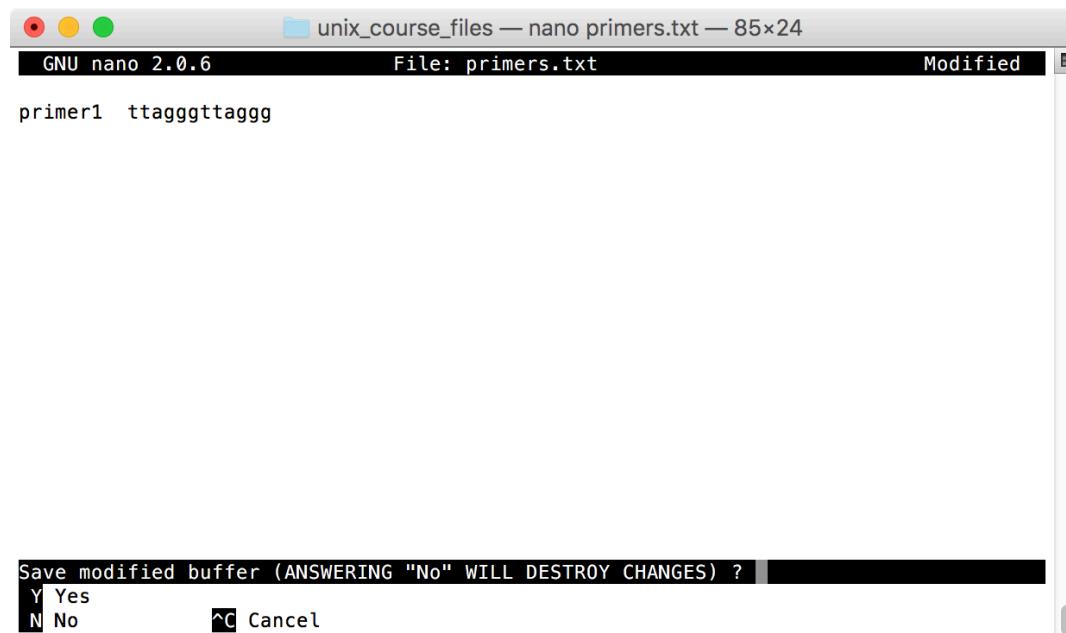
Caret or up arrow is the control key

Control X to exit. Remember to save



# Editing A File Using nano

To save use “ctrl-x”



# File Permissions

- Each file in Linux has an associated permission level
- This allows the user to prevent others from reading/writing/executing their files or directories
- Use “ls -l *filename*” to find the permission level of that file

# Permission levels

- “r” means “read only” permission
- “w” means “write” permission
- “x” means “execute” permission
  - In case of directory, “x” grants permission to list directory contents

# File Permissions

```
L01891689:unix_course_files squiresrb$ ls -lg
```

```
total 1944
```

-rwxr-xr-x@ 1	squiresrb	NIH\Domain Users	25 Aug 29 05:51	DNA.txt		
-rwxr-xr-x@ 1	squiresrb	NIH\Domain Users	25 Oct 12 15:03	DNA2.tx		
-rwxr-xr-x@ 1	squiresrb	NIH\Domain Users	96 Aug 29 05:51	genes.c		
-rwxr-xr-x@ 1	squiresrb	NIH\Domain Users	965786 Aug 29 05:51	origin_		
drwxrwxr-x 2	squiresrb	NIH\Domain Users	68 Oct 12 15:00	output		
-rwxr-xr-x@ 1	squiresrb	NIH\Domain Users	45 Aug 29 05:51	redunda		
-rwxr-xr-x@ 1	squiresrb	NIH\Domain Users	32 Aug 29 05:51	three_l		
-rwxr-xr-x@ 1	squiresrb	NIH\Domain Users	30 Aug 29 05:51	two_lin		
-rwxr-xr-x@ 1	squiresrb	NIH\Domain Users	26 Aug 29 05:51	unsorte		

User (you)

Group

Everyone

## Command: chmod

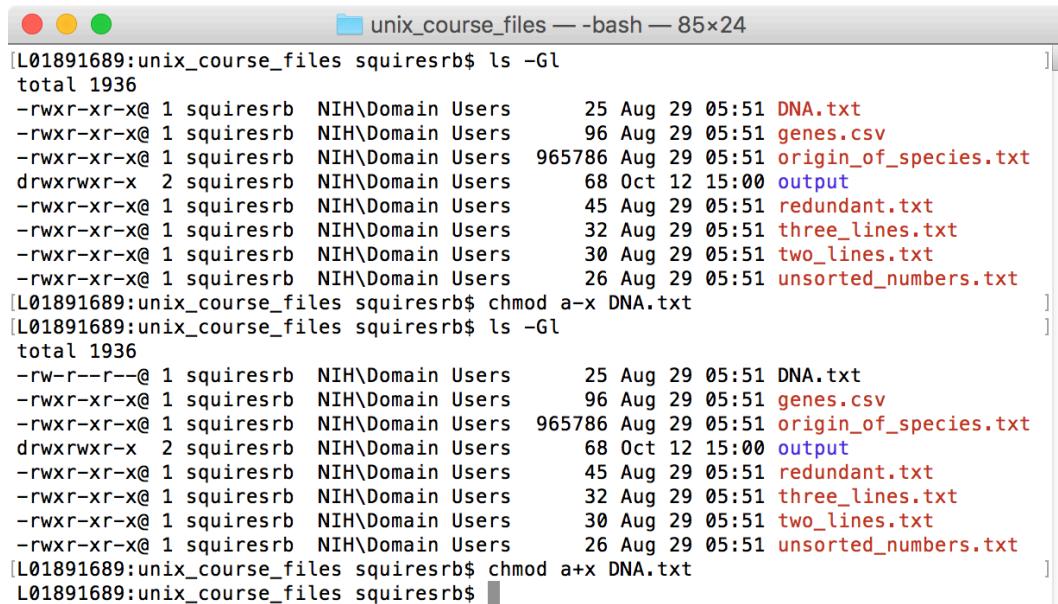
“change mode”

If you own the file, you can change it's permissions with “chmod”

Syntax: chmod  
[user/group/others/all]+[permission] [file(s)]

To grant execute permission to all type:

chmod a+x DNA.txt



The screenshot shows a terminal window titled "unix\_course\_files — bash — 85x24". It displays a list of files with their permissions, last modified date, and names. The user "squiresrb" runs three commands: "ls -Gl" to show the initial state, "chmod a-x DNA.txt" to change the permissions of "DNA.txt" to "a-x", and another "ls -Gl" to show the updated permissions. The permissions for "DNA.txt" are changed from "rwxr-xr-x" to "rw-r--r--".

```
[L01891689:unix_course_files squiresrb$ ls -Gl
total 1936
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  2 squiresrb NIH\Domain Users    68 Oct 12 15:00 output
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    26 Aug 29 05:51 unsorted_numbers.txt
[L01891689:unix_course_files squiresrb$ chmod a-x DNA.txt
[L01891689:unix_course_files squiresrb$ ls -Gl
total 1936
-rw-r--r--@ 1 squiresrb NIH\Domain Users    25 Aug 29 05:51 DNA.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    96 Aug 29 05:51 genes.csv
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users  965786 Aug 29 05:51 origin_of_species.txt
drwxrwxr-x  2 squiresrb NIH\Domain Users    68 Oct 12 15:00 output
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    45 Aug 29 05:51 redundant.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    32 Aug 29 05:51 three_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    30 Aug 29 05:51 two_lines.txt
-rwxr-xr-x@ 1 squiresrb NIH\Domain Users    26 Aug 29 05:51 unsorted_numbers.txt
[L01891689:unix_course_files squiresrb$ chmod a+x DNA.txt
L01891689:unix_course_files squiresrb$ ]
```

# Useful Unix Programs

## Commands: **sort**

**sort genes.csv**

alphabetize “genes.csv”

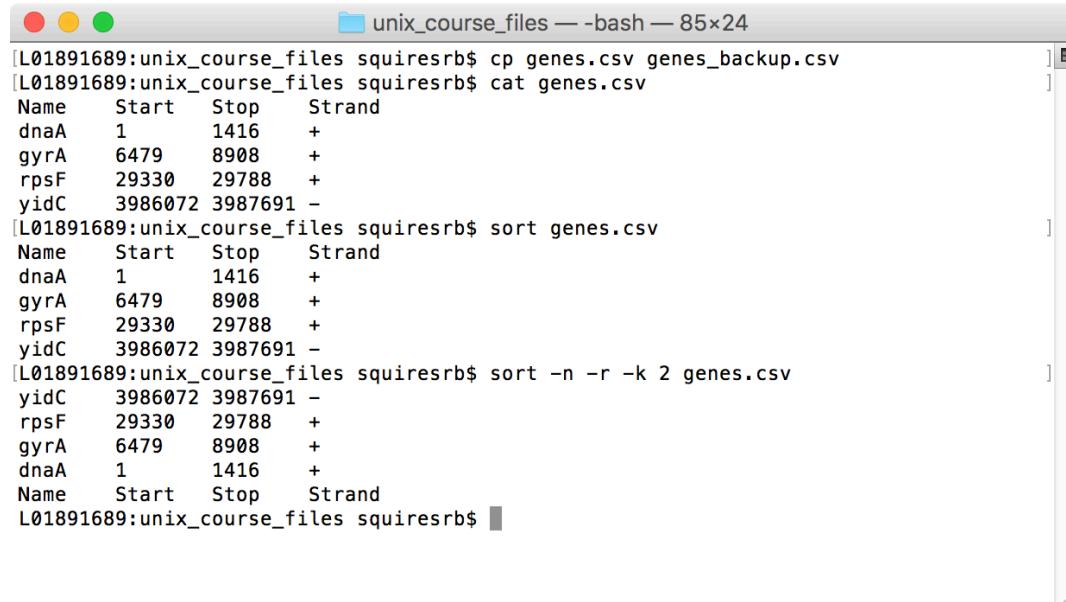
**sort -n genes.csv**

sort “file1” by number

**sort -n -r -k 2**

**genes.csv**

sort “file1” by the second  
word or column in reverse  
numerical order



The screenshot shows a terminal window titled "unix\_course\_files — bash — 85x24". The user has copied a file named "genes.csv" to "genes\_backup.csv" and then used the "cat" command to view its contents. The data is as follows:

Name	Start	Stop	Strand
dnaA	1	1416	+
gyrA	6479	8908	+
rpsF	29330	29788	+
yidC	3986072	3987691	-

Next, the user runs the "sort" command on "genes.csv". The output is identical to the input, showing the same data rows. Finally, the user runs "sort -n -r -k 2 genes.csv", which sorts the data by the second column (Stop) in reverse numerical order. The resulting output is:

Name	Start	Stop	Strand
yidC	3986072	3987691	-
rpsF	29330	29788	+
gyrA	6479	8908	+
dnaA	1	1416	+

# Command: diff

- To compare two files for differences use “diff”
  - Try: diff /dev/null hello.txt
  - /dev/null is a special address -- it is always empty, and anything moved there is deleted

# Basic Linux Commands: grep (search within files)

- ***g/re/p*** (*globally search a regular expression and print*)
- **grep key file\*** report the file name and line where “key” appears in file\*
- **grep -v key file\*** report the file names of files that do not match “key”
- **man grep** see other functions of grep. (lots! regular expressions!)

# Basic Linux Commands: File Compression

`gzip temp/*`

compress every file in “temp”; adds .gz extension

`gunzip temp/*.gz`

expand every “gzipped” file in “temp”

`tar -zcvf myfiles.tar.gz temp/*` create a single archive of every file in “temp”

`tar -xvf test_data.tar.gz`

copy every file out of the archive

“tarball” ≠



# Basic Linux Commands: Pipelining

`ls | wc`

count the number of files in a directory

`grep | sort > file1`

pull out searched-for lines, sort them, and write a new file

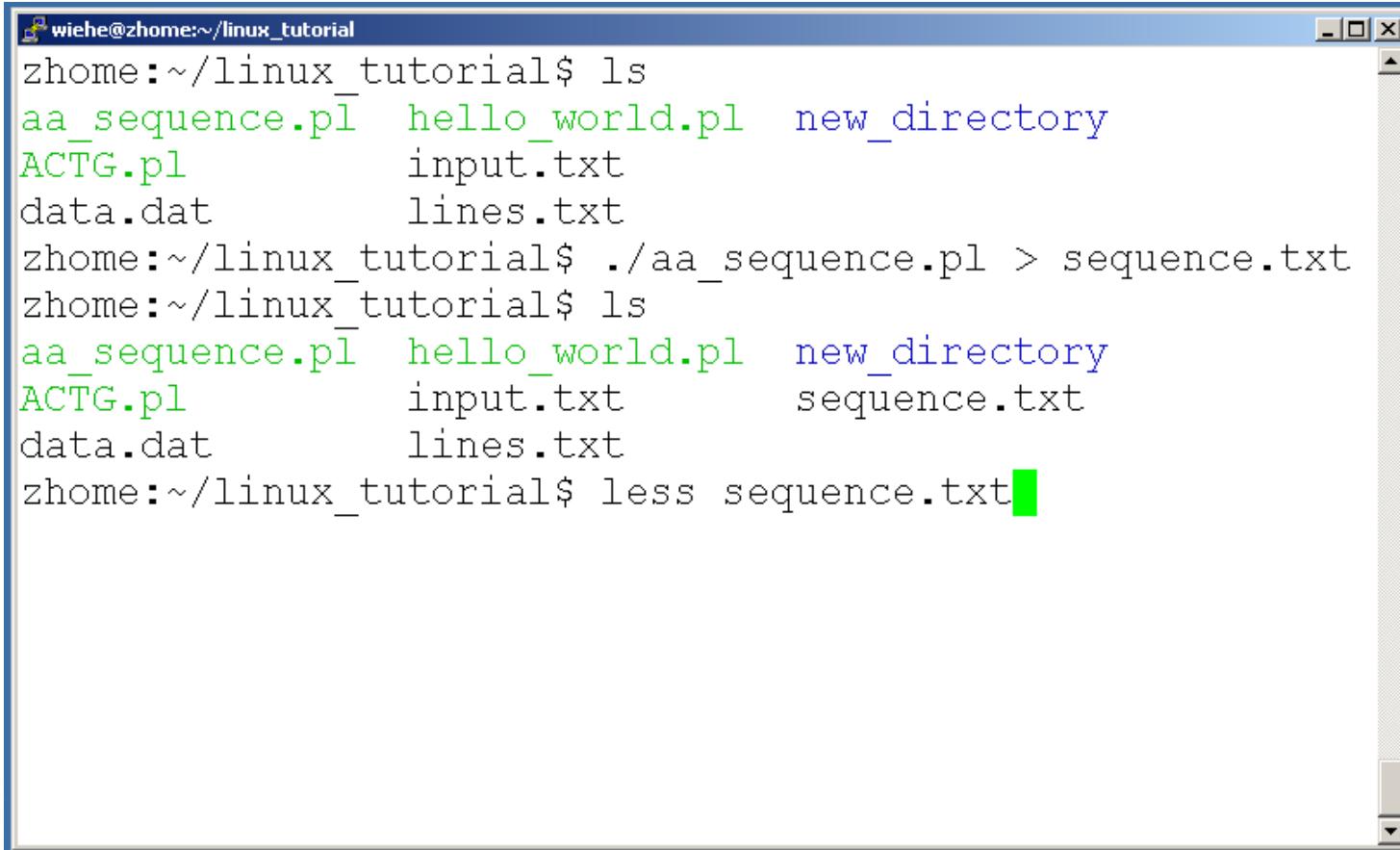
## Exercise 2:

`head -n 2000 lymph1k.fastq | gzip > head2K.txt.gz`

# Input/Output Redirection (“piping”)

- Programs can output to other programs
- Called “piping”
- “program\_a | program\_b”
  - program\_a’s output becomes program\_b’s input
- “program\_a > file.txt”
  - program\_a’s output is written to a file called “file.txt”
- “program\_a < input.txt”
  - program\_a gets its input from a file called “input.txt”

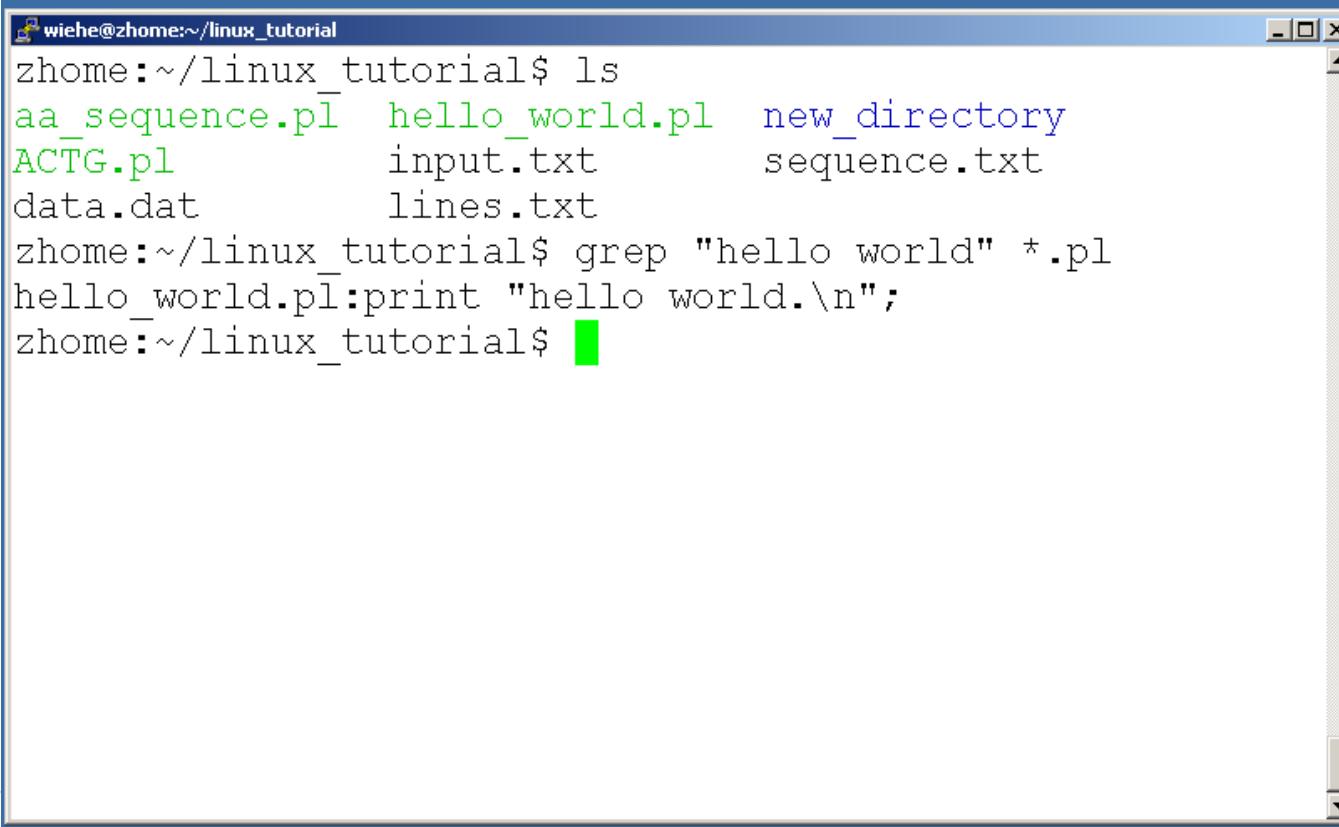
# Examples Of Piping



wiehe@zhome:~/linux\_tutorial\$ ls  
zhome:~/linux\_tutorial\$ aa\_sequence.pl hello\_world.pl new\_directory  
ACTG.pl input.txt  
data.dat lines.txt  
zhome:~/linux\_tutorial\$ ./aa\_sequence.pl > sequence.txt  
zhome:~/linux\_tutorial\$ ls  
aa\_sequence.pl hello\_world.pl new\_directory  
ACTG.pl input.txt sequence.txt  
data.dat lines.txt  
zhome:~/linux\_tutorial\$ less sequence.txt

# Command: grep

- Search files in a directory for a specific string use “grep”



wiehe@zhome:~/linux\_tutorial  
zhome:~/linux\_tutorial\$ ls  
aa\_sequence.pl hello\_world.pl new\_directory  
ACTG.pl input.txt sequence.txt  
data.dat lines.txt  
zhome:~/linux\_tutorial\$ grep "hello world" \*.pl  
hello\_world.pl:print "hello world.\n";  
zhome:~/linux\_tutorial\$ █

# Running A Program (a.k.a. a job)

- Make sure the program has executable permissions
- Use “./” to run the program

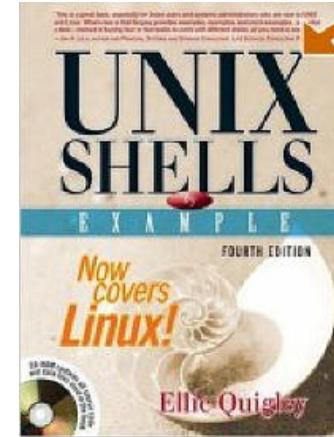
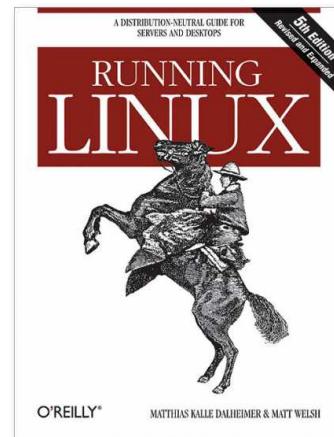
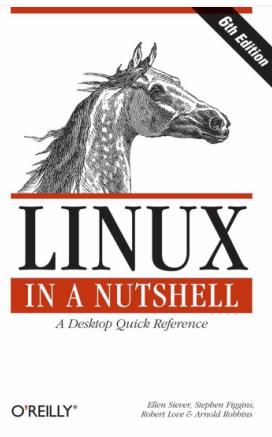
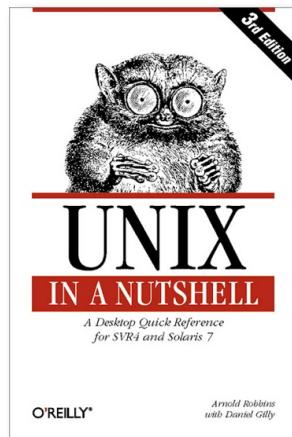
# Commands: ssh (“secure shell”), scp (“secure copy”)

- ssh is used to securely log in to remote systems, successor to telnet
  - ssh [username]@[hostname]
  - Try:  
**ssh yourusername@localhost**  
Type “exit” to log out of session
- 
- Scp is used to copy files to/from remote systems, syntax is similar to cp:
    - scp [local path] [username]@[hostname]:[remote file path]
  - Try:
    - **scp hello.txt yourusername@localhost:scp-test.txt**

# Linux Web Resources

- <https://www.tutorialspoint.com/unix/>
- <http://www.ee.surrey.ac.uk/Teaching/Linux/>
- <http://www.ugu.com/sui/ugu/show?help.beginners>
- <http://en.wikipedia.org/wiki/Linux>

# Recommended Reading



## Linux in a Nutshell, Sixth Edition

Ellen Siever, Stephen Figgins,  
Robert Love, Arnold Robbins

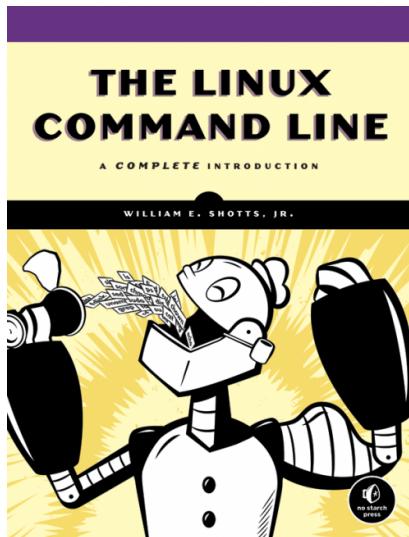
## Running Linux, 5th Edition

Matthias Kalle  
Dalheimer, Matt  
Welsh

## Linux® Shells by Example, Fourth Edition

Ellie Quigley

# Recommended Reading



The Linux Command Line  
By William Shotts  
Download free from [here](#).

# Objectives

- Explain one or more advantages to using the command line for research
- Explain one or more advantages to using the command line for programming
- Be able to get to the command line on your computer
- Understand how to navigate a linux operating system
- Know one or more command to examine and manipulate large files (1 gigbit+)

# Questions?