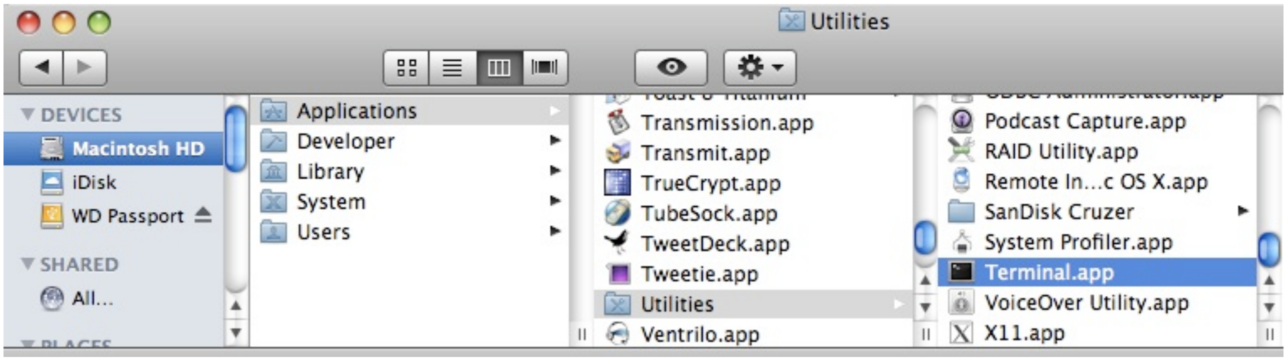
Introduction to Terminal

Learning Objectives

Obtain a basic understanding of what the Terminal is and how to run basic network commands.

Open Terminal

Click on the Finder icon in your Mac OS X Dock to open a new finder window
Select your hard drive on the left hand side of your newly opened Finder window.
Navigate to /Applications/Utilities and double-click on Terminal to open.



What is Terminal?

Terminal, is a Mac OS X application that allows you to execute text-based commands on your system. Terminal is a front end to the shell or command-line environment of your Operating System. There are many different Terminal like applications on a number of different operating systems and platforms. Terminal is simply the default shell application that comes with your Mac.

There are several different common keyboard shortcuts that allow you to achieve different actions within Terminal. A brief list of them are as follows.

^ = Ctrl
⎋ = Esc
⌘ = Command (Apple Key)
⇧ = Shift

Command	Result
^ A	Move the cursor to the beginning of the command line
^ B	Move the cursor back one character.
^ C	Break out of the command without any change to the settings.
^ D	Delete the character at the cursor.
^ E	Move the cursor to the end of the command line.
^ F	Move the cursor forward one character.
^ I	Recall a complete command name; same as the Tab key operation.
^ K	Delete all characters from the cursor to the end of the command line.
^ L	Redisplay the current command line; same as the ^ R key.
^ N	Recall the most recent command in the command history relative to the current pointer in the history list.
^ P	Recall the oldest command in the command history, beginning with the most recent command.
^ R	Redisplay the current command line; same as the ^ L key.
^ T	Transpose the characters to the left of the cursor with the character located at the cursor.
^ U	Delete all characters from the cursor to the beginning of the command line.
^ V	Insert a code to indicate that the immediately following value is a command entry.
^ W	Delete the word to the left of the cursor.
^ X	Delete all characters from the cursor to the beginning of the command line.
^ Y	Recall the most recent entry in the buffer (which contains the last ten items you deleted.)
⎋ B	Move the cursor back one word.
⎋ C	Capitalize the word at the cursor.
⎋ D	Delete from the cursor to the end of the word.
⎋ F	Move the cursor forward one word.
⎋ L	Change the word at the cursor to lowercase.

Command	Result
⎋ Q	Insert a code to indicate that the immediately following value is a command entry.
⎋ U	Capitalize letters from the cursor to the end of the word.
⎋ Y	Recall the next deleted buffer. (Effective after using ^ Y.)

There are also a set of Mac OS X only keyboard shortcuts that are as follows.

Command	Result
⌘ N	Create a new shell window.
⌘ T	Create a new shell tab.
⌘ `	Next Terminal window.
⌘⇧ ~	Previous Terminal window.
⌘⇧ }	Next Terminal tab.
⌘⇧ {	Previous Terminal tab.
⌘ W	Close window or tab.

Executing the PING Command

When you first open Terminal, you will see a command prompt window. First you will ping google.com by sending a number of packets, or bits of information back and forth between your computer and Google's website. This command is often used to see if a host, or website, can be found and communicated with. To ping google.com, type the following lines and hit the Return key on your keyboard.

```
ping -c 10 google.com
```

Upon execution of the previous command, you should an output similar to the following in the Terminal.

```
PING google.com (64.233.167.99): 56 data bytes
64 bytes from 64.233.167.99: icmp_seq=0 ttl=241 time=54.080 ms
64 bytes from 64.233.167.99: icmp_seq=1 ttl=241 time=42.806 ms
64 bytes from 64.233.167.99: icmp_seq=2 ttl=241 time=44.326 ms
64 bytes from 64.233.167.99: icmp_seq=3 ttl=241 time=46.532 ms
```

```
64 bytes from 64.233.167.99: icmp_seq=4 ttl=241 time=46.182 ms
64 bytes from 64.233.167.99: icmp_seq=5 ttl=241 time=42.599 ms
64 bytes from 64.233.167.99: icmp_seq=6 ttl=241 time=45.129 ms
64 bytes from 64.233.167.99: icmp_seq=7 ttl=241 time=41.089 ms
64 bytes from 64.233.167.99: icmp_seq=8 ttl=241 time=36.847 ms
64 bytes from 64.233.167.99: icmp_seq=9 ttl=241 time=43.601 ms

--- google.com ping statistics ---
10 packets transmitted, 10 packets received, 0% packet loss
round-trip min/avg/max/stddev = 36.847/44.319/54.080/4.199 ms
```

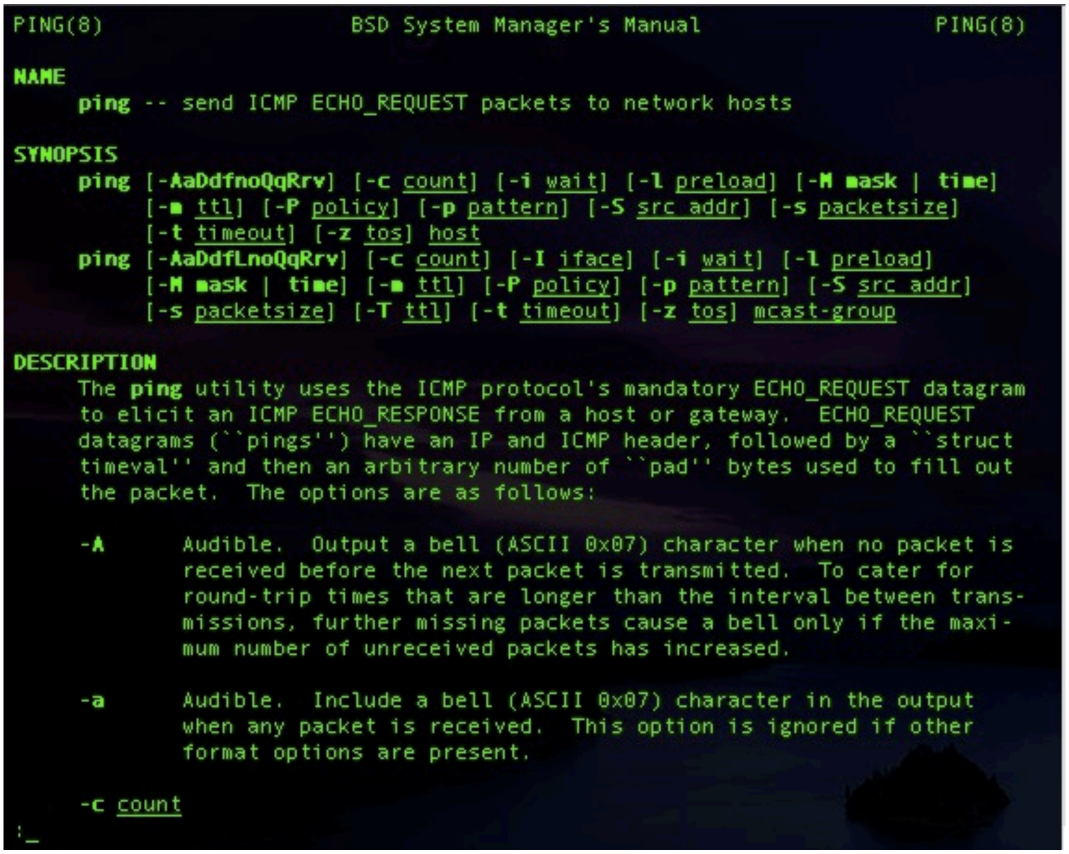
So, what does all that output mean? Essentially, it is telling you that it is first attempting to ping google.com, which resolved to the IP address 64.233.167.99. Then, a packet that contained 54 bytes of data was sent to the host, was returned 10 times, and it took X amount of ms for that information to be returned You are then given an array of statistics for the entire process.

The Manual

Now that you have executed a simple ping command, you should now look at the manual page for the ping command as well. You do that by executing the following command, the same way you did previously.

```
man ping
```

Upon execution of the previous command, you should see output that is similar to the following. The man command allows you to read the manual page for any given Terminal command.



```
PING(8)                BSD System Manager's Manual                PING(8)

NAME
     ping -- send ICMP ECHO_REQUEST packets to network hosts

SYNOPSIS
     ping [-AaDdfnoQqRrv] [-c count] [-i wait] [-l preload] [-M mask | time]
          [-m ttl] [-P policy] [-p pattern] [-S src_addr] [-s packetsize]
          [-t timeout] [-z tos] host
     ping [-AaDdfLnoQqRrv] [-c count] [-I iface] [-i wait] [-l preload]
          [-M mask | time] [-m ttl] [-P policy] [-p pattern] [-S src_addr]
          [-s packetsize] [-T ttl] [-t timeout] [-z tos] mcast-group

DESCRIPTION
     The ping utility uses the ICMP protocol's mandatory ECHO_REQUEST datagram
     to elicit an ICMP ECHO_RESPONSE from a host or gateway.  ECHO_REQUEST
     datagrams (``pings'') have an IP and ICMP header, followed by a ``struct
     timeval'' and then an arbitrary number of ``pad'' bytes used to fill out
     the packet.  The options are as follows:

     -A      Audible.  Output a bell (ASCII 0x07) character when no packet is
             received before the next packet is transmitted.  To cater for
             round-trip times that are longer than the interval between trans-
             missions, further missing packets cause a bell only if the maxi-
             mum number of unreceived packets has increased.

     -a      Audible.  Include a bell (ASCII 0x07) character in the output
             when any packet is received.  This option is ignored if other
             format options are present.

     -c count
:_
```

If you take a look at the SYNOPSIS section, you will see that the -c option you used when executing the ping command is setting the count or number of packets to request. An option is a value that you can set that changes the default action of the command. You can also read more about each option/command by hitting the spacebar on your keyboard repeatedly, which will effectively scroll the manual page; hit the Q key on your keyboard to close the manual page.

The value that you used previously for the -c option was 10. Try again, but this time specify that you only want it to request 5 packets. Upon execution of that command, you should now see a similar output as before, but this time there will only be 5 lines of data, instead of 10.

Expanding Your Knowledge

Now that you have a basic understanding of how the ping command works, take the time to play around with various options and observe the output that is received.

Basic Terminal Commands

Learning Objectives

Obtain a thorough understanding of basic Terminal commands.

Open Terminal

Open the Terminal application as outlined in the previous Introduction to Terminal tutorial.

Where are You?

When Terminal is first launched, the initial working directory is generally the given user's home directory. To the left of your user name is a ~, which indicates your home directory. You can tell the exact location of your working directory by executing the following command.

```
pwd
```

The command pwd stands for *Print Working Directory*. Upon execution of the previous command, the output in your terminal window should look similar to this.

```
/Users/home/yourname
```

This command can be extremely useful; so learn it, know it, use it.

What's in Here?

Another basic tool that you need to be aware of is how to display a list of contents in a given directory. To list the contents of your user folder, execute the following command.

```
ls
```

A list similar to the following should be presented to you.

```
Applications    Documents        Library      Music       Public
Desktop         Downloads        Movies       Pictures
```

You can also retrieve even more information by adding adding options to the ls command. To list all directories/files and all of their properties, run the following command.

```
ls -l
```

There are several options available to you with the ls command. You can use the same command as you used in Introduction to Terminal tutorial to learn more about the additional options.

```
man ls
```

Take the time to experiment with the various options.

Creating Directories

Often times when working in a server environment, you do not have access to a GUI. In instances like this, creating a directory may seem like a daunting task, but it most certainly is not. It may even be easier to do it via the command-line than in a graphical environment. Execute the following command to create a new directory called test.

```
mkdir test
```

Note that the mkdir command used the supplied argument (information that is passed to the command) of test for the directory name. To assure that the directory was in fact created, list the contents of the directory as you did previously. To learn more about the mkdir command, read it's manual page.

Moving Directories or Files

Instead of having a directory called test, you can change the directory name to nss by using the mv command, which will move a file or directory from one location to another. It also has the added ability of modifying file and directory names. To change the directory name to nss, us the following command.

```
mv test nss
```

The first part of the mv command is the input and the second part is the output. You may also use this to move files and directories to another location, by providing the path to the file or directory.

Changing Directories

Now that you have created your nss directory, how do you start working inside of it? To do so, you execute the following command.

```
cd nss
```

You should now be inside the nss folder that you created. To assure that you are, you can use the pwd command.

Creating Files

There are a number of ways to create a file in Terminal environment, but being as though most files you will need to create are text-based files, you will learn how to create a file using a basic unix text editor, nano. You'll now create a new text file that stores a bit of information about yourself. To open a new file for editing, execute the following command, replacing yourname with your first name.

```
nano yourname.txt
```

The basic text editor nano will appear in place of the command line. Type your full name, followed by your favorite food on a separate line and use the ^ X keyboard combination to exit. You will prompted as to whether or not you would like to save the changes, to accept press the Y key on your keyboard and then the Return key to save as the file name you specified earlier.

Clearing the Terminal Window

At this point, your terminal window may be filled up and be a bit overwhelming. To give yourself a clean slate to work, execute the following command, which will simply clear your Terminal.

```
clear
```

Copying Files

Now that you have a text file that contains some information about yourself, you'll learn to copy that file and modify it. To do so, execute the following command, changing yourname to your first name.

```
cp yourname.txt info.txt
```

This will copy your text file to a new text file called info.txt. You will now edit this file using the technique you used to create a new file, by executing the following command.

```
nano info.txt
```

Modify this text file by adding your favorite 17th century philosopher … or favorite color and save as previously instructed.

Removing Files

You should no longer need the original file that contains information on yourself. To remove this file, execute the following command, again replacing yourname with your first name.

```
rm yourname.txt
```

If you were to list the contents of the directory, you should only see info.txt.

Printing File Contents

To print the contents of the info.txt file to the screen, execute the following command.

```
cat info.txt
```

You should see the contents of the text file. The cat command makes viewing the contents of a file far quicker than editing it with a text-editor such as nano.

Expanding Your Knowledge

There are many modifications you can make to each of these commands using various options. Use the knowledge that you have obtained in this tutorial to view the manual for the rm command, traverse up a directory to your home directory and remove the entire nss directory.

Apply What You’ve Learned

Now that you understand how to use these commands and how to navigate the file system we’re going to put you to the test. **Using only Terminal**, create the exact same file structure that I have outlined below. You will need to create files and directories and navigate through the directory structure. Tip - **Start by creating the “earth” directory on your desktop. When creating files you can use nano to make a dummy file like we did earlier, then copy it with different names.**

- **~/Desktop**
    - **earth**
        - **asia**
            - beijing.bak
            - hong_kong
            - seoul.txt
            - tokyo.jpg
            - ulaanbaatar.html
        - **europe**
            - barcelona
            - london.jpg
            - marseille.txt
            - oslo.html
            - rome.bak
        - **north_america**
            - new_york
            - toronto.jpg
            - vancouver.txt
            - orlando.html
            - sacramento.bak
        - **south_america**
            - buenos aires.txt
            - patagonia.bak
            - rio.jpg
            - san_salvador
            - montevideo.html

*Notice that just because you’ve named something with a standard file extension such as .jpg or .html, doesn’t mean that you’ve made that kind of file. Try to view london.jpg in the GUI and you’ll see an error message stating that it cannot be opened. That’s because it’s not really a .jpg file. This is the danger with (but also the great thing about) Terminal. It assumes you know what you’re doing.

Once you have finished it I want you to perform a long-format recursive list in Terminal (see manual). Now that that's done I want you to change it (still using Terminal) to match the file structure that I've provided below. You will need to ***move*** and ***rename*** files and create directories.

- **~/Desktop**
  - **earth**
    - **asia**
      - **china**
        - peiking.bak
        - hong_kong
      - **japan**
        - kyoto.jpg
      - **korea**
        - incheon.txt
      - **mongolia**
        - hovd.html
    - **europe**
      - **france**
        - paris.txt
      - **italy**
        - venice.bak
      - **norway**
        - bergen.html
      - **spain**
        - barcelona
      - **uk**
        - manchester.jpg
    - **north_america**
      - **us**
        - new_york
        - orlando.mp3
        - cleveland.bak
      - **canada**
        - quebec.jpg
        - halifax.txt
    - **south_america**
      - **argentina**
        - cordoba.txt
        - san_juan.bak
      - **brazil**
        - sao_paulo.jpg
      - **el_salvador**
        - san_salvador
      - **uruguay**
        - maldonado.html

Once you are done, perform a long-format recursive list and export the contents of your Terminal output to a .txt file. To do this click on Shell > Export Text As…. in the menu bar. Make sure you include all your work. I want to see everything. Use the standard naming convention for the class and use "terminal" for the assignment name. i.e. lastname_first name_terminal.txt. Upload the file to FSO.