Looking Around

Now that we know how to move from working directory to working directory, we're going to take a tour of our Linux system and, along the way, learn some things about what makes it tick. But before we begin, we have to learn about some tools that will come in handy during our journey. These are:

- `ls` (list files and directories)
- `less` (view text files)
- `file` (classify a file's contents)

ls

The **ls** command is used to list the contents of a directory. It is probably the most commonly used Linux command. It can be used in a number of different ways. Here are some examples:

Command	Result
`ls`	List the files in the working directory
`ls /bin`	List the files in the `/bin` directory (or any other directory we care to specify)
`ls -l`	List the files in the working directory in long format
`ls -l /etc /bin`	List the files in the `/bin` directory and the `/etc` directory in long format
`ls -la ..`	List all files (even ones with names beginning with a period character, which are normally hidden) in the parent of the working directory in long format

Examples of the ls command

These examples also point out an important concept about commands. Most commands operate like this:

```
command -options arguments
```

where *command* is the name of the command, *-options* is one or more adjustments to the command's behavior, and *arguments* is one or more "things" upon which the command operates.

In the case of **ls**, we see that **ls** is the name of the command, and that it can have one or more options, such as **-a** and **-l**, and it can operate on one or more files or directories.

A Closer Look at Long Format

If we use the **-l** option with **ls**, you will get a file listing that contains a wealth of information about the files being listed. Here's an example:

```
-rw-------     1 me       me               576 Apr 17  2019 weather.txt
drwxr-xr-x     6 me       me              1024 Oct  9  2019 web_page
-rw-rw-r--     1 me       me            276480 Feb 11 20:41 web_site.tar
-rw-------     1 me       me              5743 Dec 16  2018 xmas_file.txt


----------     -------   -------   --------- ------------ --------------
    |             |         |          |         |              |
    |             |         |          |         |           File Name
    |             |         |          |         |
    |             |         |          |         +---   Modification Time
    |             |         |          |
    |             |         |          +------------    Size (in bytes)
    |             |         |
    |             |         +----------------------        Group
    |             |
    |             +------------------------------          Owner
    |
    +------------------------------------------------   File Permissions
```

File Name
      The name of the file or directory.
Modification Time
      The last time the file was modified. If the last modification occurred more than six months in the past, the date and year are displayed. Otherwise, the time of day is shown.
Size
      The size of the file in bytes.
Group
      The name of the group that has file permissions in addition to the file's owner.
Owner
      The name of the user who owns the file.
File Permissions
      A representation of the file's access permissions. The first character is the type of file. A "-" indicates a regular (ordinary) file. A "d" indicates a directory. The second set of three characters represent the read, write, and execution rights of the file's owner. The next three represent the rights of the file's group, and the final three represent the rights granted to everybody else. We'll discuss this in more detail in a later lesson.

less

`less` is a program that lets us view text files. This is very handy since many of the files used to control and configure Linux are human readable.

What is "text"?

There are many ways to represent information on a computer. All methods involve defining a relationship between the information and some numbers that will be used to represent it. Computers, after all, only understand numbers and all data is converted to numeric representation.

Some of these representation systems are very complex (such as compressed multimedia files), while others are rather simple. One of the earliest and simplest is called *ASCII text*. ASCII (pronounced "As-Key") is short for American Standard Code for Information Interchange. This is a simple encoding scheme that was first used on Teletype machines to map keyboard characters to numbers.

Text is a simple one-to-one mapping of characters to numbers. It is very compact. Fifty characters of text translates to fifty bytes of data. Throughout a Linux system, many files are stored in text format and there are many Linux tools that work with text files. Even Windows systems recognize the importance of this format. The well-known NOTEPAD.EXE program is an editor for plain ASCII text files.

The **less** program is invoked by simply typing:

```
less text_file
```

This will display the file.

Controlling less

Once started, **less** will display the text file one page at a time. We can use the Page Up and Page Down keys to move through the text file. To exit **less**, we type "q". Here are some commands that **less** will accept:

Command	Action
Page Up or b	Scroll back one page
Page Down or space	Scroll forward one page
G	Go to the end of the text file
1G	Go to the beginning of the text file
/*characters*	Search forward in the text file for an occurrence of the specified *characters*
n	Repeat the previous search
h	Display a complete list less commands and options
q	Quit

Keyboard commands for the less program

file

As we wander around our Linux system, it is helpful to determine what kind of data a file contains before we try to view it. This is where the **file** command comes in. **file** will examine a file and tell us what kind of file it is.

To use the **file** program, we just type:

```
file name_of_file
```

The **file** program can recognize most types of files, such as:

File Type	Description	Viewable as text?

ASCII text	The name says it all	yes
Bourne-Again shell script text	A **bash** script	yes
ELF 64-bit LSB executable	An executable binary program	no
ELF 64-bit LSB shared object	A shared library	no
GNU tar archive	A tape archive file. A common way of storing groups of files.	no, use **tar tvf** to view listing.
gzip compressed data	An archive compressed with **gzip**	no
HTML document text	A web page	yes
JPEG image data	A compressed JPEG image	no
PostScript document text	A PostScript file	yes
Zip archive data	An archive compressed with **zip**	no

Various kinds of files

While it may seem that most files cannot be viewed as text, a surprising number can be. This is especially true of the important configuration files. During our adventure we will see that many features of the operating system are controlled by text configuration files and shell scripts. In Linux, there are no secrets!