Class Tutorials 2

In-class tutorials for Set 2

• Read over each tutorial before you attempt it to make sure you understand the instructions.
• Read the posted material for more information and additional examples.

Tutorials in in this set:

2.1 Viewing the calendar
2.2 Working with the date
2.3 Viewing history
2.4 Viewing who is logged on
2.5 Redirecting output

2.1 Viewing the calendar

The cal command without any option or argument will output the current month's calendar. Try:

```
cal
```

You can specify the year as an argument

```
cal 1897
```

You can specify the month and yar as arguments

```
cal 2 1991
```

To get information about other options and arguments for the command cal, examine its manual page

```
man cal
```

How do you display the **calendar** for the current month?

_____

How do you display the **calendar** for the same month last year.

_____

How do you display the **calendar** for the year 1980.

_____

Examine the manual page for each command.

2.2 Working with the date

Display the current time and date

```
date
```

To get information about other options and arguments for the command date, examine its manual page

```
man date
```

You can show formatted date and time. Check the manual page for the formatting options.

```
date +'Date: %m/%d/%Y%nTime: %r'
```

Display date in the following format: Today is Mon MM/DD/YY

Examine the manual page for each command.

2.3 Viewing history

List your **history** of commands
```
history
```

Redo command line 10 assuming that history shows at least 10 commands
```
!10
```

To get information about other options and arguments for the command history, examine its manual page
```
man history
```

Repeat last cal command but pipe it to the word count command to return the number of lines in the output
```
!cal | wc -l
```

Print without executing event number 6
```
!6:p
```

What will the following command do?
```
history 7
```

What will the following command do?
```
!!
```

What will the following command do? Explain
```
History 5
```

Give the command to redo line 21 of your history

Examine the manual page for each command.

2.4 Viewing who is logged on

Examine the manual pages for the command **who**. What does the short description say?

Type the following command and note what it does
```
who am i
```

Examine the manual page for each command.

2.5 Redirecting output

Now type the cat command and redirect its STDOUT to a file called "randomNames"
```
cat > randomNames
```

Your cursor will just sit there and wait for you to do something, type the following names, pressing Enter after each one.
```
Marie Curie
Albert Einstein
Mark Twain
Wolfgang Amadeus Mozart
Stephen Hawking
Isaac Newton
```

Finish the random names list and press CTRL+D to finish. What does the CTRL+D stand for?

Display the calendar for December 2009 and save the output in a file called Dec.2009.
What command did you use?

Append to your Dec.2009 file the current date and time:

What is the difference between running the following two commands?
```
cal 1 2010; cal 2 2010; cal 3 2010 >> winter.2010
(cal 1 2010; cal 2 2010; cal 3 2010) >> winter.2010
```

List your history of commands and save them in a file called tutorial2.txt *(at the prompt, type the following line, followed by the <enter> key)*
```
history > tutorial2.txt
```

Note the results and your observations after executing each command.
Examine the manual page for each command.

2.6 Using pipes

• *You can use a pipeline (symbolized by "|") to make the output of one command serve as the input to another command.* This idea can be used to create a combination of commands to accomplish something no single command can do.
Enter this command:
```
$ echo "cherry apple peach"
cherry apple peach
```

Okay, let's say we want to sort these words alphabetically. There is a command "sort", but it sorts entire lines, not words, so we need to break this single line into individual lines, one line per word.
Step one: pipe the output of "echo" into a translation (tr) command that will replace spaces with linefeeds (represented by "\n"):

```
$ echo "cherry apple peach" | tr " " "\n"
cherry
apple
peach
```

Success: each word appears on a separate line. Now we are ready to sort.
Step two: add the sort command:

```
$ echo "cherry apple peach" | tr " " "\n" | sort
apple
cherry
peach
```

Let's try reversing the order of the sort:

```
$ echo "cherry apple peach" | tr " " "\n" | sort -r
peach
cherry
apple
```

• **Remember:** *A pipeline ("|") takes the output of one command and makes it the input to another command.*

Examine the manual page for each command.

3:31:15 PM 1/19/2012