**Linux Exercises: Basic Commands to Survive**

1. Run a command to find your current location on the server

pwd

1. Change directory from your current location to /var/log

cd /var/log

1. Go back up a directory

cd ..

1. Find your current location again

pwd

1. List all the files in the current directory

ls

1. Now list all the files with the long listing

ls -l

1. Now list all the files with the long listing in reverse order, with the newest appearing at the bottom of your screen

ls -ltr

1. Change to root directory

cd /

1. Change back to your home directory

cd ~

1. Go up a level in directory structure

cd ..

1. Find out more information about the ls command

man ls

1. Go back to /var

cd /var

1. Show the contents of this directory with details reverse sorted by size (you may have to use man to help)

ls -ltrS

1. Find out the hardware version you are running

uname -i

1. Run the following command to store your results in the directory that the trainer has provided â substitute your name and the date in your file

history > name.YYMMDD.module2

**Linux Exercises: Advanced Commands**

If you have not been doing this all along, clean up your home directory removing any files you no longer need.

Run the fixGenerator.sh script in your home directory again to prep for exercises in this module.

./fixGenerator.sh &

Use Linux to complete the following exercises.

1. Create a directory named logs.

mkdir ~/logs

1. Move the log output from the fixGenerator script into the logs directory. (Be sure the script has finished before doing this.)

mv fixlog\* logs/.

1. From the command line, replace all instances of MTHREE in the file with M3 and put the output into a new file named log in the logs directory.

cat fixlog\*.log | sed 's/MTHREE/M3/g' > fixlog2.log

1. Run a command to pull all fill messages from fixlog2.log and put the output into a new logfile named log. (You may need to look up how to tell if a message is a fill.)

grep 35=8 fixlog2.log | grep -v 32=0 > fills.log

1. Run a command to pull all cancel acknowledgement messages (39=4) from fixlog2.log into a new log named log in the same directory.

grep 39=4 fixlog2.log > cancels.log

1. Run a command to create a new log file named log and add the partial fills from fills.log to the new file.

grep 39=1 fills.log > partiallFills.log

1. Use awk to create a new file out of the partial fill log that has the following tags only: Symbol (55); orderID (11); side(54); fill price (31); fill quantity (32); execution id (17). Name the file parsedPartialFills.log and make sure you print the columns in the order listed here.

awk '{print $7, $9, $13, $10, $15, $16}' partiallFills.log > parsedPartialFills.log

1. Using an editor, remove the first part of every fix tag (so you are left with the value only) and turn the file into a comma separated list with no spaces. This is how you might have to get a file ready to send to a trader.
2. In the file, add a row of column headers separated by commas. The headers should be Symbol, OrderID, Side, Price, Qty, and ExecID.
3. Save the file as .module10.csv in the location specified by your instructor.
4. Make a copy of the cancels file and name it log.

cp cancels.log cancels2.log

1. Open the cancels2.log file in an editor. Find the first symbol (tag 55) in the first line and add the letter A to the beginning of the value. (If it was 55=GOOG, it will become 55=AGOOG.)
2. Run a difference between the original cancels file and the new file you just edited.

diff cancels.log cancels2.log

1. Now run the history command and put it into an output file named YYMMDD.module10 in the directory specified by your instructor.

**Linux Exercises: Using Vi**

## Part 1: Avengers

Practice the vi commands presented in the Lesson 8 slides before completing these exercises.

1. Open a new file named avengers in your home directory in your editor and paste in the following text:

Six stones, three teams, one shot. Five years ago, we lost. All of us. We lost friends. We lost family. We lost a part of ourselves. Today, we have a chance to take it all back. You know your teams. You know your missions. Get the stones. Get them back. One round trip each. No mistakes. No do-overs. Most of us are going somewhere we know. that doesn't mean we should know what to expect. Be careful. Look out for each other. This is the fight of our lives, and we're gonna win. Whatever it takes. Good luck.

1. Exit and save the file.
2. Run a word count on the file.
3. Edit the file to add a blank line and type in "word count" followed by the value you just got.
4. Now use search and replace within the editor to replace every instance of the word the with THE. Ensure that you only catch the word the and not words that contain the letters the (like them).
5. Copy the first line of the file and paste it after your word count entry at the bottom of the file.
6. Delete the first line of the file.
7. Undo that deletion.
8. Insert at the top of the file the text "Captain America â Endgame" followed by an empty line.
9. Save the changes and exit the editor.
10. Find out the number of lines in the file now.
11. Write the number of lines into the file one line above the word count with the text "line count" plus the value.
12. Delete "Good luck" wherever it occurs in the file.
13. Save and exit the file
14. Rename the file yourname and put it in the location specified by the instructor.

### Answer

Run a comparison on the two versions of the avengers file, but the final file should look like this:

Captain America - Endgame

Six stones, three teams, one shot. Five years ago, we lost. All of us. We lost friends. We lost family. We lost a part of ourselves. Today, we have a chance to take it all back. You know your teams. You know your missions. Get THE stones. Get them back. One round trip each. No mistakes. No do-overs. Most of us are going somewhere we know. that doesn't mean we should know what to expect. Be careful. Look out for each other. This is THE fight of our lives, and we're gonna win. Whatever it takes.

line count 6

word count 98

## Part 2: Log Files

Run the fixGenerator.sh script again from your home directory using the following commands:

cd ~

./fixGenerator.sh &

You should have a fix log output now from running the script earlier.

1. Run a search for all new order singles in the fix log output and put the output of that search into a file named log in your home directory.
2. Open newOrders.log in your editor.
3. Duplicate the first line in the file.
4. Go to the fifth line in the file and duplicate it also.
5. Find and replace every instance of MTHREE with M3.
6. Find and replace every ; (semicolon) with a , (comma) and ensure there is no space after each ,
7. Save and exit the file
8. Rename your file to .log and save it in the directory specified by the instructor.

### Answer

Because the log file is randomly generated each time, each person's file will be different. Check specifically for the following:

1. All the lines are new orders â e.g., 35=D.
2. The first and fifth lines are duplicated.
3. All MTHREEs have been replaced with M3.
4. Every ; is replaced with , essentially creating a csv file.

**Solutions:Variables**

Complete the following activities using the skills presented in Lesson 4.

1. Create a variable named STOCK and assign it a value of AAPL

STOCK=AAPL

1. Print the value of the variable out to the screen

echo $STOCK

1. Change the value of the variable STOCK to be the previous value with \_US appended onto the end

STOCK=$STOCK"\_US"

1. Create a new variable named PRICE with value 290.84

PRICE=290.84

1. Print out to the screen a sentence âThe current price of is â where you call the [variables](https://academy.engagelms.com/mod/page/view.php?id=180285) and get their values

echo "The current price of $STOCK is $PRICE"

1. Create a variable named WHOAMI and assign it your current username. Do this in one command.

WHOAMI=$USER

1. Using just $ and () assign a variable named SERVERTYPE by using the uname command

SERVERTYPE=$(uname)

1. Write a command to show to the screen the value of running the command assigned to SERVERTYPE

echo "$SERVERTYPE"

1. Assign a variable named CMD the command uname -i (not the value of running the command but the command itself)

CMD="uname -i"

1. Using echo and the variable CMD, show how you echo the value of CMD (not execute the command)

echo $CMD

1. Using echo and the variable CMD, print what the CMD returns when run to the screen

echo `$CMD`

1. Change directory to your home directory

cd ~

1. Create a directory named scripts

mkdir scripts

1. Print out your $PATH Value

echo $PATH

1. Append your newly created script directory to your path variable

export PATH=$PATH:~/scripts/

1. Check you can see your new path appended to the end of the path variable

echo $PATH

1. If you have Algo1 still created from previous exercises, remove the directory structure and all files within it
2. cd ~

rm -rf Algo1

1. Make a directory named Payment1

mkdir Payment1

1. Within Payment1, make three directories named logs, configuration, and scripts

mkdir Payment1/logs Payment1/configuration Payment1/scripts

1. Within scripts, create an empty file named start.sh

touch Payment1/scripts/start.sh

1. Within configuration, create an empty file named instance.properties

touch Payment1/configuration/instance.properties

1. Give everyone permission to read the instance.properties file, but assign write and execute permissons only to yourself

chmod 744 Payment1/configuration/instance.properties

1. Set permissions on the file start.sh so that everyone can execute and so that the group and owner can read and write

chmod 771 Payment1/scripts/start.sh

1. Direct the output of man cat into a file in your home directory named cat.output

man cat > cat.output

1. Change the permission of that file so everyone can read it but only the group can write and execute

chmod 474 cat.output

1. Go back to your home directory

cd ~

1. Prove your location

pwd

1. Remove the directory Payment1 along with all subdirectories

rm -rf Payment1

1. Delete the cat.output file, but with a warning first to the user

rm -i cat.output

1. Create a directory named temp

mkdir temp

1. Set a variable named MYTEMP that references the new temp directory you just created

MYTEMP=~/temp

1. Run a command to see the value of the MYTEMP variable

echo $MYTEMP

1. Now set the value of $MYTEMP to nothing

MYTEMP=

1. Remove the temp directory that you created

rm -rf temp

1. Run the following command to store your results in the specified directory the trainer gave. Replace the name and date with the correct values:

History > name.YYMMDD.module4