HDFS HOMEWORK DUE: SEPTEMBER 5, 11:59PM EST VIA GRADESCOPE

1. Instructions

1.1. **What to look at.** The only references you may use for this lab are:

- The HDFS documentation `https://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html`
- Any material on Canvas

Do not use stackoverflow, chegg, or LLMs because: (1) they can often give you a really bad answer (2) it is an academic integrity violation.

1.2. **How to collaborate.**

- Write down (in your file) the names of people you collaborate with.
- You *cannot* copy/email/transmit files to another person. Your code must be typed by you without looking at other people's code.
- You cannot look at another person's work (e.g., hdfshw.txt file).

1.3. **The Assignment.** You now have 3 file systems to deal with. First is the local file system on your computer which you access through visual folders or on the terminal. That file system lives on the disk in your computer. Second is the local file system of the gateway node that you log into (e.g., e5-cse-trantor01.cse.psu.edu). This file system lives on the disk of that node. You access it through the terminal (after you have done ssh) with commands like **ls** to see what is in the current directory and **cd** to change directories. This is, for example, where one copy of your github repository should be. The third file system is HDFS. You access it using **hdfs** commands once you have logged into the cluster as shown in class (this means that logging into the cluster gives you access to two different file systems). This is where the big data is stored (don't store code in HDFS).

What you need to do:

(1) This assignment uses the github repository you cloned on the cluster. Use vi to edit the files for this assignment.
(2) Use updatestarter to get the files for this assignment. You will get an hdfshw folder and inside there will be the file hdfshw.txt. There will also be a "words" directory in github containing files called part-00000 through part-00003 (in general it is bad to have data files in github, but this will simplify what you need to do).
(3) Files in your HDFS home directory have the prefix **/user/$USER/**. So if you had a file called **cooldata**, its location in HDFS is **/user/$USER/cooldata**. For context, $USER is a shell variable and the hdfs command will replace it with your penn state user id. So, for example, if your psu id is abc3219, then for you, /user/$USER is the same as /user/abc3219.
(4) Using the HDFS documentation link (top of page) create a directory **myhdfshwdata** in your HDFS home directory. Write the command(s) you used to do this in hdfshw.txt.
(5) Upload the files part-00000 through part-00003 (they are in the "words" directory in your github repository) into the hdfs directory you just created. Write the command you used to do this in hdfshw.txt. **This should include the commands you use to navigate into the words directory from the shell on the gateway node.**
(6) Display the contents of your home directory in HDFS. Write the command(s) you used to do this in hdfshw.txt.
(7) Display the contents of your home directory on the local drive of the gateway node you are logged into. Write the command(s) you used to do this in hdfshw.txt.
(8) Display the contents of the **myhdfshwdata** directory you created previously. Write the command(s) you used to do this in hdfshw.txt.
(9) Move (not upload) the files from the **myhdfshwdata** directory to a new HDFS directory called **words**. Write the command(s) you used to do this in hdfshw.txt.
(10) Delete the **myhdfshwdata** directory on HDFS. Write the command(s) you used to do this in hdfshw.txt.
(11) Display the last 1kb of the part-00002 file that you earlier put in HDFS. Write the command you used to do this in hdfshw.txt.
(12) Display information about the filesize in bytes, replication information, user name of the file owner, and modification date for the part-00001 file that you earlier put in HDFS. Write the command(s) you used to do this in hdfshw.txt. There are two different commands that can solve this question.
(13) Make sure to add/commit/push etc. your changes to hdfshw.txt (When in doubt check what **git status** tells you). Then submit to gradescope.
(14) Check what you have on gradescope. Make sure it is what you intended to submit.