RIT | Golisano College of Computing and Information Sciences
School of Information

NSSA-220 Task Automation Using Interpretive Languages
Lab 1: Linux Commands

Instructions

For the lab, complete the required tasks. Additionally, review the *submission document* for the mandatory screenshots and the grading rubric. You must provide screenshots from the terminal that includes the commands and relevant output to validate the completion of each task. The lab should be completed and submitted individually, but feel free to work with other classmates and ask for help from your instructor and the teaching assistant as needed. The "official" due date for each assignment will be posted to myCourses. When complete, submit the document to the assignment section in myCourses.

Preparation

- Read through this document
- Attend Lecture and take notes
- Review the reading assignments

Activity Summary

Activity 1 – Modify your shell prompt
Activity 2 – Linux Command One Liner
Activity 3 – Parsing File Information

Activities

Activity 1 – Configure the Shell Prompt

For this activity, the .bashrc file will be edited to show your RIT ID. Throughout the labs, you will be providing screenshots to verify that the activity/task has been completed, and the screenshot must show your RIT ID to receive credit. The file is located in the student home directory or /home/student (see Figure 1). To change the prompt, open a Terminal and using an editor of your choice (vim is recommended), add the following line under *"User specific aliases and functions,"* in the *".bashrc"* file (see Figure 2). Please note the spacing and use of single quotes.

```
PS1 = '[yourID \w]\$ '
```

Figure 1 – Location of .bashrc File

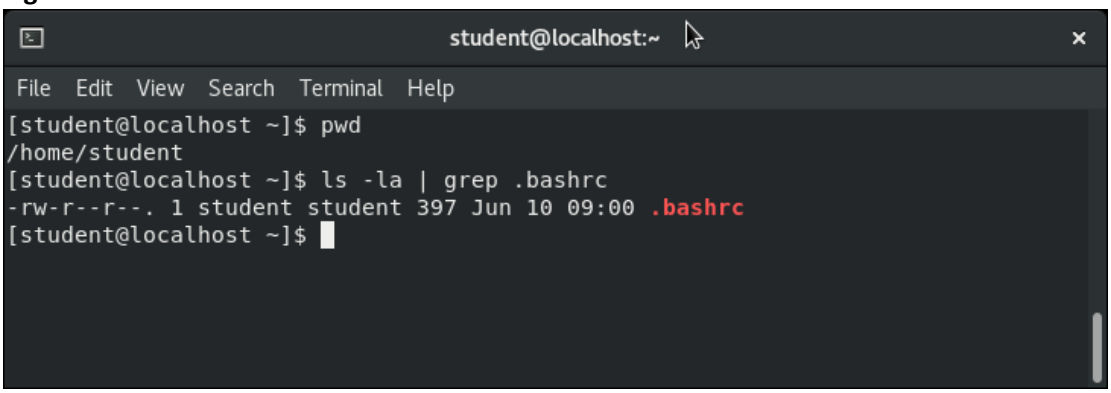
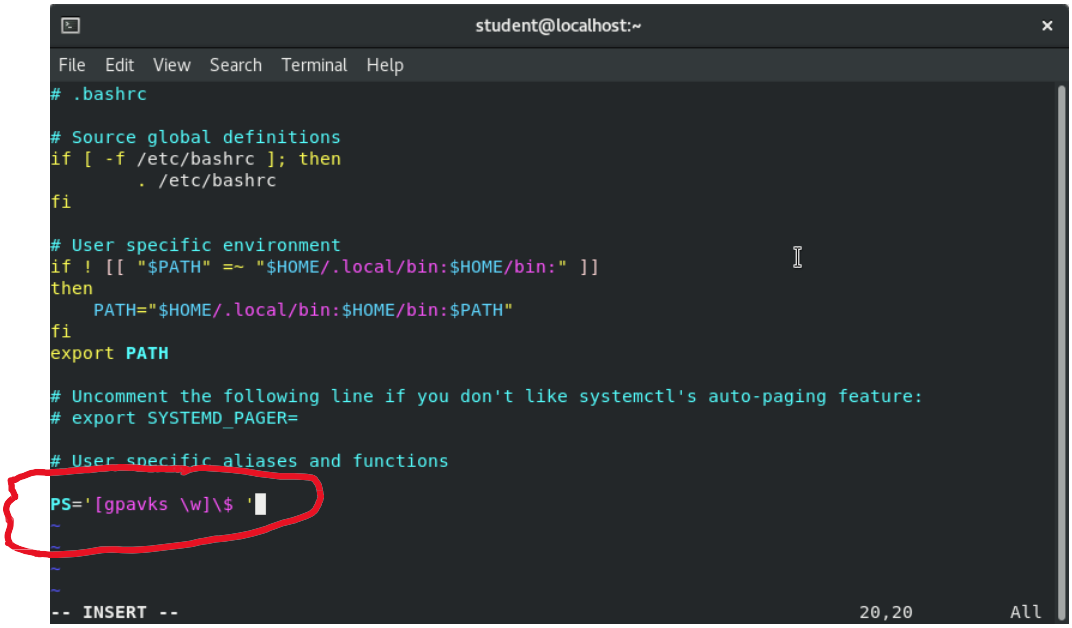
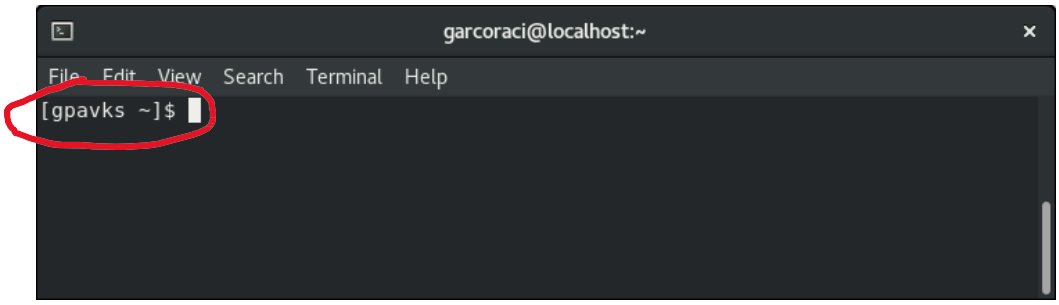


Figure 2 – Edited File



Please note that you will need to exit the terminal and open a new instance for the change to take effect, your prompt should look similar to Figure 3.

Figure 3 – New Prompt



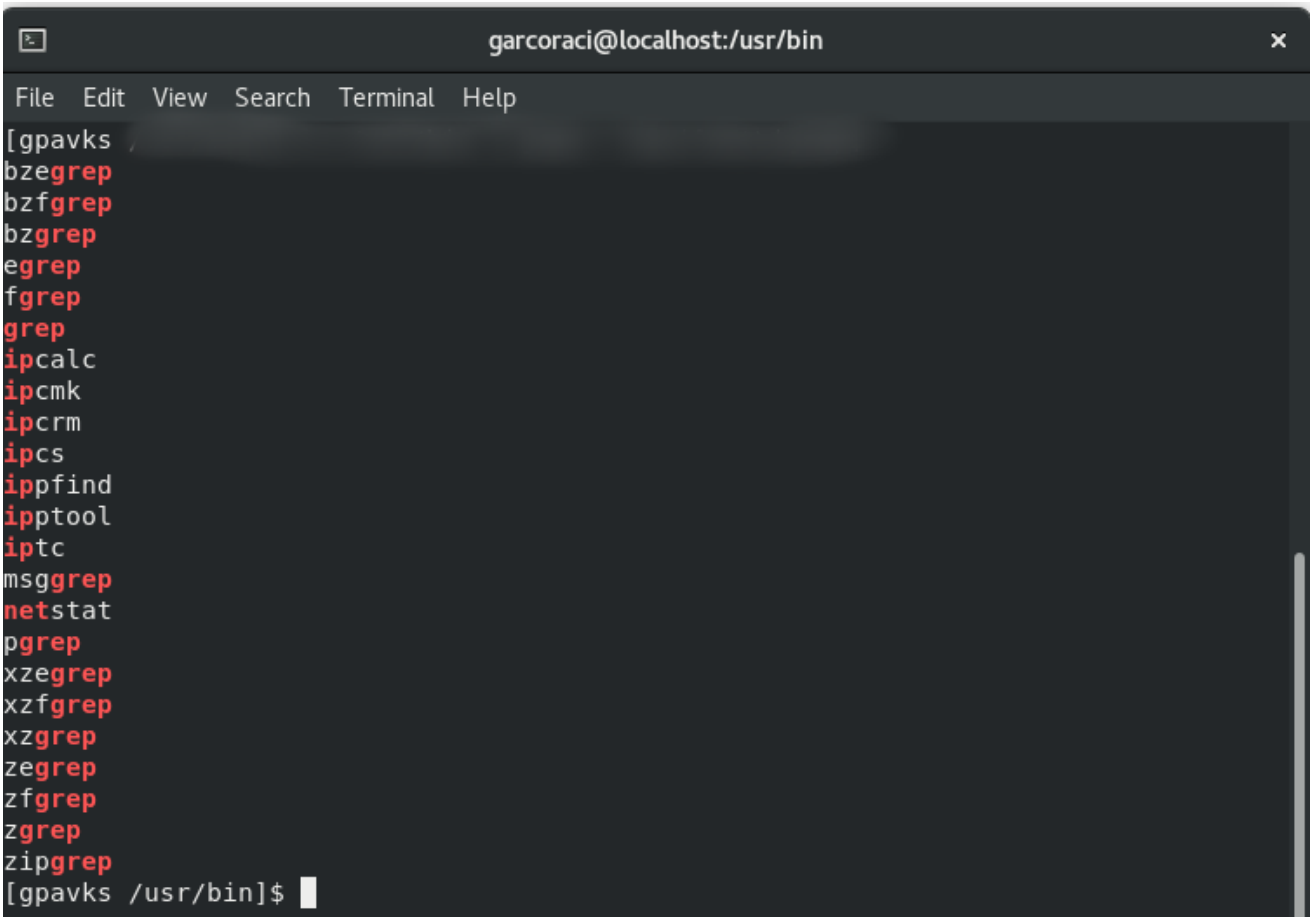
Activity 2 –Linux Command One-liners

For this activity, you will be performing a series of tasks. For each task, include a screenshot that clearly shows the command or commands you used to accomplish it. The screenshot must show your username the prompt. Figure 4 provides a sample of the output (obviously, the command used is blurred); your output may differ depending on the CentOS version. Regardless, the output should only show files starting with 'ip' and 'net' or ending with 'grep.'

Task 1

Write a single command that outputs a list of all files in /usr/bin that begin with “ip”, and “net”, or end with “grep”. Hint: when using a $ in a regex, the $ comes *after* the string you want to match at the end of line.

Figure 4 – Sample Output from Command



Task 2

This task will require two commands. The first command will output a list of all subdirectories in the /etc directory that a standard user *cannot* access (i.e., they are denied permission). The command's output needs to be redirected to a file called *'output.log*.' Next, use the **`cut`** command to show the absolute path of the subdirectories without any special characters like the colon. See the output below for a sample of the expected output and format. Include a screenshot of output.log file using the cat command.

Figure 5 – Sample Output for Task 2

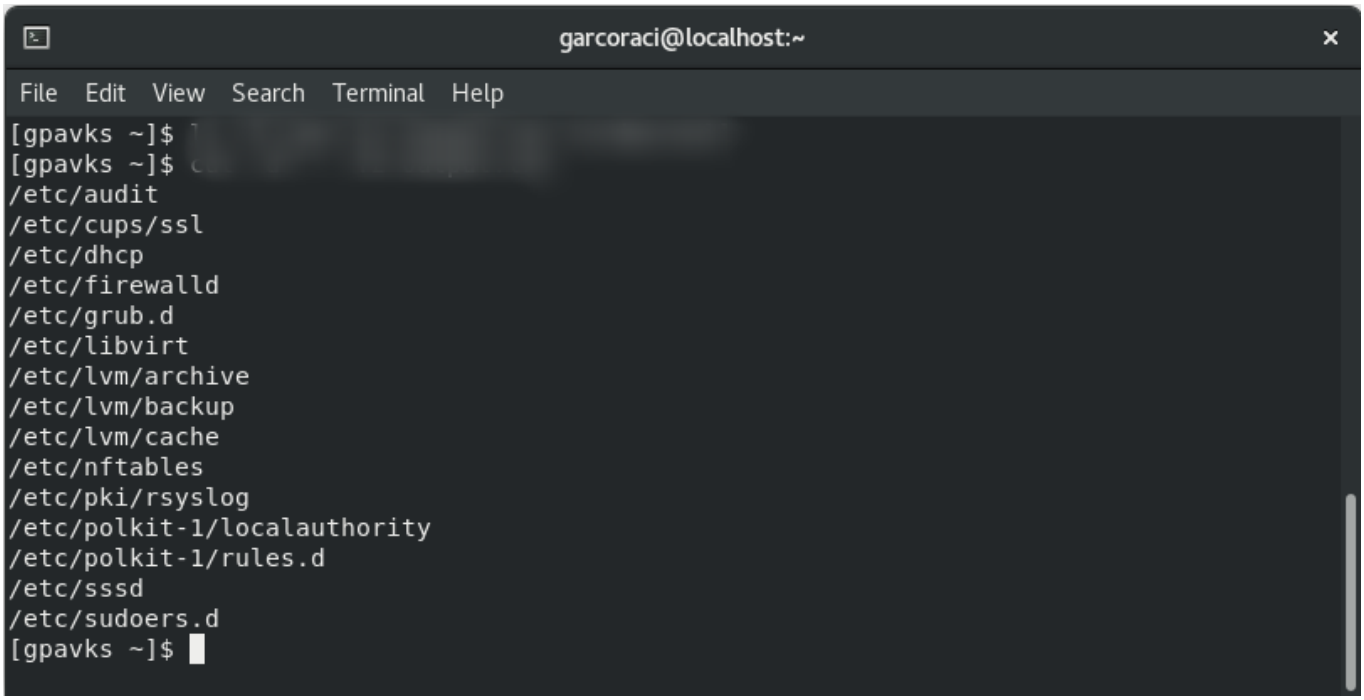


Figure 6 – Contents of the "*output.log*" File

Ted Williams was a baseball player who played Major League Baseball for 19 seasons between 1939 and 1960 and is considered "*The Greatest Hitter Who Ever Lived*." Some of his statistics are stored in TeddyBallgame.csv on myCourses. Several tasks will be completed using this file. See the top line of the file for the definition of each field in the file. Note that since this is a .csv file, all fields are separated by commas.

Task 1

Using TeddyBallgame.csv, write a single command that removes the top line of the file and replaces commas with spaces. The output should be redirected to a file TeddyBallgame.txt and to standard output (Figure 6). Use the cat

command to show the contents of the TeddyBallgame.txt file (Figure 7). The output from the cat command should match the information in standard output.

Figure 6 – Expected Output from the Command

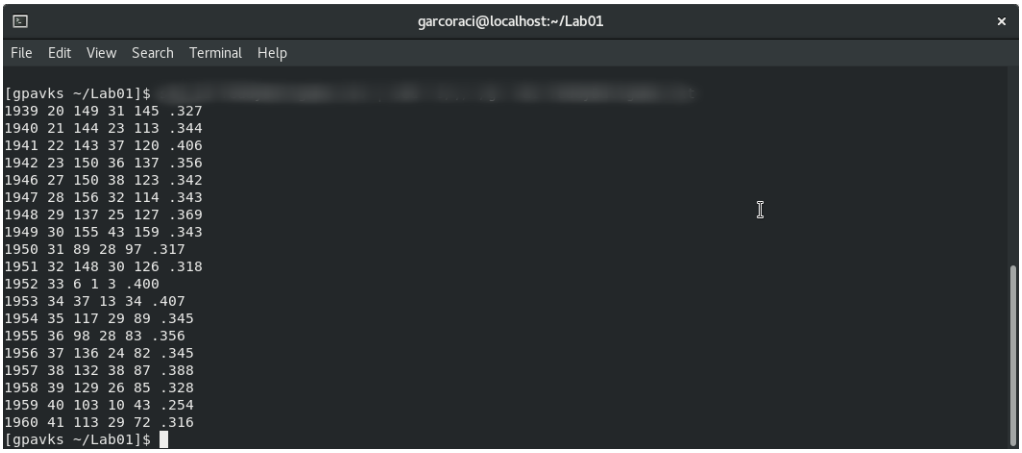


Figure 7 – Expected Output from the `cat` Command

Task 2

Using TeddyBallgame.txt from the previous task, write a single command to output the list of all seasons where Ted Williams had 100 or more Runs Batted In or RBI. The output should only include the year and the number of Runs Batted In for that year. For sample output, refer to Figure 8. Please note that the command should not alter the contents of the file. Hint: Using an awk if statement is helpful.

Task 3

Using the TeddyBallgame.txt from Task 2, write a single command to output the list of all seasons where Ted Williams hit 30 or more Home Runs. The list should be sorted from most to least Home Runs, and the final output should include the fields for the Year, Home Runs, Runs Batted In, and Batting Average for those seasons. Refer to Figure 9 for the expected output.

Figure 9 – Expected Output from Task 3

Task 4

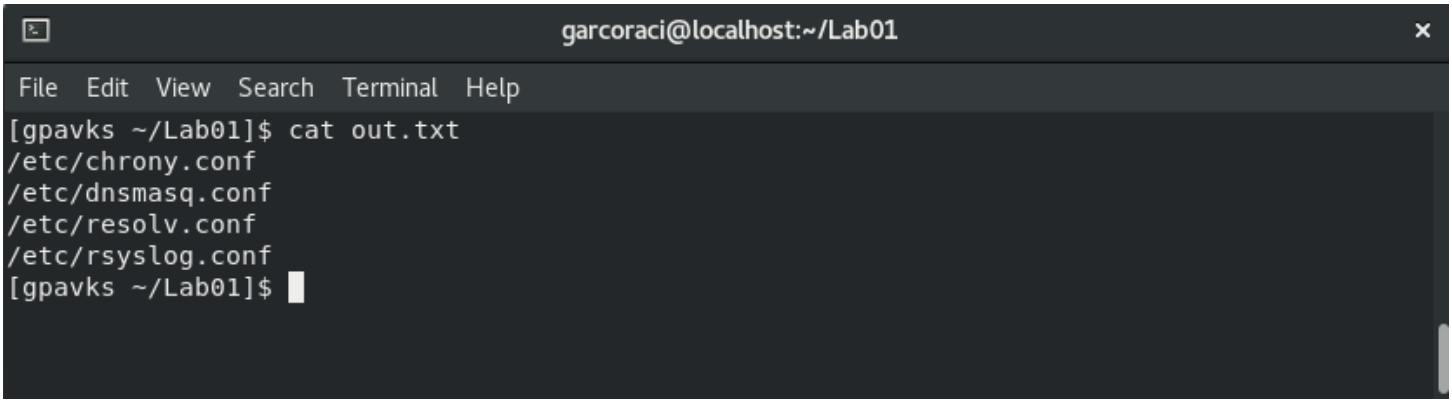
Modify the command used in Task 3 to sort the seasons by Runs Batted In from low to high. Refer to Figure 10 for the expected output.

Figure 10 – Expected Output for Task 4

Task 5

Write a single command that stores the names of all .conf files located in /etc that contain an IP address beginning with "192.168" and redirects the output to a file called "out.txt." Use the cat command to show the contents of the file and include the screenshot. See expected contents of out.txt below. Any errors should be redirected to /dev/null so that they are not in the file.

Figure 11 – Expected Output for Task 5



Grading Rubric and Screenshots

↓↓↓↓↓↓↓↓↓↓↓↓ **PLEASE READ** ↓↓↓↓↓↓↓↓↓↓↓↓

For the submission, screenshots are required to validate the completion of the tasks. For each screenshot, specific information is requested. This information must be in a single screenshot; multiple screenshots will not receive credit. **ALL** information must appear in a **single** screenshot when requested. Screenshots that are illegible, blurry, or otherwise unreadable will not receive credit. Any attempt to alter the information in the screenshots is academic dishonesty and results in a zero grade for the assignment.