1. **grep Lane GrepLab**
2. **Text

   Description automatically generated**
3. The grep commands searches every line, and only display the lines that match a pattern. If we add the word Lane, then it would only look for lines that contain the word, Lane.
4. **grep ^H GreLab**
5. **Text

   Description automatically generated**
6. This command searches for any line that starts with a capital H. By using the character (**^**) you are specifying that the line have to start with H. the **^** character means that a line must begin with.
7. **grep 000$ GrepLab**
8. **Text

   Description automatically generated**
9. This command will search for lines that end with 3 0s. using the dollar sign (**$**) during a search command means the end of a line. In this command we are using it to mean that we want the end of a line to have 3 0s.
10. **grep -v 408 GrepLab**
11. **Text

    Description automatically generated**
12. With this command we are searching for lines that do not contain the number 408. By using the option (**-v**) when searching a file, we are telling the command to ignore the number **408**. So, it only displays lines that do not contain the number **408**.
13. **cat GrepLab | grep “././35”**
14. **A picture containing text

    Description automatically generated**
15. With this command we are searching for dates that end in **35**. We are doing this by using a dot (**.**). when using the dot during a search command it means to match any character. Since the date format is **mm/dd/yy** we make it match any 2 characters followed by a slash, followed by any two characters, then a slash, then the number 35.
16. **cat Greplab | grep “^[A-Z].\*:8..-“**
17. **Text

    Description automatically generated with medium confidence**
18. With this command we are searching for area code numbers that start with the number **8**. We do this by searching for lines that start with capital letters **[A-Z]** followed by any character (**.**), we use the star **\*** to match any previous character until we find a colon (**:**). Then find the number 8 followed by any 2 characters (**..**) then a dash (**-**).
19. **cat GrepLab | grep "[A-Z][a-z][a-z][a-z][a-z] [A-Z]"**
20. **Text

    Description automatically generated**
21. With this command we are searching all lines containing an uppercase letter, followed by 4 lowercase letters, a space and one uppercase letter. We do this by using capital letters as the start [**A-Z**], then followed by 4 lower case letters using [**a-z**] four times, then we add a space and then use [**A-Z**] to find the capital letter.
22. **cat GrepLab | grep -E ":[0-9]{2,3} "**
23. **Text

    Description automatically generated**
24. With this command we are searching for addresses that contain 2 or 3 numbers. **The -E option allows the use of regular expressions**. We do this by first finding the colon (**:**), then find any number **[0-9]**, then match the previous numbers 2 or 3 times **{2,3}**. **This is the easiest way I found to do this command, but I’m pretty sure there are better ways**.
25. **grep -E 'MA|IL' GrepLab**
26. **Text

    Description automatically generated**
27. With this command we are searching for lines with people that line in **MA** or **IL**. We do this by using the word MA and using the (**|**) to say and then the word **IL**.
28. **grep -v 'Street' GrepLab | grep -v 'St'**
29. **Text

    Description automatically generated**
30. With this command we are searching for people with addresses that are not a street. We do this by using the previous options for the command (**-v**) to show only non-matching lines. In this instance we are looking for line that do not contain the word **Street** or the abbreviated **St.**

**SOURCES:**

“Grep(1) - Linux Manual Page.” *Man7.org*, man7.org/linux/man-pages/man1/grep.1.html.

McKay, Dave. “How to Use Regular Expressions (Regexes) on Linux.” *How-to Geek*, www.howtogeek.com/661101/how-to-use-regular-expressions-regexes-on-linux/. Accessed 6 Feb. 2023.

“Using Grep & Regular Expressions to Search for Text Patterns in Linux | DigitalOcean.” *Www.digitalocean.com*, www.digitalocean.com/community/tutorials/using-grep-regular-expressions-to-search-for-text-patterns-in-linux. Accessed 6 Feb. 2023.

‌

<https://www.cyberciti.biz/faq/grep-regular-expressions/>