**1. Print line numbers of lines beginning with “O”**

bash

Copy code

sed -n '/^O/=' g.txt

**Explanation:** The sed command checks each line starting with “O” (^O) and prints only the line numbers (=). The -n option suppresses automatic printing, so only matching line numbers are shown.

**Expected Output:**

Copy code

1

2

7

**2. Delete digits in the given input file**

bash

Copy code

sed 's/[0-9]//g' g.txt

**Explanation:** The s/[0-9]//g command searches for all digits ([0-9]) and removes them globally (g flag).

**Expected Output:**

sql

Copy code

Open the door to opportunity.

Ordinary day with surprises.

- This line has some digits.

Room is booked.

...

**3. Delete lines that contain both "BEGIN" and "END"**

bash

Copy code

sed '/BEGIN.\*END/d' g.txt

**Explanation:** The sed command searches for lines containing both "BEGIN" and "END" and deletes them (d). The .\* wildcard matches any characters between "BEGIN" and "END."

**Expected Output:**

sql

Copy code

Open the door to opportunity.

Ordinary day with surprises.

12345 - This line has some digits.

Room 101 is booked.

BEGIN but there is no END in this line.

...

**4. Delete lines that contain "BEGIN" but not "END"**

bash

Copy code

sed '/BEGIN/ { /END/!d }' g.txt

**Explanation:** This command deletes lines that have "BEGIN" but lack "END." It matches lines with "BEGIN" (/BEGIN/) and then checks if they do not have "END" (/END/!) before deleting them (d).

**Expected Output:**

sql

Copy code

Open the door to opportunity.

Ordinary day with surprises.

12345 - This line has some digits.

Room 101 is booked.

...

**5. Delete the first character in each line in a file**

bash

Copy code

sed 's/^./&/' g.txt

**Explanation:** The command searches for the first character in each line (^.) and removes it.

**Expected Output:**

sql

Copy code

pen the door to opportunity.

rdinary day with surprises.

2345 - This line has some digits.

oom 101 is booked.

...

**6. Delete the last character in each line in a file**

bash

Copy code

sed 's/.$//' g.txt

**Explanation:** This command removes the last character on each line by matching any character at the end (.$) and deleting it.

**Expected Output:**

sql

Copy code

Open the door to opportunity

Ordinary day with surprises

12345 - This line has some digits

Room 101 is booked

...

**7. Delete the first character if it’s numeric in each line**

bash

Copy code

sed 's/^[0-9]//' g.txt

**Explanation:** The sed command checks if the first character is numeric (^[0-9]) and deletes it if so.

**Expected Output:**

sql

Copy code

Open the door to opportunity.

Ordinary day with surprises.

2345 - This line has some digits.

Room 101 is booked.

...

**8. Delete the character before the last character in each line**

bash

Copy code

sed 's/.\(.\)$/\1/' g.txt

**Explanation:** The command removes the character before the last character on each line by matching any single character before the last character (.\(.\)$) and replaces it with the last character (\1).

**Expected Output:**

sql

Copy code

Open the door to opportunity.

Ordinary day with surprises.

12345 - This line has some digit.

Room 101 is booke.

...

**9. Swap the first and second words in each line**

bash

Copy code

sed -E 's/^([[:alnum:]]+) ([[:alnum:]]+)/\2 \1/' g.txt

**Explanation:** This command uses grouping to swap the first and second words. It captures two alphanumeric words (^([[:alnum:]]+) ([[:alnum:]]+)) and rearranges them as \2 \1 to swap their positions.

**Expected Output:**

sql

Copy code

door Open the to opportunity.

day Ordinary with surprises.

- 12345 This line has some digits.

101 Room is booked.

...

**10. Replace "mcis" with "sois" in the first five lines**

bash

Copy code

sed '1,5s/mcis/sois/g' g.txt

**Explanation:** This command only affects lines 1 to 5 (1,5) and replaces any occurrence of "mcis" with "sois" (s/mcis/sois/g).

**Expected Output:**

arduino

Copy code

sois program is at the university.

Another line.

This line also has "mcis".

mcis not affected in this line.

The program from "mcis" now shows "sois".

...

Let me know if you would like a full list for the remaining commands!

4o

**You said:**

give next 10

**ChatGPT said:**

ChatGPT

Certainly! Here are the next 10 commands (11-20), with explanations and expected outputs using g.txt as the sample file.

**11. Add “SOIS” prefix to all the lines**

bash

Copy code

sed 's/^/SOIS /' g.txt

**Explanation:** This command inserts "SOIS " at the beginning of each line by matching the start of the line (^) and replacing it with "SOIS ".

**Expected Output:**

sql

Copy code

SOIS Open the door to opportunity.

SOIS Ordinary day with surprises.

SOIS 12345 - This line has some digits.

SOIS Room 101 is booked.

...

**12. Add “.” at the end of each line in the file**

bash

Copy code

sed 's/$/./' g.txt

**Explanation:** This command adds a period at the end of each line by matching the end of the line ($) and appending a period.

**Expected Output:**

sql

Copy code

Open the door to opportunity..

Ordinary day with surprises..

12345 - This line has some digits..

Room 101 is booked..

...

**13. Pick the line with a Social Security number in the format 999-99-9999**

bash

Copy code

sed -n '/[0-9]\{3\}-[0-9]\{2\}-[0-9]\{4\}/p' g.txt

**Explanation:** This command searches for lines containing a Social Security number pattern ([0-9]\{3\}-[0-9]\{2\}-[0-9]\{4\}) and prints them. The -n flag suppresses automatic printing, so only matching lines are displayed.

**Expected Output:**

yaml

Copy code

Social security number: 123-45-6789

**14. Pick the valid IP address of the computer (e.g., 192.168.0.1)**

bash

Copy code

sed -n '/\b[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}\b/p' g.txt

**Explanation:** This command matches IP addresses with the standard four-segment format. The \b ensures word boundaries so the IP address is an isolated match. Only lines with valid IP addresses are printed.

**Expected Output:**

yaml

Copy code

Valid IP: 192.168.0.1.

**15. Pick the valid URL beginning with “http://”**

bash

Copy code

sed -n '/http:\/\/[a-zA-Z0-9.-]\+\.[a-z]\{2,4\}/p' g.txt

**Explanation:** This command looks for URLs starting with "http://" followed by typical domain patterns. The -n option restricts output to matching lines.

**Expected Output:**

arduino

Copy code

Website URL: http://manipal.edu.

**16. Pick the valid email address**

bash

Copy code

sed -n '/[a-z0-9.-]\+@[a-z0-9.-]\+\.\(com\|net\)/p' g.txt

**Explanation:** This command matches valid email addresses that contain lowercase letters, numbers, hyphens, and dots, with domains ending in ".com" or ".net".

**Expected Output:**

css

Copy code

Email address: sample@manipal.edu

**17. Print all lines containing words that start with "f" and end with "s"**

bash

Copy code

sed -n '/\b[fF][a-zA-Z]\*s\b/p' g.txt

**Explanation:** This command matches words that start with "f" or "F" and end with "s" by using word boundaries (\b) to isolate them.

**Expected Output:** (Assuming lines contain such words)

Copy code

Files and folders.

**18. Print lines containing a capital "L," but not as the first character**

bash

Copy code

sed -n '/^[^L]\*L/p' g.txt

**Explanation:** This command finds lines with an uppercase "L" that is not the first character. The pattern ^[^L]\*L ensures "L" isn’t at the start.

**Expected Output:**

scss

Copy code

She loves L in the middle.

**19. Match dates in the American MM-DD-YYYY format**

bash

Copy code

sed -n '/\b\(0[1-9]\|1[0-2]\|[1-9]\)[-/]\(0[1-9]\|[12][0-9]\|3[01]\|[1-9]\)[-/][12][0-9]\{3\}\b/p' g.txt

**Explanation:** This command captures dates in MM-DD-YYYY or MM/DD/YYYY formats with valid ranges for months (1–12), days (1–31), and years starting with 1 or 2, using - or / as delimiters.

**Expected Output:**

makefile

Copy code

Dates: 12-31-2023, 01/01/2020.

**20. Print all lines of a file that are less than 10 characters in length**

bash

Copy code

sed -n '/^.\{1,9\}$/p' g.txt

**Explanation:** This command prints lines containing 1 to 9 characters using the range {1,9}.

**Expected Output:**

arduino

Copy code

Short line.

Another short line.

Let me know if you'd like the next set of commands!

4o

**You said:**

next 10

**ChatGPT said:**

ChatGPT

Here are the next 10 commands (21-30), with explanations and expected outputs for g.txt.

**21. Replace all occurrences in a file of "the" with "a" and "The" with "A"**

bash

Copy code

sed 's/\bthe\b/a/g; s/\bThe\b/A/g' g.txt

**Explanation:** This command replaces lowercase "the" with "a" and capitalized "The" with "A" using word boundaries (\b) to ensure whole words are matched.

**Expected Output:**

css

Copy code

A cat and a mouse played in a field.

**22. Substitute the word "button" for "tree", only if "tree" occurs at the end of a line**

bash

Copy code

sed 's/tree$/button/' g.txt

**Explanation:** This command matches "tree" only at the end of a line ($) and replaces it with "button".

**Expected Output:**

arduino

Copy code

There is a button at the end of the line.

**23. Substitute the word "foible" for the word "tree" but not the word "trees"**

bash

Copy code

sed 's/\btree\b/foible/g' g.txt

**Explanation:** Using word boundaries (\b), this command matches only the standalone word "tree" and replaces it with "foible", leaving "trees" unaffected.

**Expected Output:**

Copy code

The foible stands tall among trees.

**24. Change a text so that every period at the end of a sentence is changed to an exclamation point (!) and every question mark is replaced with an ellipsis (...)**

bash

Copy code

sed 's/\.\s\*$/!/g; s/?/.../g' g.txt

**Explanation:** This command matches periods (\.) at the end of sentences (allowing for spaces) and changes them to exclamation marks. It also replaces any question mark with ellipses.

**Expected Output:**

sql

Copy code

Hello! Are you there...

**25. Change every occurrence of the word 'me' with 'you' but only at the end of a line**

bash

Copy code

sed 's/me$/you/' g.txt

**Explanation:** This command matches "me" only at the end of a line ($) and replaces it with "you".

**Expected Output:**

css

Copy code

It's all about you

**26. Change every occurrence of a three-letter month abbreviation to the appropriate number (e.g., "Jan" to "1")**

bash

Copy code

sed -e 's/\bJan\b/1/g' -e 's/\bFeb\b/2/g' -e 's/\bMar\b/3/g' -e 's/\bApr\b/4/g' -e 's/\bMay\b/5/g' -e 's/\bJun\b/6/g' -e 's/\bJul\b/7/g' -e 's/\bAug\b/8/g' -e 's/\bSep\b/9/g' -e 's/\bOct\b/10/g' -e 's/\bNov\b/11/g' -e 's/\bDec\b/12/g' g.txt

**Explanation:** This command replaces month abbreviations with their numeric equivalents using multiple -e options to apply each replacement.

**Expected Output:**

csharp

Copy code

The meeting is scheduled for 3 25.

**27. Match lines containing variations like (i) Jefferies, jeffery, jeffeys (ii) hitchen, hitchin, hitching (iii) Heard, herd, Hird (iv) dix, dicks, dickson, dixon (v) Mcgee, mcghee, magee**

bash

Copy code

sed -n '/\b\(Jefferies\|jeffery\|jeffeys\|hitchen\|hitchin\|hitching\|Heard\|herd\|Hird\|dix\|dicks\|dickson\|dixon\|Mcgee\|mcghee\|magee\)\b/p' g.txt

**Explanation:** Using alternation (\|) inside \( \) groups, this command searches for any listed variations of names.

**Expected Output:**

csharp

Copy code

Jefferies works with Mcgee.

**28. Replace all multiple spaces in a string by only one space**

bash

Copy code

sed 's/ \*/ /g' g.txt

**Explanation:** This command replaces two or more spaces ( \*) with a single space.

**Expected Output:**

sql

Copy code

This is a line with only one space between words.

**29. Transform C++ style // comments to /\* ... \*/ enclosed comments**

bash

Copy code

sed 's|//\(.\*\)|/\* \1 \*/|' g.txt

**Explanation:** This command matches // followed by any characters (\(.\*\)) and replaces it with /\* ... \*/ style comments.

**Expected Output:**

arduino

Copy code

/\* This is a comment \*/

**30. Print alternative lines (line 1, 3, 5, ...)**

bash

Copy code

sed -n 'p;n' g.txt

**Explanation:** The p;n command prints the current line and skips the next (n), effectively selecting alternate lines.

**Expected Output:**

arduino

Copy code

Line 1: The first line.

Line 3: The third line.

...

Let me know if you'd like the next set of commands!

4o

**You said:**

next 10

**ChatGPT said:**

ChatGPT

Here are the next 10 commands (31-40) with explanations and expected outputs for g.txt.

**31. Change the letters "dog" to "HORSE" everywhere it occurs on all lines**

bash

Copy code

sed 's/dog/HORSE/g' g.txt

**Explanation:** This command replaces every occurrence of "dog" with "HORSE" globally (g) on each line.

**Expected Output:**

bash

Copy code

The HORSE chased the cat.

**32. Change all occurrences of "Man" at the beginning of a line to "Person"**

bash

Copy code

sed 's/^Man/Person/' g.txt

**Explanation:** This command matches "Man" only at the beginning of a line (^) and replaces it with "Person".

**Expected Output:**

css

Copy code

Person went to the store.

**33. Change all occurrences of "stick" followed by any punctuation at the end of a line to "Stick." (replacing punctuation with a period)**

bash

Copy code

sed 's/stick[[:punct:]]$/Stick./' g.txt

**Explanation:** This command matches "stick" followed by any punctuation ([[:punct:]]) at the end of a line ($) and replaces it with "Stick.".

**Expected Output:**

css

Copy code

He picked up a Stick.

**34. Change all occurrences of "Dog" or "dog" to "COW"**

bash

Copy code

sed 's/[Dd]og/COW/g' g.txt

**Explanation:** The [Dd]og pattern matches both "Dog" and "dog", and g ensures all occurrences are replaced with "COW".

**Expected Output:**

bash

Copy code

The COW chased the cat. A COW ran by.

**35. Change all Canadian or American spellings of "colour" (color) to "Color"**

bash

Copy code

sed 's/\bcolour\b/Color/g; s/\bcolor\b/Color/g' g.txt

**Explanation:** This command replaces both "colour" and "color" with "Color", regardless of Canadian or American spelling.

**Expected Output:**

css

Copy code

The wall was painted in a bright Color.

**36. Double all vowels in every word on every line**

bash

Copy code

sed 's/[aeiouAEIOU]/&&/g' g.txt

**Explanation:** This command matches each vowel ([aeiouAEIOU]) and uses && to repeat it, effectively doubling each vowel.

**Expected Output:**

css

Copy code

I looovee the aaesthetic oof thiss.

**37. Triple the amount of space between every word**

bash

Copy code

sed 's/ / /g' g.txt

**Explanation:** This command replaces each single space ( ) with three spaces, effectively tripling the gap between words.

**Expected Output:**

bash

Copy code

This is a test sentence.

**38. Find and print lines that contain "dog" followed by any number of digits, then "cat"**

bash

Copy code

sed -n '/dog[0-9]\*cat/p' g.txt

**Explanation:** This command searches for "dog" followed by zero or more digits ([0-9]\*), then "cat", and prints (p) the matching lines.

**Expected Output:**

csharp

Copy code

The dog123cat is here.

**39. Find and print lines that contain the letters "dog" followed anywhere by the letters "cat"**

bash

Copy code

sed -n '/dog.\*cat/p' g.txt

**Explanation:** This command matches "dog" followed by any characters (.\*), then "cat", and prints the matching lines.

**Expected Output:**

css

Copy code

A dog walked past a cat.

**40. Change all occurrences of one or more digits to the single word "NUMBER"**

bash

Copy code

sed 's/[0-9]\+/NUMBER/g' g.txt

**Explanation:** This command replaces one or more digits ([0-9]\+) with the word "NUMBER".

**Expected Output:**

css

Copy code

I have NUMBER apples.

Let me know if you'd like the next set!

4o

**You said:**

next 10

**ChatGPT said:**

ChatGPT

Here are commands 41-50 with explanations and expected outputs for g.txt.

**41. Replace all occurrences of one or more blanks with a single blank**

bash

Copy code

sed 's/ \+/ /g' g.txt

**Explanation:** This command replaces multiple spaces ( \+) with a single space, effectively condensing extra spaces.

**Expected Output:**

bash

Copy code

This is a test sentence.

**42. Replace all occurrences of one or more tabs or blanks with a single blank**

bash

Copy code

sed 's/[ \t]\+/ /g' g.txt

**Explanation:** This command matches one or more spaces or tabs ([ \t]\+) and replaces them with a single space.

**Expected Output:**

bash

Copy code

This is a test sentence.

**43. Remove the first 8 characters from every line**

bash

Copy code

sed 's/^.\{8\}//' g.txt

**Explanation:** This command matches the first 8 characters at the beginning of each line (^.\{8\}) and removes them.

**Expected Output (if original line is "This is a line to modify"):**

css

Copy code

a line to modify

**44. Remove all leading blanks or tabs from all lines**

bash

Copy code

sed 's/^[ \t]\*//' g.txt

**Explanation:** This command matches all leading spaces or tabs (^[ \t]\*) and removes them.

**Expected Output:**

scss

Copy code

This line has no leading whitespace.

**45. Remove all trailing blanks or tabs from all lines**

bash

Copy code

sed 's/[ \t]\*$//' g.txt

**Explanation:** This command matches all trailing spaces or tabs ([ \t]\*$) at the end of each line and removes them.

**Expected Output:**

scss

Copy code

Line without trailing whitespace.

**46. Replace all tab characters with eight spaces**

bash

Copy code

sed 's/\t/ /g' g.txt

**Explanation:** This command replaces each tab character (\t) with eight spaces.

**Expected Output:**

arduino

Copy code

This line has tabs replaced.

**47. Change all punctuation so that the sentence period lies outside of the closing double quote**

bash

Copy code

sed 's/\("\)\./\1/' g.txt

**Explanation:** This command finds periods directly before the closing double quote and removes them from that position, so the period appears outside the quotes.

**Expected Output:**

arduino

Copy code

"Hello there".

**48. Remove everything leading up to and including the last blank on each line**

bash

Copy code

sed 's/.\* //g' g.txt

**Explanation:** This command removes all content up to and including the last space on each line.

**Expected Output (if the original line is "Keep only last part"):**

Copy code

part

**49. Remove everything including and after the first blank on each line**

bash

Copy code

sed 's/ .\*$//' g.txt

**Explanation:** This command matches the first space and everything after it on each line and removes it.

**Expected Output (if the original line is "Keep only first part"):**

Copy code

Keep

**50. Put double quotes around every occurrence of the phrase "user-friendly"**

bash

Copy code

sed 's/user-friendly/"&"/g' g.txt

**Explanation:** The & symbol represents the matched text, so this command wraps each occurrence of "user-friendly" with double quotes.

**Expected Output:**

csharp

Copy code

This tool is "user-friendly".

Let me know if you'd like the next set of commands!

4o

**You said:**

next 10

**ChatGPT said:**

ChatGPT

Here are commands 51-60 with explanations and expected outputs for g.txt.

**51. Add an extra blank after every period at the end of a sentence**

bash

Copy code

sed 's/\.\s\*/. /g' g.txt

**Explanation:** This command matches periods followed by any amount of whitespace and replaces them with a period followed by two spaces.

**Expected Output:**

csharp

Copy code

This is a sentence. Here is another sentence.

**52. Make sure that every period at the end of a sentence is followed by exactly two blanks**

bash

Copy code

sed 's/\.\s\*/. /g' g.txt

**Explanation:** This command ensures each period at the end of a sentence is followed by exactly two spaces, replacing any whitespace that might follow a period.

**Expected Output:**

css

Copy code

It was a sunny day. The sky was clear.

**53. Truncate every line to ten characters**

bash

Copy code

sed 's/^\(.\{10\}\).\*/\1/' g.txt

**Explanation:** This command matches the first 10 characters of each line (\(.\{10\}\)) and removes the rest.

**Expected Output (if original line is "Truncate this line"):**

sql

Copy code

Truncate t

**54. Exchange the first 10 characters with the next 15 characters on every line**

bash

Copy code

sed 's/^\(.\{10\}\)\(.\{15\}\)/\2\1/' g.txt

**Explanation:** This command captures the first 10 characters as \1 and the next 15 characters as \2, then swaps them.

**Expected Output (if original line is "1234567890abcdefghijklmno..."):**

Copy code

abcdefghijklmno1234567890...

**55. Exchange the first number with the second number on every line**

bash

Copy code

sed -E 's/([0-9]+)([^0-9]\*)([0-9]+)/\3\2\1/' g.txt

**Explanation:** This command captures the first number as \1, any characters in between as \2, and the second number as \3, then swaps the first and second numbers.

**Expected Output (if the original line is "12 apples 34 oranges"):**

Copy code

34 apples 12 oranges

**56. Remove all leading zeroes from the first number on each line**

bash

Copy code

sed 's/\b0\*\([1-9][0-9]\*\)/\1/' g.txt

**Explanation:** This command matches leading zeroes in the first number, ensuring numbers like "00012" become "12".

**Expected Output:**

Copy code

He has 12 apples, not 00034.

**57. Find and print lines that contain all the vowels in alphabetical order (a, e, i, o, u)**

bash

Copy code

sed -n '/a.\*e.\*i.\*o.\*u/p' g.txt

**Explanation:** This command checks if each line contains the vowels "a", "e", "i", "o", and "u" in that order (even if separated by other characters) and prints those lines.

**Expected Output:**

scss

Copy code

This line contains all vowels in alphabetical order.

**58. Find and print lines that contain all the vowels in any order**

bash

Copy code

sed -n '/[aA].\*[eE].\*[iI].\*[oO].\*[uU]/p' g.txt

**Explanation:** This command finds lines that contain all vowels at least once in any order, regardless of case.

**Expected Output:**

python

Copy code

An example line with all vowels.

**59. Change all occurrences of one or more digits surrounded by spaces to the word "NUMBER" also surrounded by spaces**

bash

Copy code

sed 's/ [0-9]\+ / NUMBER /g' g.txt

**Explanation:** This command replaces sequences of one or more digits that are surrounded by spaces with the word "NUMBER".

**Expected Output:**

css

Copy code

I have NUMBER apples and NUMBER oranges.

**60. Change only the second occurrence of a single blank to a colon in each line**

bash

Copy code

sed 's/ /:/2' g.txt

**Explanation:** This command matches the second occurrence of a single space on each line and replaces it with a colon.

**Expected Output (if the line is "First Second Third Fourth"):**

sql

Copy code

First Second:Third Fourth

Let me know if you'd like the next set!

4o

**You said:**

give next 10

**ChatGPT said:**

ChatGPT

Here are commands 61-70 with explanations and expected outputs for g.txt.

**61. Change the second-to-last single blank to a colon in each line**

bash

Copy code

sed -E ':a; s/ (.\* )/\1:/; ta' g.txt

**Explanation:** This command identifies the second-to-last space in each line and replaces it with a colon. The ta loop ensures it iterates to the second-to-last occurrence.

**Expected Output (if the line is "This is a simple example text"):**

vbnet

Copy code

This is a simple example:text

**62. Change only the second occurrence of a string of one or more blanks to a colon in each line**

bash

Copy code

sed 's/ \*/:/2' g.txt

**Explanation:** This command matches the second occurrence of one or more spaces on each line and replaces it with a colon.

**Expected Output (if the line is "First Second Third Fourth"):**

sql

Copy code

First Second:Third Fourth

**63. Change only the second-to-last occurrence of a string of one or more blanks to a colon in each line**

bash

Copy code

sed -E ':a; s/ \*(.\* \*)/\1:/; ta' g.txt

**Explanation:** This command locates the second-to-last sequence of one or more spaces in each line and replaces it with a colon using a looping construct.

**Expected Output (if the line is "First Second Third Fourth"):**

sql

Copy code

First Second Third:Fourth

**64. Remove all occurrences of HTML tags on the same line**

bash

Copy code

sed 's/<[^>]\*>//g' g.txt

**Explanation:** This command matches HTML tags by identifying patterns within angle brackets (<[^>]\*>) and removes them.

**Expected Output (if the line is "<p>This is a paragraph</p>"):**

csharp

Copy code

This is a paragraph

**65. Remove everything between double quotes, leaving only the quotes**

bash

Copy code

sed 's/"[^"]\*"/""/g' g.txt

**Explanation:** This command matches any text between double quotes and replaces it with empty quotes (""), preserving the quotes.

**Expected Output (if the line is "Here is a "quoted" text example"):**

vbnet

Copy code

Here is a "" text example

**66. Find lines that contain only one single quote character (an unmatched quote)**

bash

Copy code

sed -n '/^[^\']\*\('[^\']\*\)$/p' g.txt

**Explanation:** This command finds lines with only one single quote by ensuring no other single quotes appear on the line.

**Expected Output (if the line is "This is a 'test line"):**

vbnet

Copy code

This is a 'test line

**67. Put double quotes around every occurrence of "user-friendly" unless it already has double quotes**

bash

Copy code

sed 's/\([^"]\)\(user-friendly\)\([^"]\)/\1"\2"\3/g' g.txt

**Explanation:** This command looks for "user-friendly" when it does not already have quotes around it and then wraps it in quotes.

**Expected Output (if the line is "This tool is user-friendly."):**

csharp

Copy code

This tool is "user-friendly".

**68. Convert "$number" to "numberCDN" and do the reverse**

bash

Copy code

# Convert $123.45 to 123.45CDN

sed 's/\$\([0-9]\*\.[0-9]\*\)/\1CDN/g' g.txt

# Reverse: convert 123.45CDN to $123.45

sed 's/\([0-9]\*\.[0-9]\*\)CDN/\$\1/g' g.txt

**Explanation:** The first command removes $ and appends CDN. The second command reverses it by adding $ and removing CDN.

**Expected Output:**

yaml

Copy code

Original: $123.45 -> Converted: 123.45CDN

Reverse: 123.45CDN -> Converted: $123.45

**69. Replace periods in decimal numbers with commas, then reverse it**

bash

Copy code

# Convert decimal points to commas

sed 's/\([0-9]\+\)\.\([0-9]\+\)/\1,\2/g' g.txt

# Reverse: Convert commas back to decimal points

sed 's/\([0-9]\+\),\([0-9]\+\)/\1.\2/g' g.txt

**Explanation:** This first command changes decimal points in numbers to commas, and the second one reverses the change.

**Expected Output:**

yaml

Copy code

Original: 123.45 -> Converted: 123,45

Reverse: 123,45 -> Converted: 123.45

**70. Replace commas in numbers with spaces, then reverse it**

bash

Copy code

# Replace commas separating groups of three digits with spaces

sed 's/\([0-9]\),\([0-9]\{3\}\)/\1 \2/g' g.txt

# Reverse: Convert spaces back to commas

sed 's/\([0-9]\) \([0-9]\{3\}\)/\1,\2/g' g.txt

**Explanation:** The first command replaces commas in thousand-separated numbers with spaces. The second command restores the commas.

**Expected Output:**

yaml

Copy code

Original: 1,234,567.23 -> Converted: 1 234 567.23

Reverse: 1 234 567.23 -> Converted: 1,234,567.23

Let me know if you'd like the next set of commands!

4o

**You said:**

next 10

**ChatGPT said:**

ChatGPT

Here are commands 71-80 with explanations and expected outputs for g.txt.

**71. Fix common misspellings of "@college.com"**

bash

Copy code

sed 's/@college[.]com/@college.com/g; s/@collage[.]com/@college.com/g' g.txt

**Explanation:** This command corrects common typos such as “@collage.com” or “@college[.]com” to “@college.com”.

**Expected Output (if the line is "student@collage.com"):**

css

Copy code

student@college.com

**72. Remove quotes from around numbers, then add quotes around numbers**

bash

Copy code

# Remove single or double quotes around numbers

sed "s/['\"]\([0-9]\+\)['\"]/\\1/g" g.txt

# Add double quotes around numbers

sed "s/\b\([0-9]\+\)\b/\"\1\"/g" g.txt

**Explanation:** The first command removes any quotes around numbers, and the second command adds double quotes around numbers.

**Expected Output (if the line is "The value is '123' or "456""):**

csharp

Copy code

Remove quotes: The value is 123 or 456

Add quotes: The value is "123" or "456"

**73. Add (HEX:) prefix to hexadecimal numbers, then remove it**

bash

Copy code

# Prefix hexadecimal numbers with (HEX:)

sed 's/\(0x[A-Fa-f0-9]\+\)/(HEX:)\1/g' g.txt

# Remove (HEX:) prefix

sed 's/(HEX:)0x/0x/g' g.txt

**Explanation:** The first command adds "(HEX:)" before hexadecimal numbers, and the second command removes it.

**Expected Output (if the line is "Memory address is 0xA0FF2375C3"):**

swift

Copy code

Add prefix: Memory address is (HEX:)0xA0FF2375C3

Remove prefix: Memory address is 0xA0FF2375C3

**74. Change "dog" to "dog-eat-dog" and "cat" to "cat-eat-cat", then reverse it**

bash

Copy code

# Convert "dog" to "dog-eat-dog" and "cat" to "cat-eat-cat"

sed 's/\bdog\b/dog-eat-dog/g; s/\bcat\b/cat-eat-cat/g' g.txt

# Reverse: Convert "dog-eat-dog" back to "dog" and "cat-eat-cat" back to "cat"

sed 's/dog-eat-dog/dog/g; s/cat-eat-cat/cat/g' g.txt

**Explanation:** The first command transforms "dog" and "cat" to their respective phrases, and the second command reverses these changes.

**Expected Output (if the line is "The dog chases the cat"):**

mathematica

Copy code

After conversion: The dog-eat-dog chases the cat-eat-cat

After reversal: The dog chases the cat

**75. Produce a list of email addresses and home pages for users**

bash

Copy code

sed -n 's/.\*\(\b[A-Za-z0-9.\_%+-]\+@[A-Za-z0-9.-]\+\.[A-Za-z]\{2,}\b\).\*/\1/p' g.txt

**Explanation:** This command extracts email addresses from each line and lists them.

**Expected Output (if the line is "Contact us at info@college.com or visit college.com"):**

css

Copy code

info@college.com

**76. Run a script to replace "dog" with "cat" across multiple files**

bash

Copy code

# Assuming `substitute.sh` contains:

# #!/bin/bash

# sed -i "$1" "$@"

# Run with:

./substitute.sh 's/dog/cat/g' g.txt

**Explanation:** This script accepts a sed substitution command and applies it to all specified files.

**Expected Output (if the line is "A dog barks"):**

bash

Copy code

A cat barks

**77. Ensure each sentence starts on a new line**

bash

Copy code

sed 's/[.!?] \*/&\n/g' g.txt

**Explanation:** This command identifies sentence-ending punctuation followed by spaces, then replaces spaces with a newline.

**Expected Output (if the line is "This is a test. And another! Now a question?"):**

css

Copy code

This is a test.

And another!

Now a question?

**78. Print lines with vowels in alphabetical order**

bash

Copy code

sed -n '/a.\*e.\*i.\*o.\*u/p' /usr/dict/words

**Explanation:** This command finds words in the dictionary that contain all vowels in alphabetical order.

**Expected Output (examples):**

Copy code

abstemious

facetious

**79. Change the second and subsequent occurrences of spaces to single spaces**

bash

Copy code

sed 's/ \*/ /2g' g.txt

**Explanation:** This command leaves the first occurrence of multiple spaces but reduces all others to single spaces.

**Expected Output (if the line is "This is a test"):**

bash

Copy code

This is a test

**80. Change every second space sequence to a colon for columnar data**

bash

Copy code

sed 's/ \*/:/2g' g.txt

**Explanation:** This command converts every second sequence of spaces in columnar data into a colon for readability.

**Expected Output (if the line is "12 34 56 78"):**

Copy code

12 34:56 78