

## Problem 1 source code:

The screenshot shows a code editor interface with the title bar "SAT4650". The left sidebar is titled "EXPLORER" and shows a file tree for "SAT4650" with "docs", "env", "submissions\Labs\Lab1", "submissions\Labs\Lab2", and files "test.py" and "health\_data\_analysis.py". The right pane displays the content of "health\_data\_analysis.py". The code is as follows:

```
6 )
7 # 3. Content stored in variable named 'raw_text'
8 raw_text = fhandle.read()
9 fhandle.close()
10 # 4. Text converted to lowercase
11 raw_text = raw_text.lower()
12
13 # 5. New string variable creation
14 clean_text = " "
15
16 raw_text = (raw_text.replace('.', ' ')
17 .replace(',', ' ')
18 .replace('!', ' ')
19 .replace('?', ' ')
20 .replace('"', ' ')
21 .replace("'", ' ')
22 .replace(';', ' ')
23 .replace(':', ' ')
24 .replace('(', ' ')
25 .replace(')', ' ')
26 .replace('-', ' ')
27 .replace('_', ' '))
28
29 # 7. Replacing line breaks with spaces
30 raw_text = raw_text.replace('\n', ' ')
31
32 # 8. This splits on ANY whitespace including the spaces we just added from 7.
33 words = raw_text.split()
34 # And joins back with single spaces
35 raw_text = ' '.join(words)
36 # 5: Assigning the raw_text content to the clean_text variable
37 clean_text = raw_text
38 # 9: Printing the final cleaned version (clean_text)
39 print(clean_text)
```

## Problem 1 output:

The screenshot shows a code editor interface with the title bar "SAT4650". The left sidebar is titled "EXPLORER" and shows a file tree for "SAT4650" with "docs", "env", "submissions\Labs\Lab1", "submissions\Labs\Lab2", and files "test.py" and "health\_data\_analysis.py". The right pane has tabs for "PROBLEMS", "OUTPUT", "DEBUG CONSOLE", "TERMINAL", and "PORTS". The "TERMINAL" tab is active, showing the command line output:

```
(C:\Users\Jones\Documents\Data Science\SAT4650)
Jones@DESKTOP-QPUCUQ9 MINGW64 ~/Documents/Data Science/SAT4650
$ "C:/Users/Jones/Documents/Data Science/SAT4650/env/python.exe" "c:/Users/Jones/Documents/Data Science/SAT4650/submissions/Labs/Lab2/health_data_analysis.py"
patients are advised to take their medication as prescribed by their healthcare provider if a patient experiences fever fever or persistent fatigue they should contact a clinician immediately regular exercise and a balanced diet help improve overall health and recovery some patients report pain headaches or dizziness after missing their medication doses healthcare workers recommend drinking enough water during illness to prevent dehydration followup appointments are important to monitor symptoms and adjust treatment plans patients with chronic conditions should keep track of their daily symptoms carefully pain and fatigue are common ignoring early signs of infection may lead to complications pain and longer recovery time healthcare teams encourage patients to rest and avoid strenuous activities when sick poor communication between patients and providers can delay treatment and increase fatigue timely treatment symptom awareness and early response to fever play a key role in preventing severe illness
(C:\Users\Jones\Documents\Data Science\SAT4650)
Jones@DESKTOP-QPUCUQ9 MINGW64 ~/Documents/Data Science/SAT4650
$
```

## Problem 2 source code:

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "SAT4650".
- Editor:** Displays the Python script `health_data_analysis.py` with the following code:

```
45 # 1. Splitting the text to have individual words
46 words = clean_text.split()
47
48 # 2. Creating dictionary to count word appearances
49 word_counts = {}
50 for word in words:
51     if word in word_counts:
52         # Increase count if the word already exists
53         word_counts[word] += 1
54     else:
55         # Else start the count at 1 if it's a new word
56         word_counts[word] = 1
57
58 # 3. Converting to a set to remove duplicates
59 unique_words_set = set(words)
60
61 # 4. Final collection of unique words
62 unique_words = list(unique_words_set)
63
64
65 # 5.a) Printing the number of unique words
66 print("Number of unique words:", len(unique_words))
67
68 # 5.b) Printing the frequency of each unique word
69 print("\nFrequency of each word:")
70 for word, count in word_counts.items():
71     print(f"{word}: {count}")
72
73 # 5c. A list of unique words with the first letter capitalized
74 capitalized_words = [word.capitalize() for word in unique_words]
75
76 print("\n Unique words capitalized:")
77 print(capitalized_words)
```

## Code output:

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "SAT4650".
- Terminal:** Displays the output of running the script:

```
54     word_counts[word] += 1
55
56     # Else start the count at 1 if it's a new word
57     word_counts[word] = 1
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
```

Output from the terminal:

```
Frequency of each word:
patients: 5
are: 3
advised: 1
to: 6
take: 1
their: 4
medication: 2
as: 1
prescribed: 1
by: 1
healthcare: 3
provider: 1
if: 1
a: 4
patient: 1
experiences: 1
fever: 3
or: 2
persistent: 1
fatigue: 3
they: 1
should: 2
contact: 1
clinic: 1
```

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a project structure under "SAT4650" with files like "test.py", "health\_data\_analysis.py", and "lab\_project\_2.pdf".
- Terminal:** Displays the output of a Python script named "health\_data\_analysis.py". The script prints word counts from a dataset. The output includes:

```
poor: 1
communication: 1
between: 1
providers: 1
can: 1
delay: 1
increase: 1
timely: 1
symptom: 1
awareness: 1
response: 1
play: 1
key: 1
role: 1
in: 1
preventing: 1
severe: 1

Unique words capitalized:
['Symptom', 'Daily', 'Headaches', 'Contact', 'And', 'Experiences', 'Exercise', 'Diet', 'Should', 'By', 'Timely', 'If', 'Workers', 'After', 'Carefully', 'Track',
'Keep', 'Adjust', 'Recovery', 'Recommend', 'When', 'They', 'Plans', 'Lead', 'Pain', 'Doses', 'Water', 'Improve', 'Advised', 'Enough', 'Or', 'Prescribed', 'Co
mmon', 'Role', 'Activities', 'Play', 'Dizziness', 'Patients', 'Infection', 'Awareness', 'Regular', 'May', 'Between', 'Drinking', 'Balanced', 'Signs', 'Preventi
ng', 'Their', 'Conditions', 'Health', 'Providers', 'Avoid', 'Overall', 'Early', 'Take', 'Treatment', 'Appointments', 'Time', 'Rest', 'Prevent', 'Immediately',
'Healthcare', 'Can', 'Medication', 'Response', 'Increase', 'In', 'Sick', 'Help', 'Patient', 'With', 'Communication', 'As', 'Ignoring', 'Persistent', 'Symptoms',
'During', 'Strenuous', 'Important', 'Fatigue', 'Illness', 'Encourage', 'Fever', 'Missing', 'Of', 'Teams', 'Followup', 'Clinic', 'Key', 'Dehydration', 'Are',
'Some', 'Monitor', 'A', 'Report', 'Delay', 'To', 'Provider', 'Chronic', 'Poor', 'Complications', 'Longer', 'Severe']
```
- Output:** Shows the command used to run the script: \$ "C:/Users/Jones/Documents/Data Science/SAT4650/env/python.exe" "c:/Users/Jones/Documents/Data Science/SAT4650/submissions/Labs/Lab2/health\_data\_analysis.py"