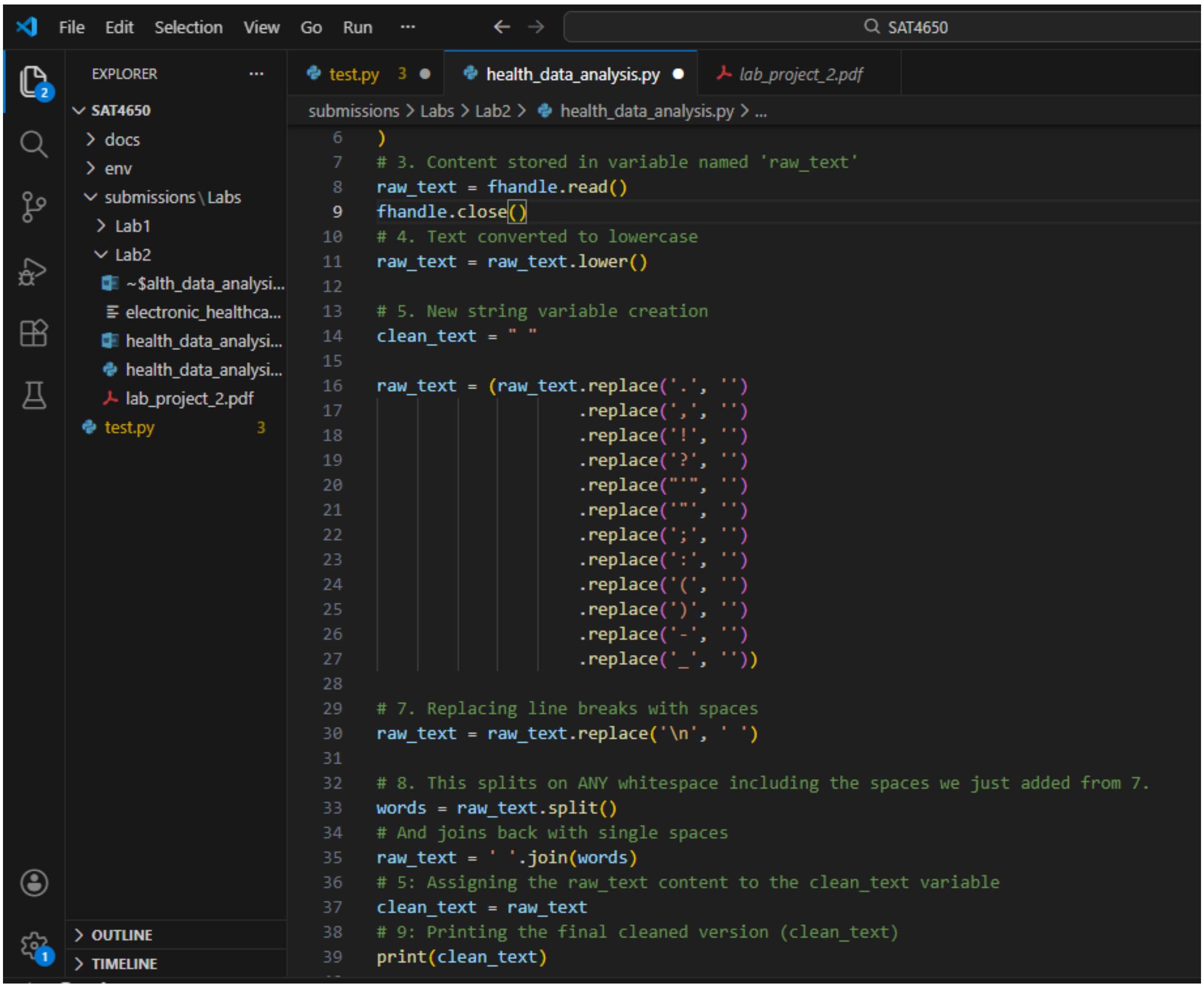
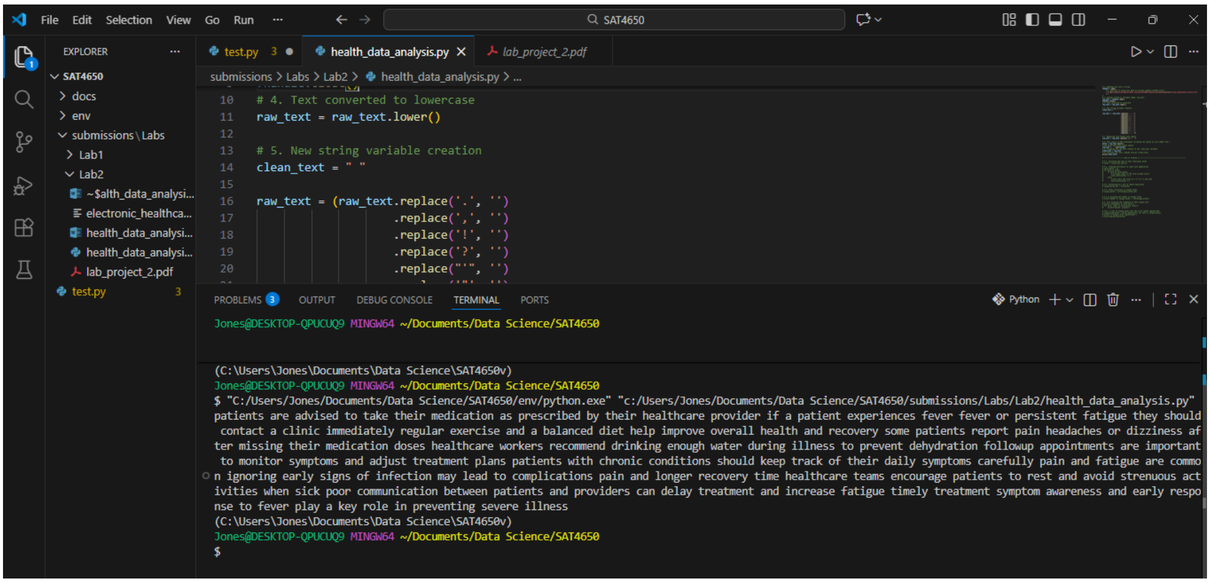
Problem 1 source code:



```python
)
# 3. Content stored in variable named 'raw_text'
raw_text = fhandle.read()
fhandle.close()
# 4. Text converted to lowercase
raw_text = raw_text.lower()

# 5. New string variable creation
clean_text = " "

raw_text = (raw_text.replace('.', '')
                    .replace(',', '')
                    .replace('!', '')
                    .replace('?', '')
                    .replace("'", '')
                    .replace('"', '')
                    .replace(';', '')
                    .replace(':', '')
                    .replace('(', '')
                    .replace(')', '')
                    .replace('-', '')
                    .replace('_', ''))

# 7. Replacing line breaks with spaces
raw_text = raw_text.replace('\n', ' ')

# 8. This splits on ANY whitespace including the spaces we just added from 7.
words = raw_text.split()
# And joins back with single spaces
raw_text = ' '.join(words)
# 5: Assigning the raw_text content to the clean_text variable
clean_text = raw_text
# 9: Printing the final cleaned version (clean_text)
print(clean_text)
```

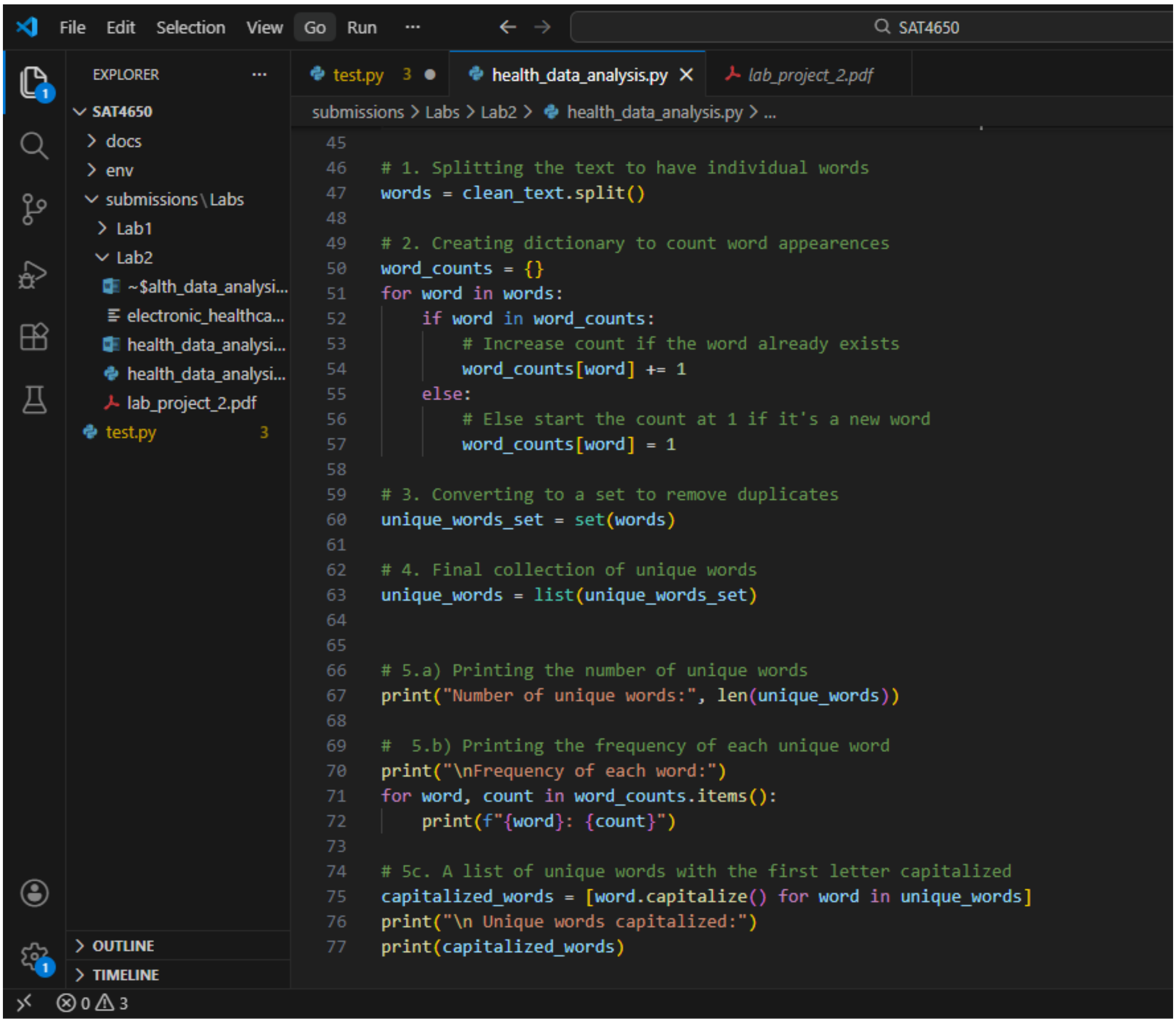
Problem 1 output:



```python
# 4. Text converted to lowercase
raw_text = raw_text.lower()

# 5. New string variable creation
clean_text = " "

raw_text = (raw_text.replace('.', '')
                    .replace(',', '')
                    .replace('!', '')
                    .replace('?', '')
                    .replace("'", '')
```

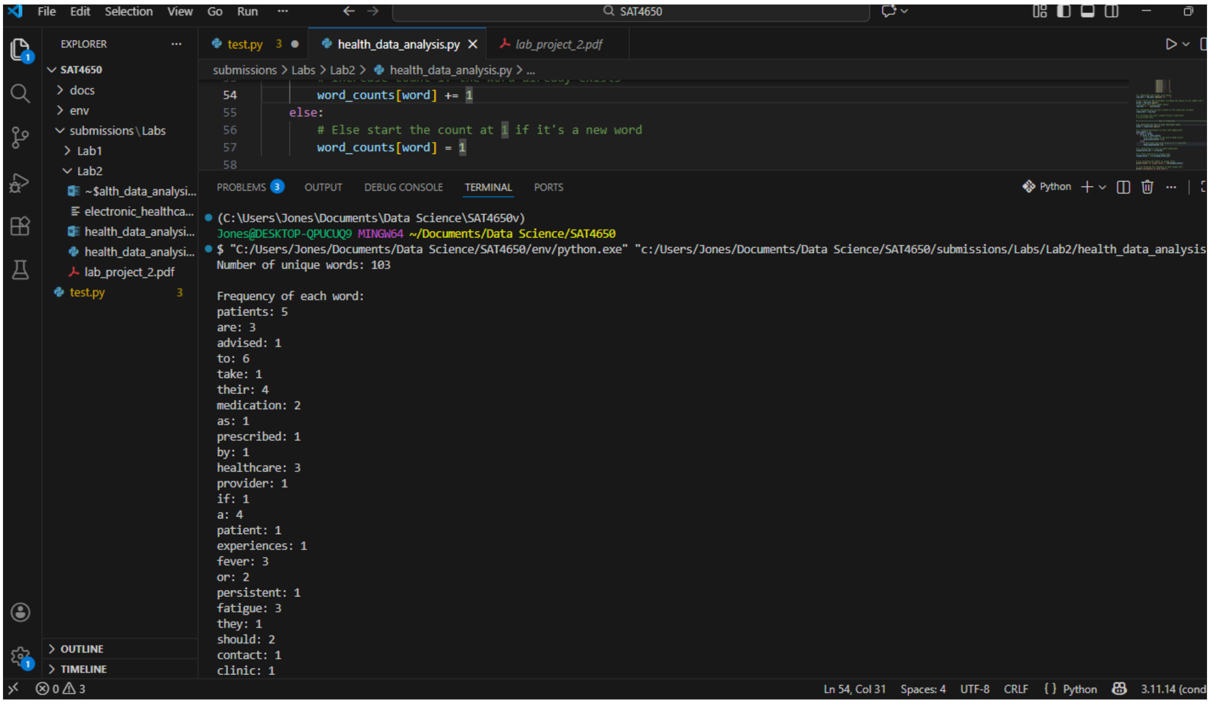
```
Jones@DESKTOP-QPUCUQ9 MINGW64 ~/Documents/Data Science/SAT4650


(C:\Users\Jones\Documents\Data Science\SAT4650v)
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 clinic immediately regular exercise and a balanced diet help improve overall health and recovery some patients report pain headaches or dizziness af
ter 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 commo
n ignoring early signs of infection may lead to complications pain and longer recovery time healthcare teams encourage patients to rest and avoid strenuous act
ivities when sick poor communication between patients and providers can delay treatment and increase fatigue timely treatment symptom awareness and early respo
nse to fever play a key role in preventing severe illness
(C:\Users\Jones\Documents\Data Science\SAT4650v)
Jones@DESKTOP-QPUCUQ9 MINGW64 ~/Documents/Data Science/SAT4650
$
```

Problem 2 source code:



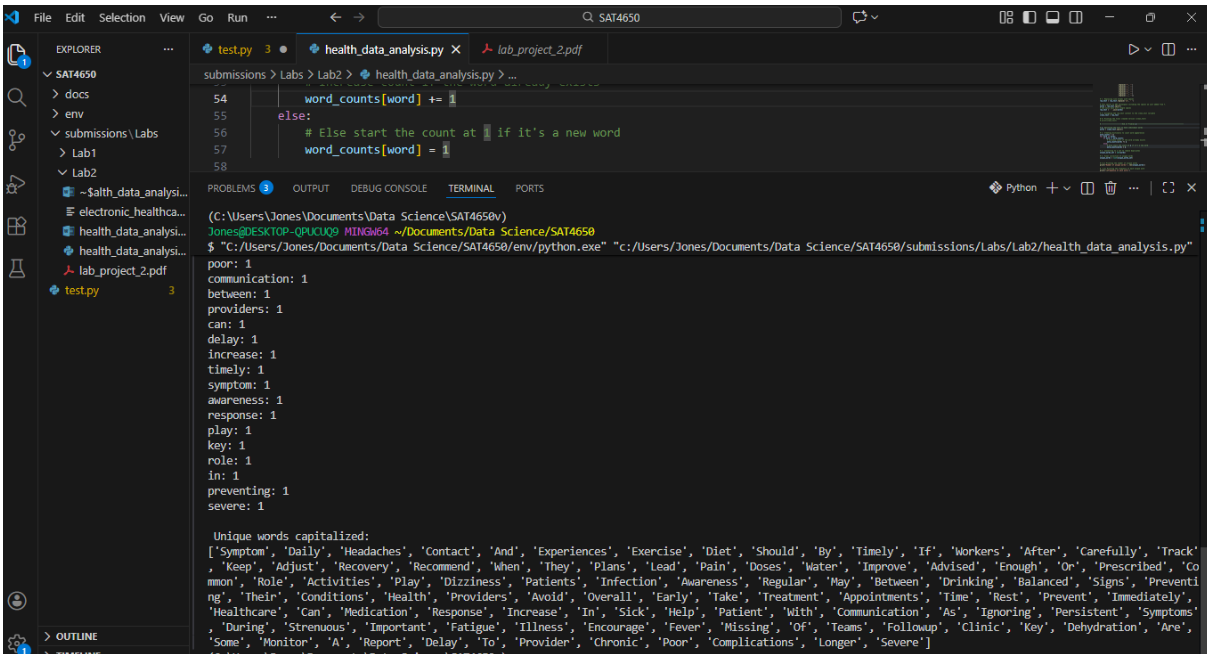
```python

# 1. Splitting the text to have individual words
words = clean_text.split()

# 2. Creating dictionary to count word appearences
word_counts = {}
for word in words:
    if word in word_counts:
        # Increase count if the word already exists
        word_counts[word] += 1
    else:
        # Else start the count at 1 if it's a new word
        word_counts[word] = 1

# 3. Converting to a set to remove duplicates
unique_words_set = set(words)

# 4. Final collection of unique words
unique_words = list(unique_words_set)


# 5.a) Printing the number of unique words
print("Number of unique words:", len(unique_words))

#  5.b) Printing the frequency of each unique word
print("\nFrequency of each word:")
for word, count in word_counts.items():
    print(f"{word}: {count}")

# 5c. A list of unique words with the first letter capitalized
capitalized_words = [word.capitalize() for word in unique_words]
print("\n Unique words capitalized:")
print(capitalized_words)
```

Code output:



```python
        word_counts[word] += 1
    else:
        # Else start the count at 1 if it's a new word
        word_counts[word] = 1

```

```
(C:\Users\Jones\Documents\Data Science\SAT4650v)
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
Number of unique words: 103

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
```



```
        word_counts[word] += 1
    else:
        # Else start the count at 1 if it's a new word
        word_counts[word] = 1
```

```
(C:\Users\Jones\Documents\Data Science\SAT4650v)
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"
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']
```