MIT | Academy of Engineering

(An Autonomous Institute Affiliated to Savitribai Phule Pune University)

Theory Activity No. 1

Name : Rohan Ajay Burande

PRN : 202401040188

Div : CS1

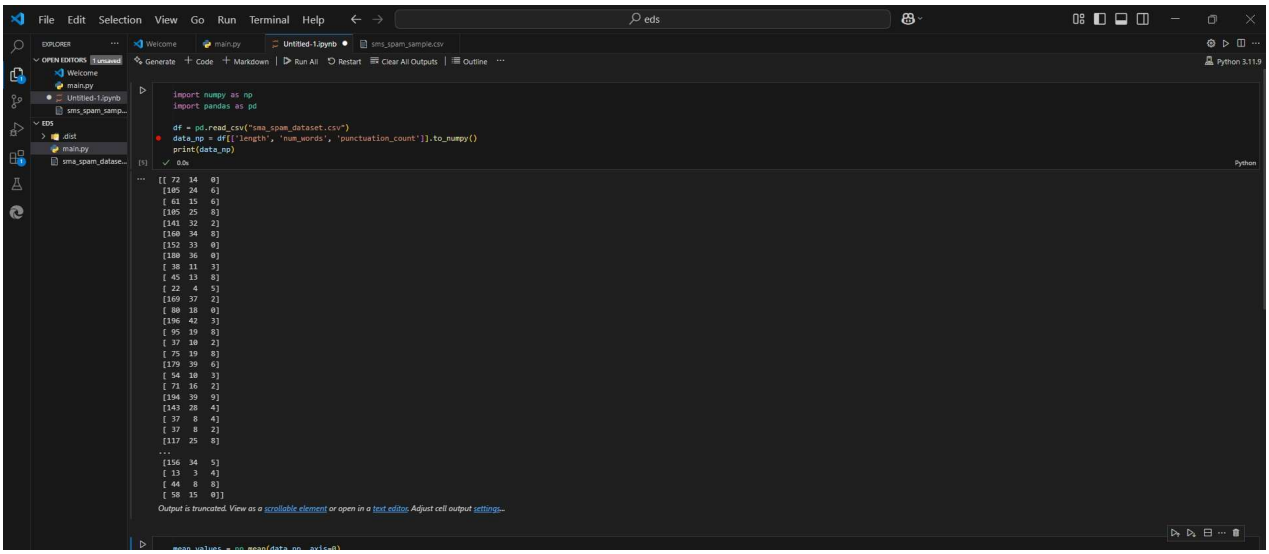
Roll No : CS1-78

Data Set : SMS SPAM COLLECTION

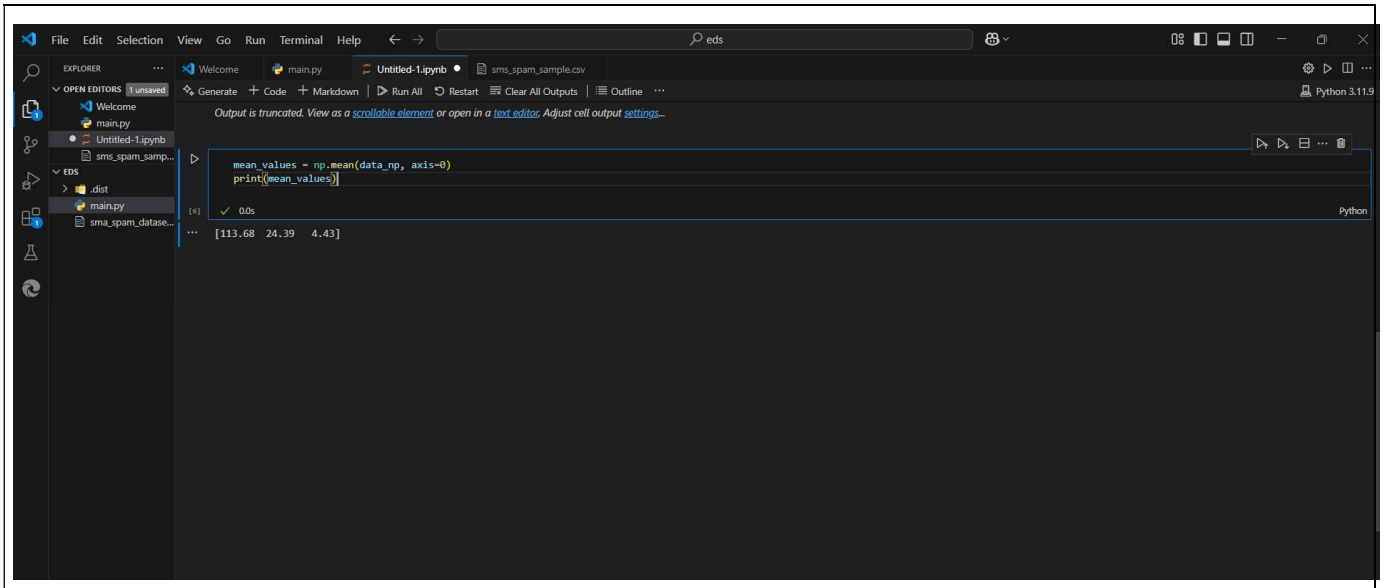
NumPy:-

1. Convert DataFrame columns to NumPy array

solution:

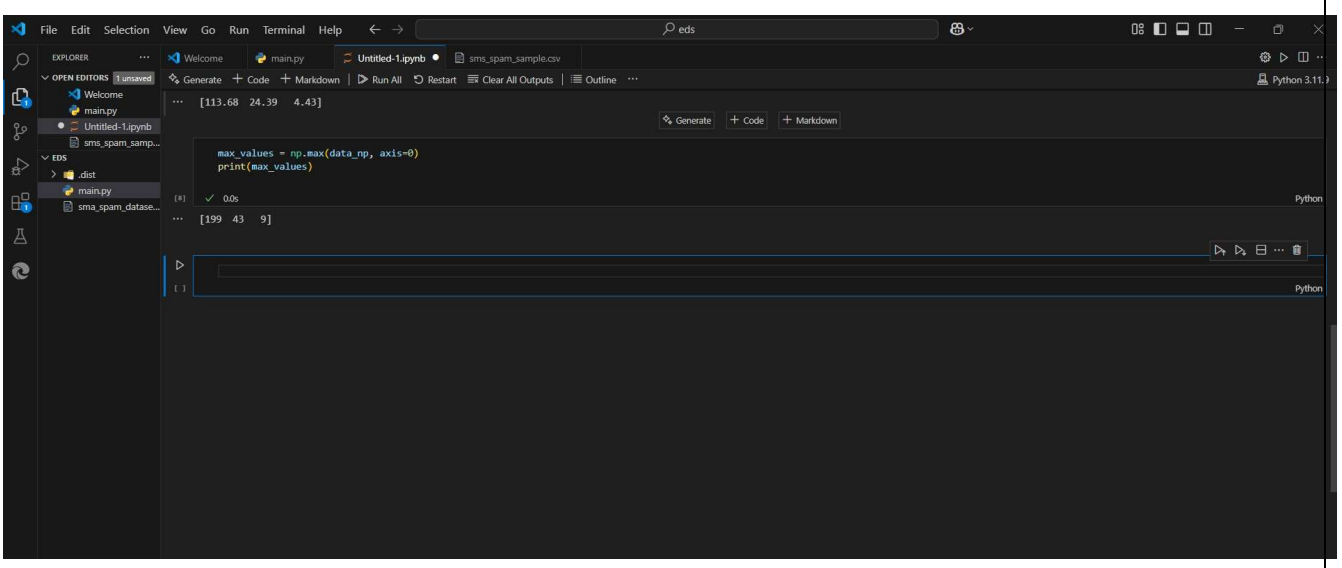


2. Calculate mean of each numeric column



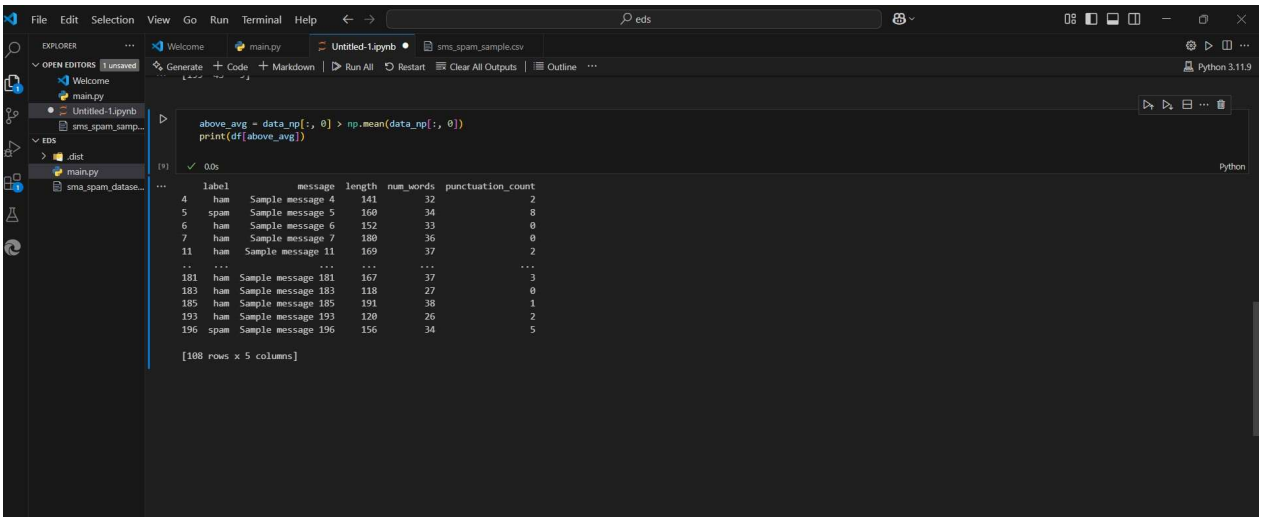
3. Find max values in each column

solution:



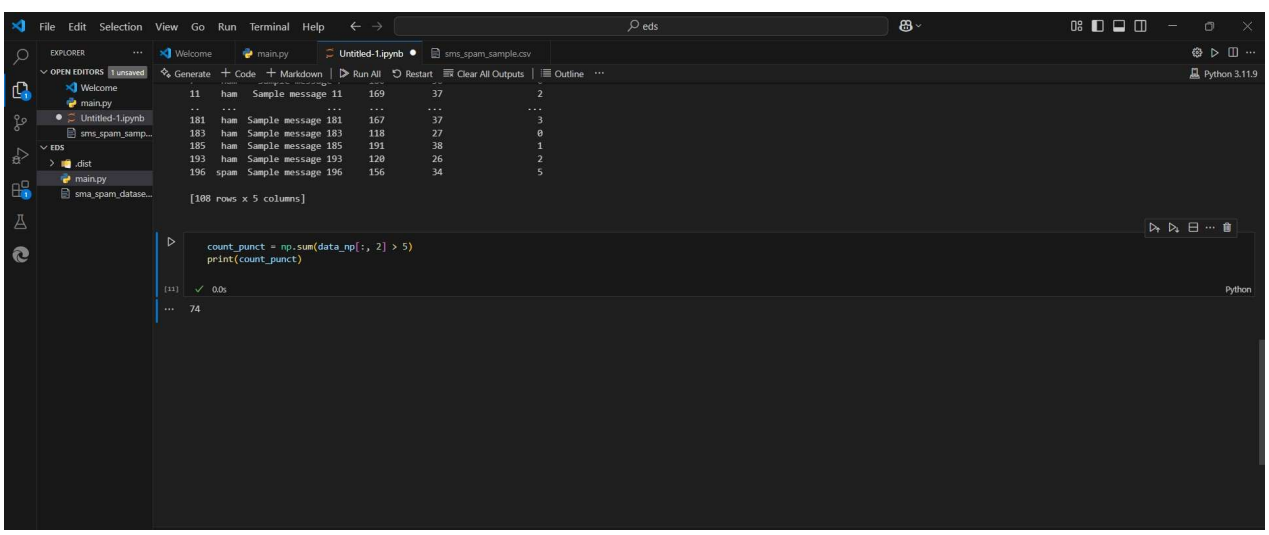
4. Find rows where message length is above average

solution:



5. Count messages with more than 5 punctuation marks

solution:



6. Normalize the 'length' column

```python
length_col = data_np[:, 0]
length_normalized = (length_col - np.min(length_col)) / (np.max(length_col) - np.min(length_col))
print(length_normalized)
```

```
[0.32446809 0.5        0.26595745 0.5        0.69148936 0.79255319
 0.75       0.89893617 0.14361702 0.18085106 0.05851064 0.84042553
 0.36702128 0.98404255 0.44680851 0.13829787 0.34042553 0.89361702
 0.2287234  0.31914894 0.97340426 0.70212766 0.13829787 0.13829787
 0.56382979 0.22340426 0.43617021 0.14893617 1.         0.38829787
 0.67021277 0.4787234  1.         0.67553191 0.63297872 0.13297872
 1.         0.63297872 0.60638298 0.00531915 0.53723404 0.81382979
 0.71808511 0.31914894 0.86702128 0.2606383  0.90425532 0.79787234
 0.30319149 0.61702128 0.84042553 0.5        0.94680851 0.59042553
 0.31914894 0.9787234  0.26595745 0.05319149 0.19680851 0.68085106
 0.68617021 0.59042553 0.52659574 0.59042553 0.96808511 0.42021277
 0.98404255 0.59042553 0.         0.68085106 0.27659574 0.45212766
 0.67553191 0.7712766  0.65957447 0.68085106 0.2712766  0.90425532
 0.84042553 0.84042553 0.35106383 0.96276596 0.96808511 0.64361702
 0.7606383  0.19148936 0.11702128 0.35638298 0.60638298 0.5106383
 0.7287234  0.75531915 0.50531915 0.64893617 0.98404255 0.36170213
 0.48404255 0.00531915 0.77659574 0.98404255 0.86170213 0.7712766
 0.46808511 0.7712766  0.77659574 0.5        0.26595745 0.84574468
 0.88297872 0.67021277 0.19680851 0.42553191 0.54255319 0.67553191
 0.04787234 0.97340426 0.93617021 0.79255319 0.83510638 0.21276596
 0.51595745 0.02659574 0.75531915 0.46808511 0.58510638 0.30851064
 0.59042553 0.         0.67553191 0.24468085 0.73404255 0.18617021
 0.84042553 0.03723404 0.51595745 0.7712766  0.24468085 0.68617021
 0.77659574 0.79787234 0.27659574 0.62765957 0.84574468 0.79787234
 0.60638298 0.38829787 0.59042553 0.86170213 0.87234043 0.54255319
...
 0.10638298 0.82978723 0.19148936 0.56914894 0.2606383  0.95744681
 0.03191489 0.13297872 0.13297872 0.10106383 0.14893617 0.50531915
```

7. Compute standard deviation of 'num_words'

.

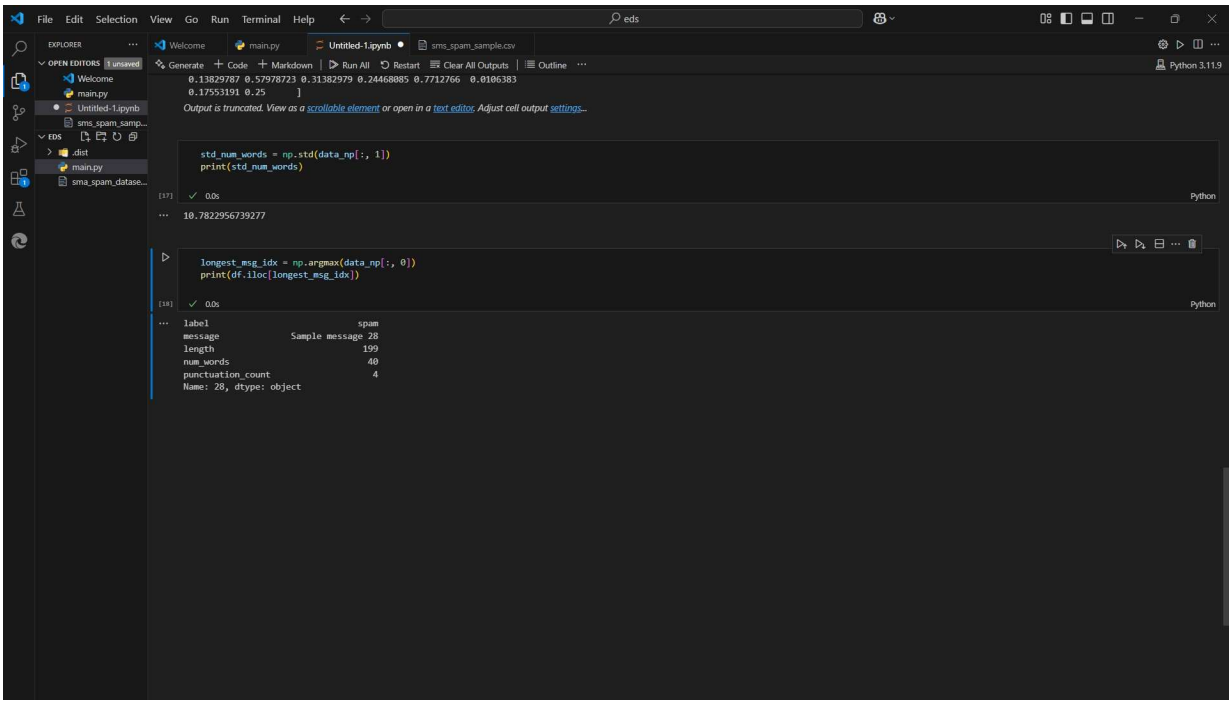
solution:

```python
std_num_words = np.std(data_np[:, 1])
print(std_num_words)
```

```
10.7822956739277
```

8. Find index of longest message.

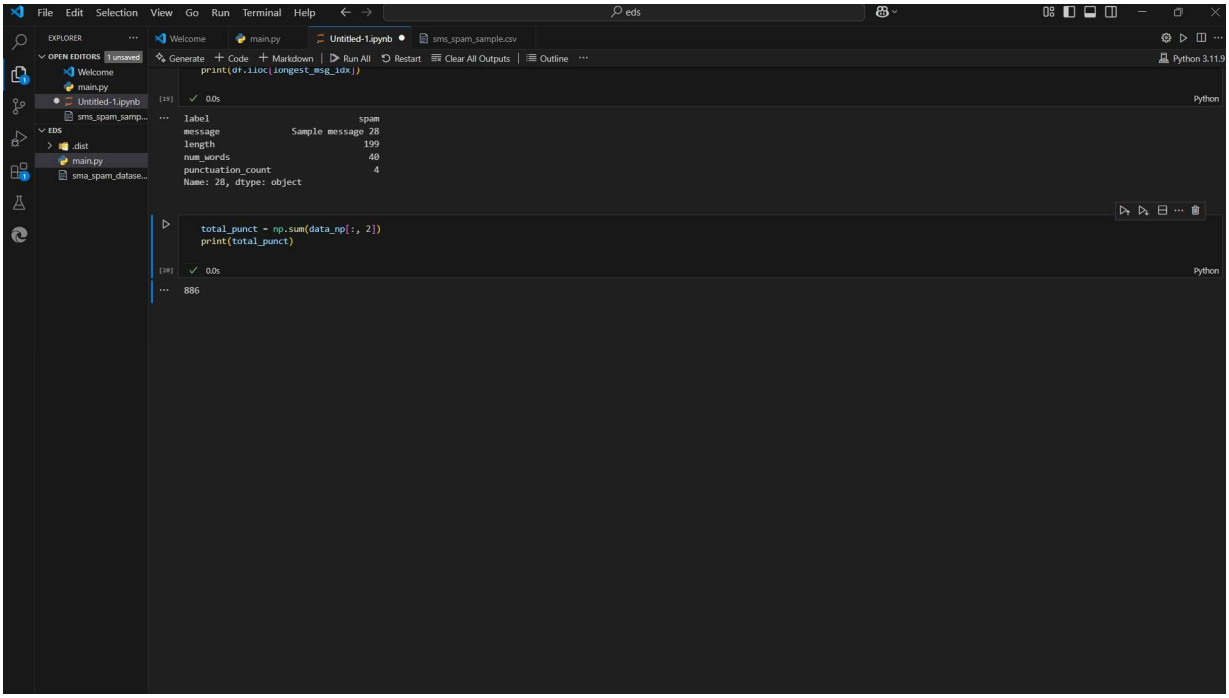
solution:



9. Sum of punctuation across all messages

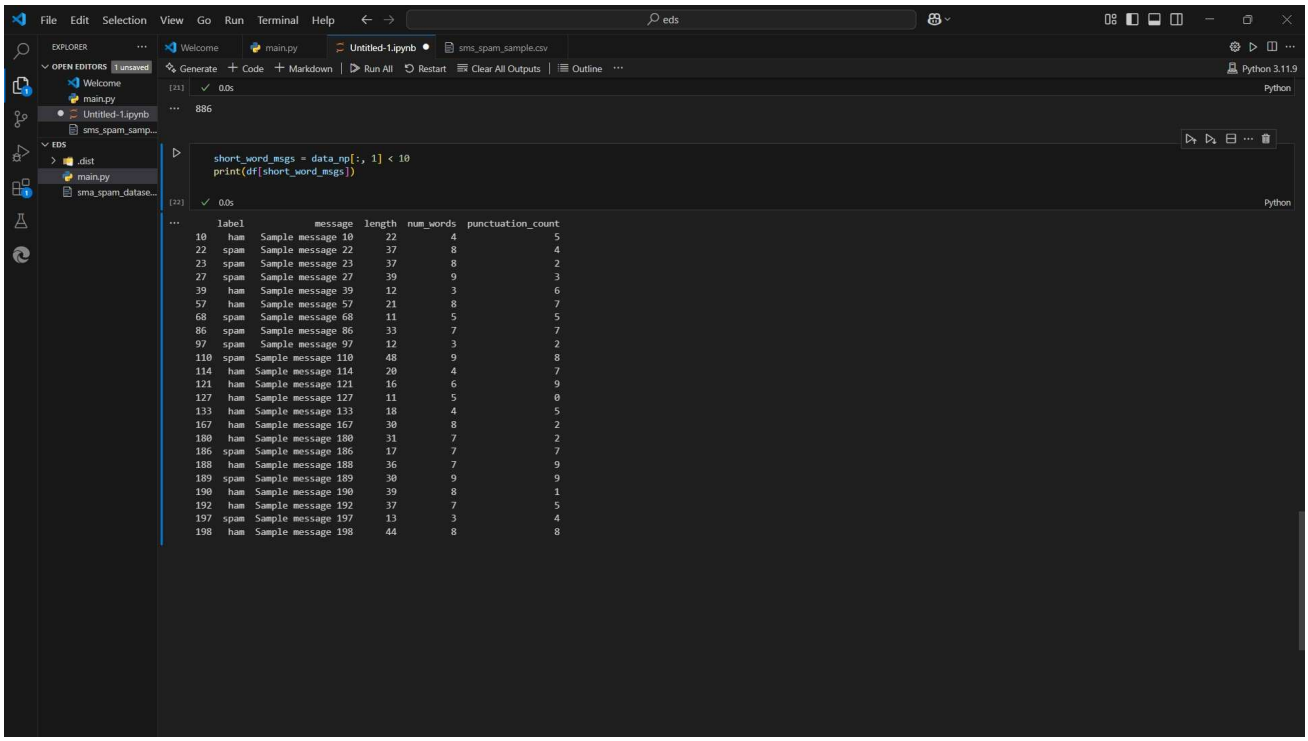
.

solution:



10. Messages where number of words is less than 10

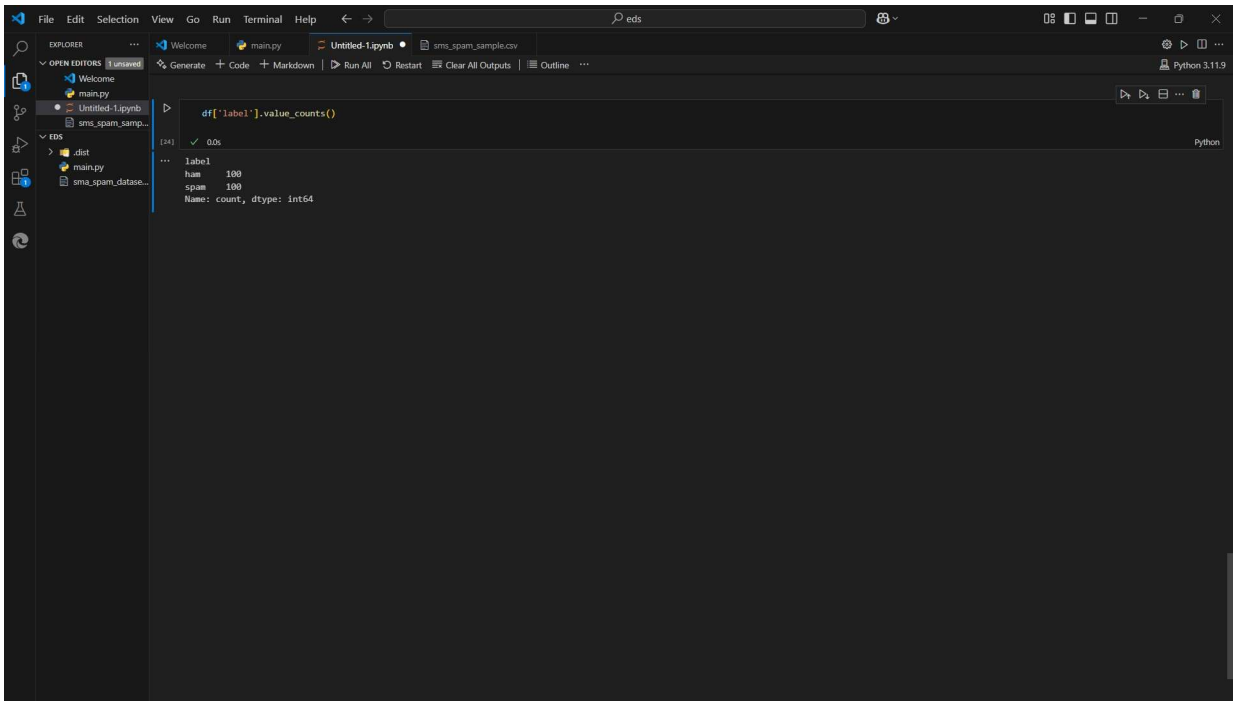
solution:



pandas:

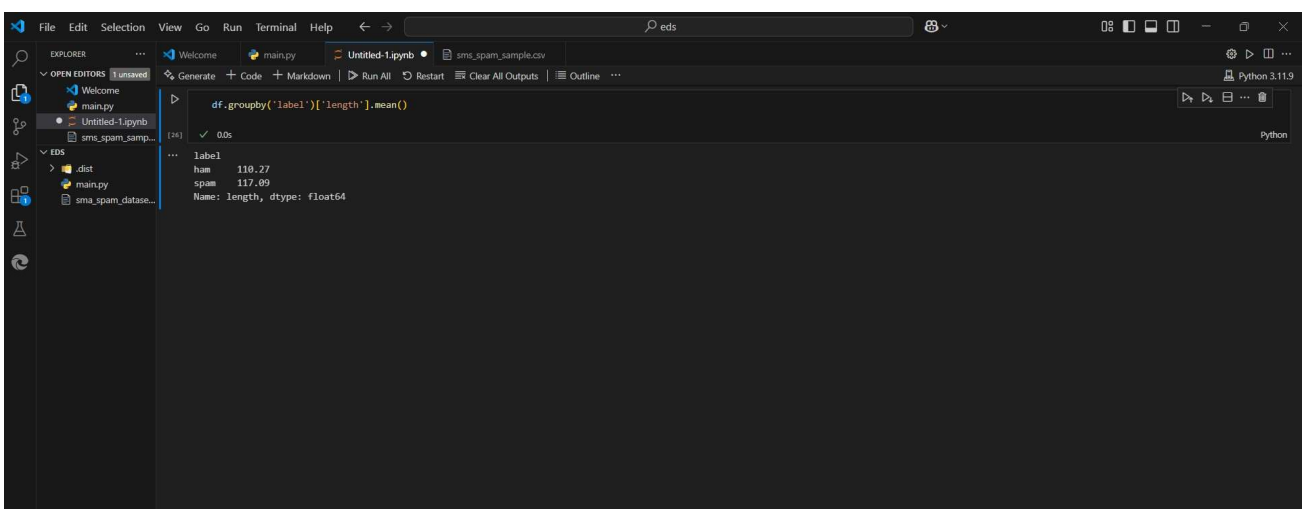
1. How many spam and ham messages are there?

solution:



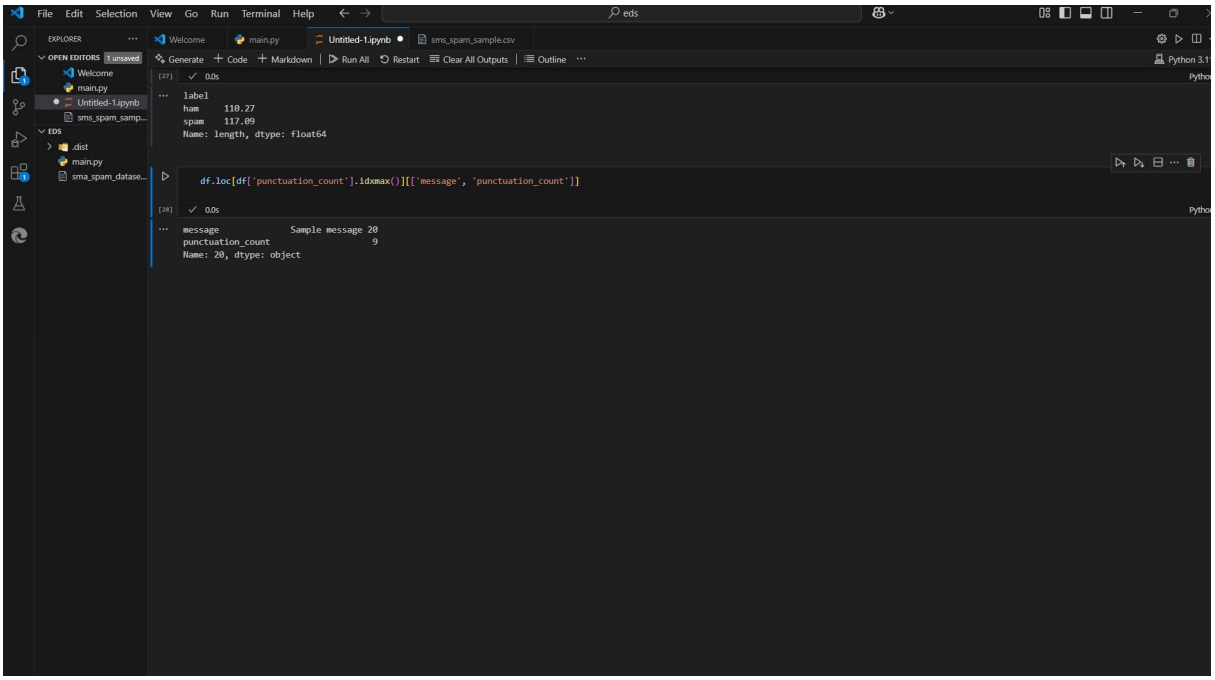
2. What is the average length of spam and ham messages?.

solution:



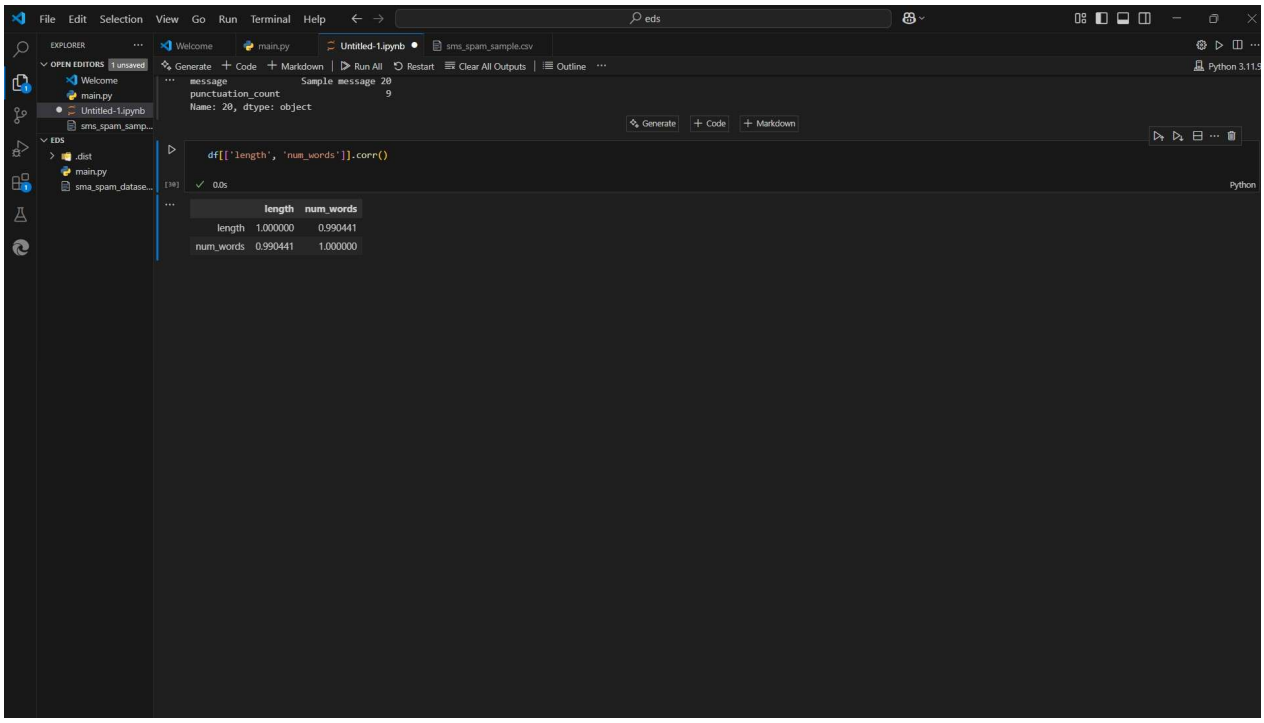
3. Which message has the most punctuation?

solution:



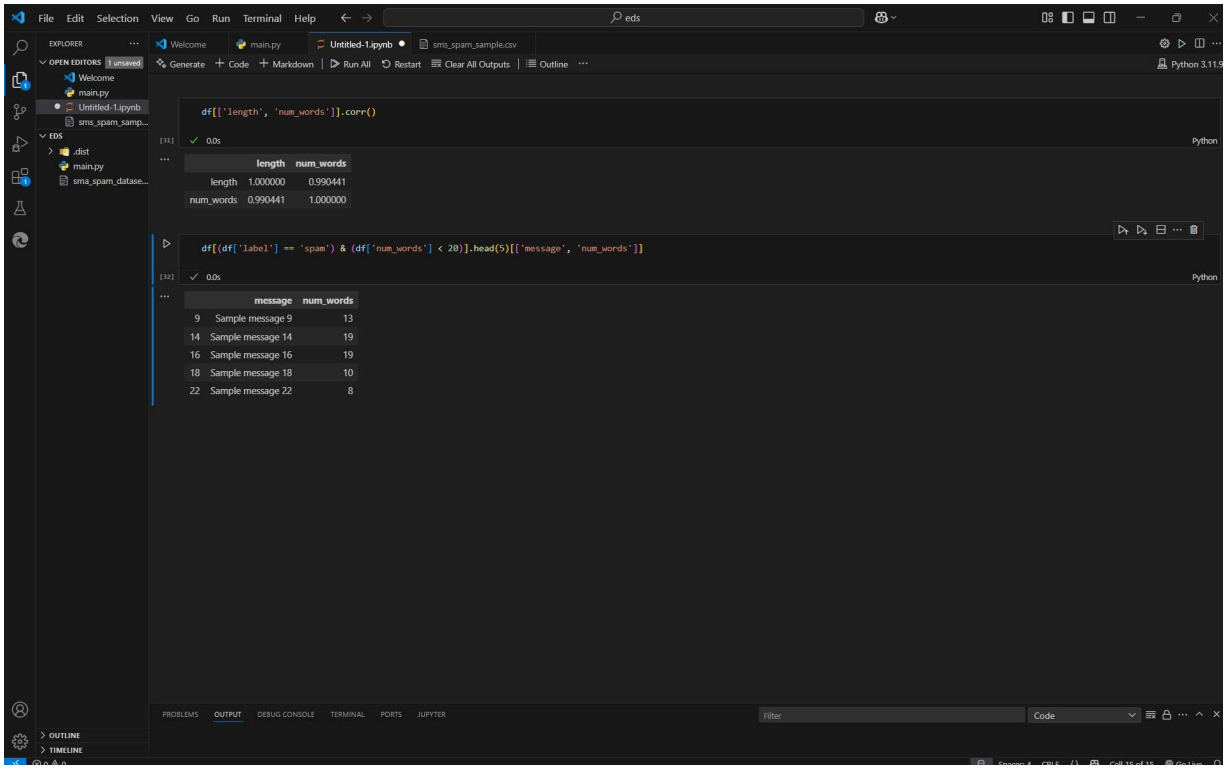
4. What is the correlation between message length and number of words?

solution:



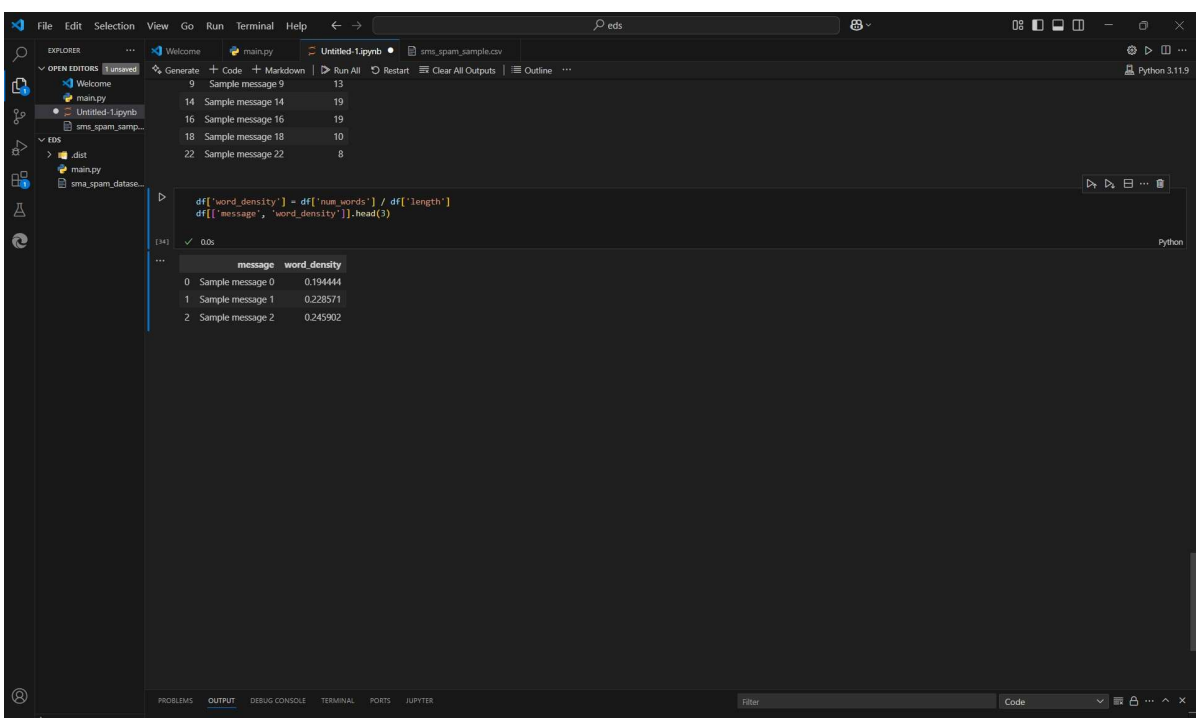
5. What are the first 5 spam messages with less than 20 words?

solution:



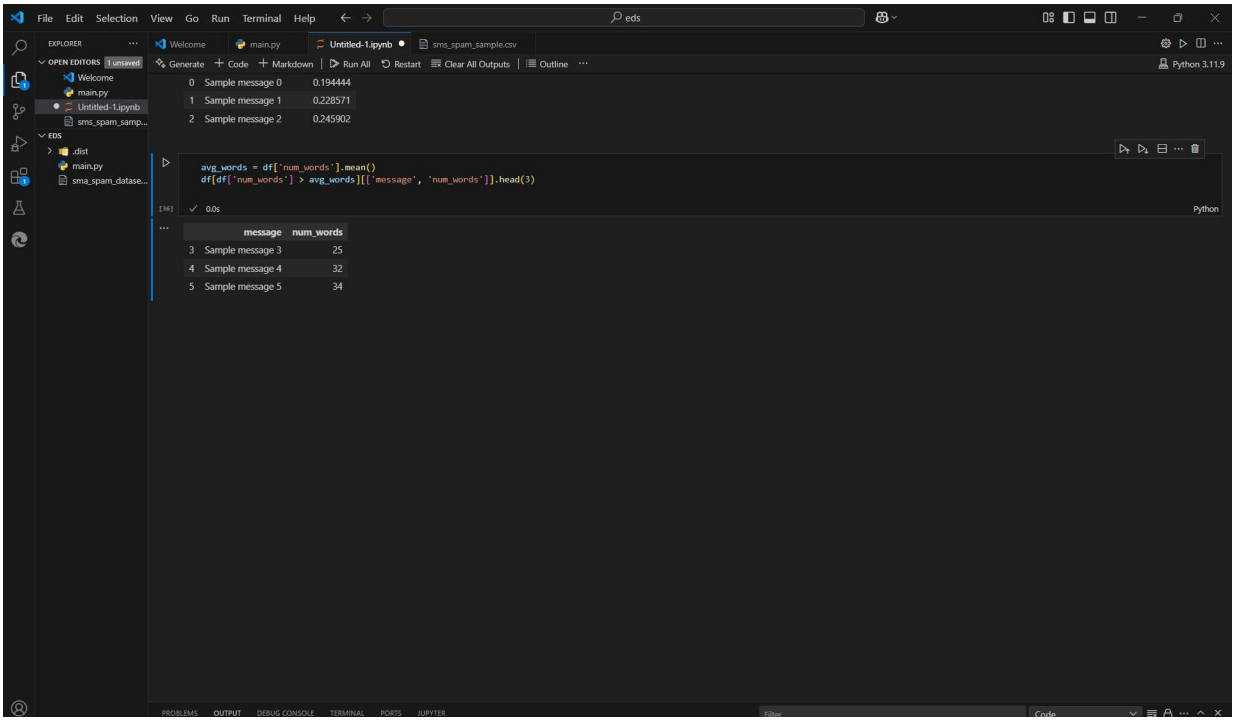
6. Add a column for word density (words per character) and show the top 3 rows

solution:



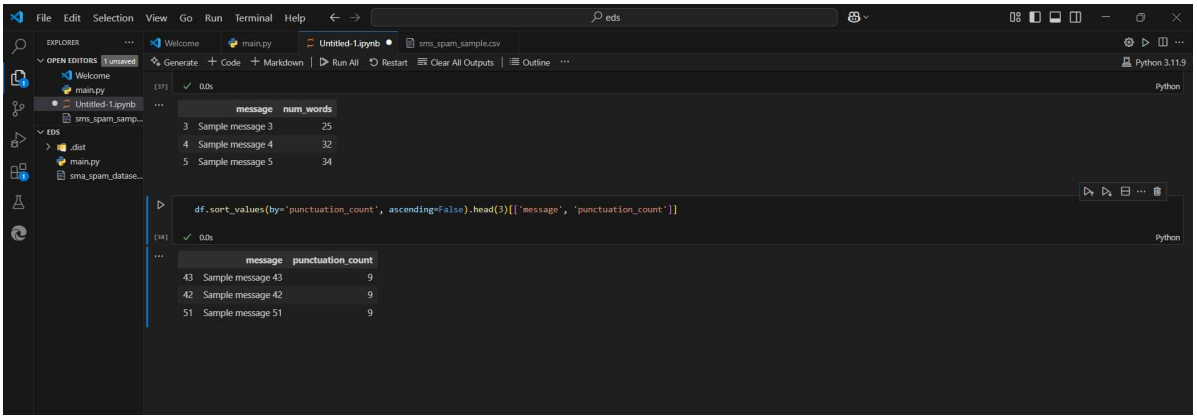
7. What are the messages with word counts above the average?

solution:



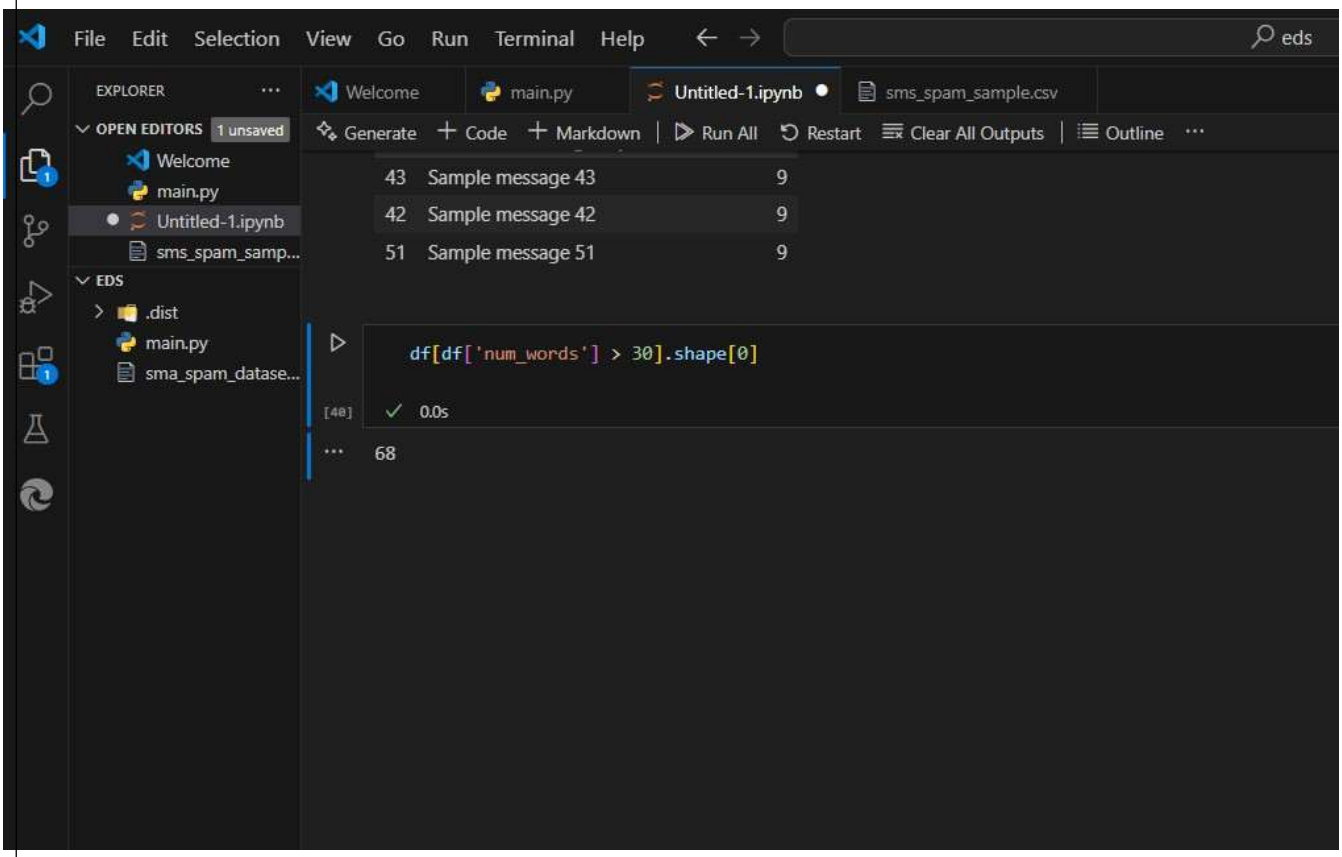
8. Sort the messages by punctuation count and show the top 3

solution:



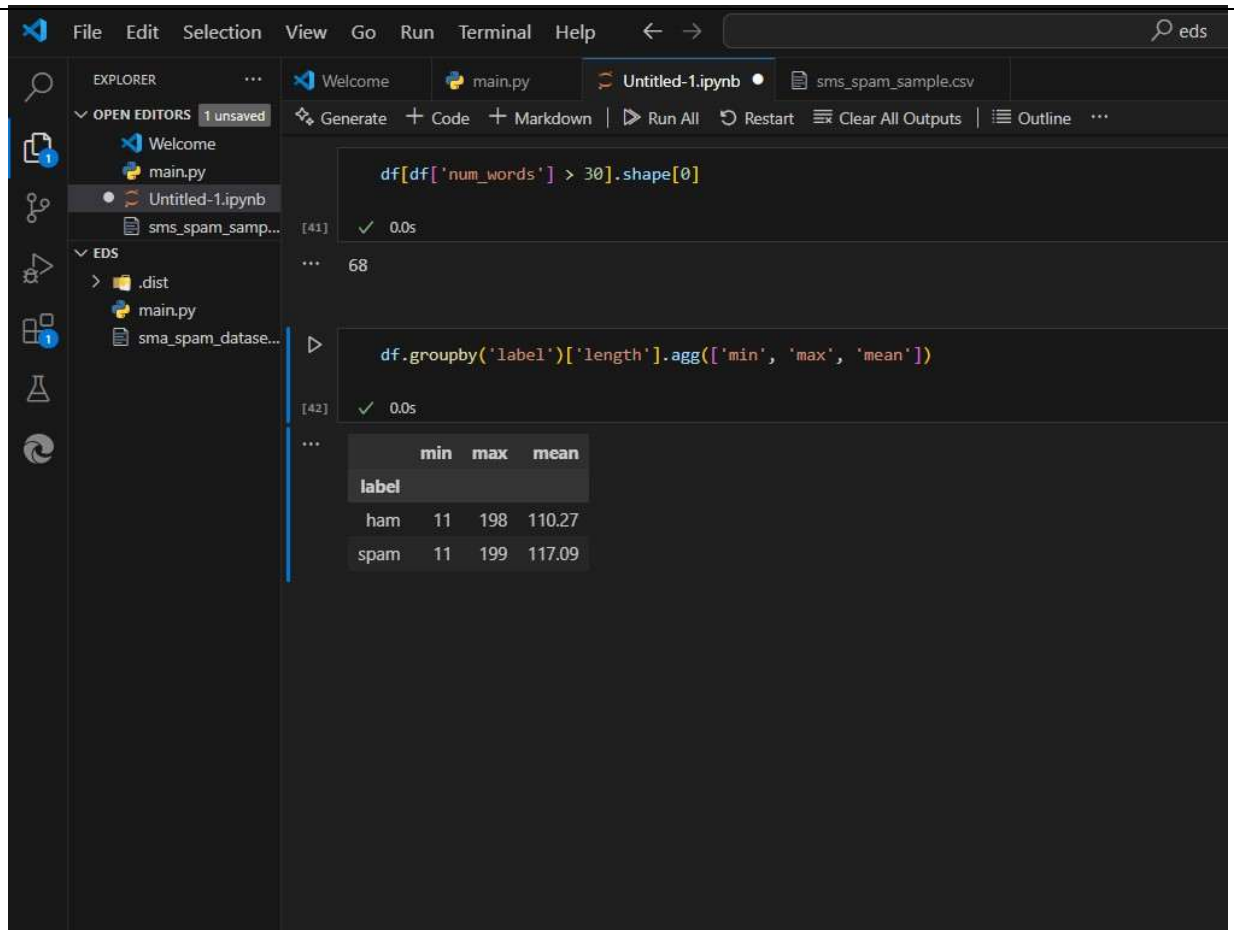
9. Count how many messages have more than 30 words

solution:



10. Group by label and get max, min, mean of length

solution:


File Edit Selection View Go Run Terminal Help
eds
EXPLORER
OPEN EDITORS 1 unsaved
Welcome
main.py
Untitled-1.ipynb
sms_spam_samp...
EDS
.dist
main.py
sma_spam_datase...
Welcome
main.py
Untitled-1.ipynb
sms_spam_sample.csv
Generate
Code
Markdown
Run All
Restart
Clear All Outputs
Outline
df[df['num_words'] > 30].shape[0]
[41]
0.0s
68
df.groupby('label')['length'].agg(['min', 'max', 'mean'])
[42]
0.0s
min max mean
label
ham 11 198 110.27
spam 11 199 117.09

---end---