DEERWALK INSTITUTE OF TECHNOLOGY

Lab 5: Read text and construct bigrams and various probability tasks.

(Artificial Intelligence)

SUBMITTED BY:

NAME: Kriston Pal

ROLL NO.: 0511

SECTION: A

SUBMITTED TO:

Birodh Rijal

KATHMANDU, NEPAL

Objective:

To take *Shakespeare.txt* as an input that contains all the works of Shakespeare. Tokenize the string and remove stop words from it. Perform a following task on obtained dataset:

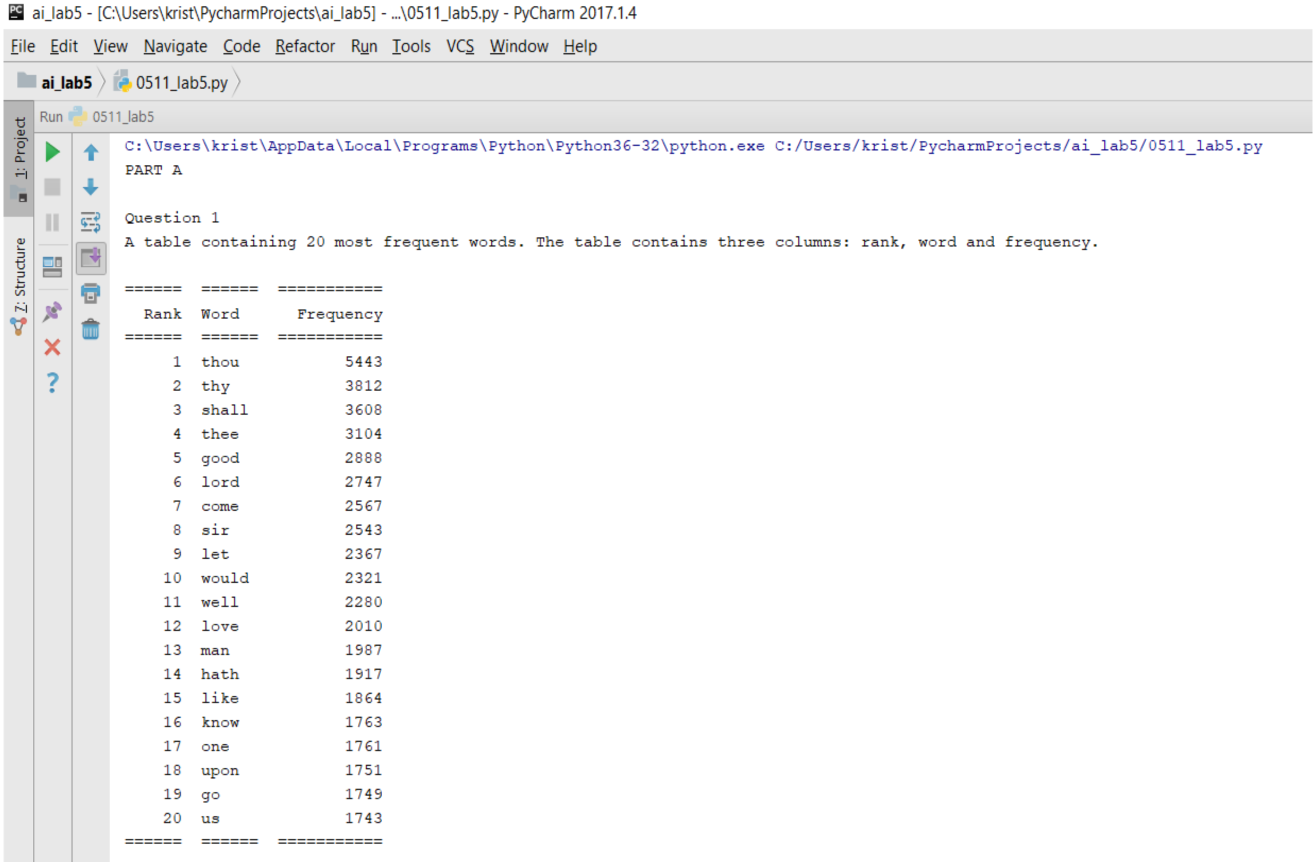
- Find frequency of each word and rank it
- Find frequency of word-pairs
- Apply different probability rule and analyze the output

Output:

Part A

1. A table containing 20 most frequent words. The table contains three columns: rank, word and frequency.

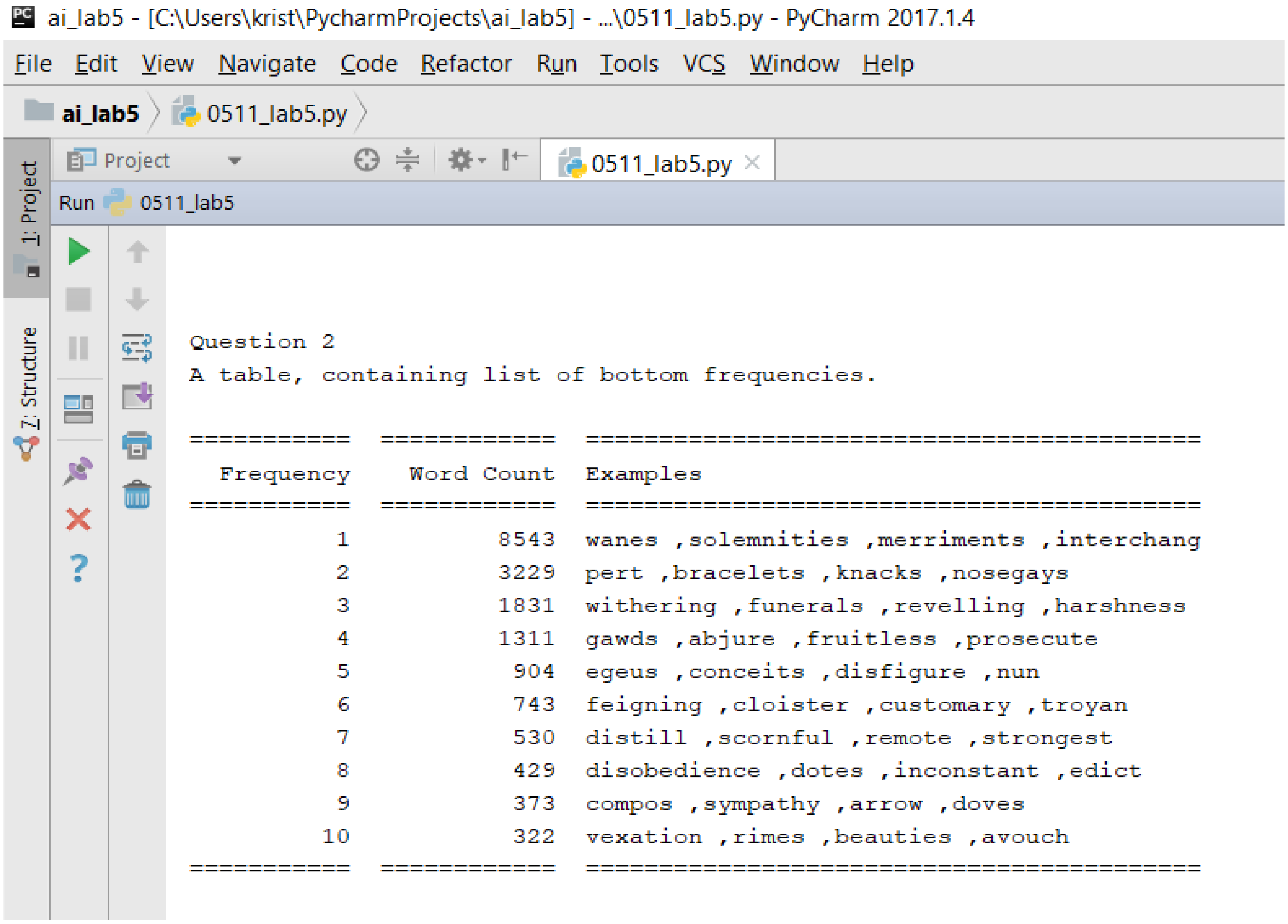
Output:


ai_lab5 - [C:\Users\krist\PycharmProjects\ai_lab5] - ...\0511_lab5.py - PyCharm 2017.1.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
ai_lab5
0511_lab5.py
Run 0511_lab5
1: Project
Z: Structure

```
C:\Users\krist\AppData\Local\Programs\Python\Python36-32\python.exe C:/Users/krist/PycharmProjects/ai_lab5/0511_lab5.py
PART A

Question 1
A table containing 20 most frequent words. The table contains three columns: rank, word and frequency.

======  ======  ===========
  Rank  Word      Frequency
======  ======  ===========
     1  thou           5443
     2  thy            3812
     3  shall          3608
     4  thee           3104
     5  good           2888
     6  lord           2747
     7  come           2567
     8  sir            2543
     9  let            2367
    10  would          2321
    11  well           2280
    12  love           2010
    13  man            1987
    14  hath           1917
    15  like           1864
    16  know           1763
    17  one            1761
    18  upon           1751
    19  go             1749
    20  us             1743
======  ======  ===========
```

2. A table, containing list of bottom frequencies. The table contains three columns: frequency, word count and example words. You are supposed to print word counts for frequencies 10 to 1. The rows in this table show how many words have frequency 10,9,8...1 with example of some of the words.

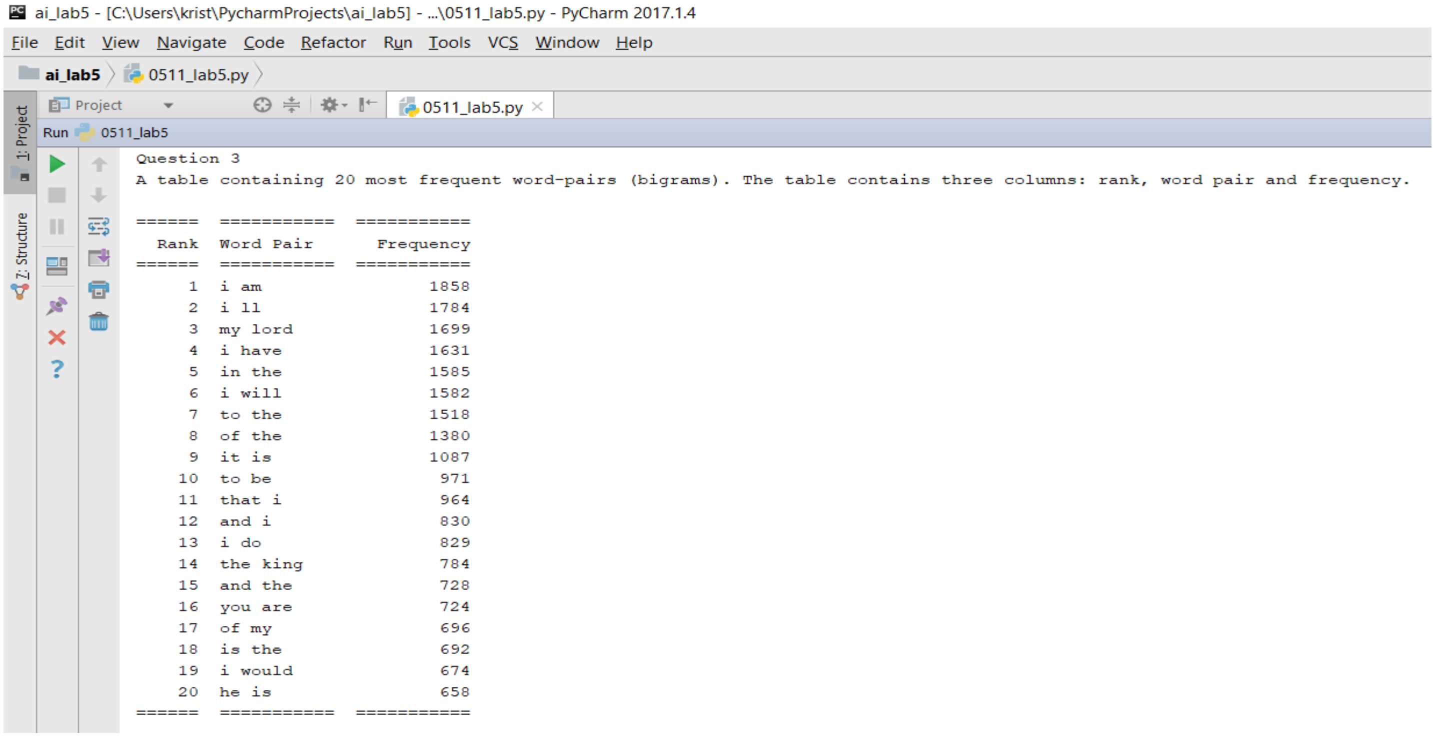
Output:



```
Question 2
A table, containing list of bottom frequencies.

===========  ============  ==========================================
  Frequency    Word Count  Examples
===========  ============  ==========================================
          1          8543  wanes ,solemnities ,merriments ,interchang
          2          3229  pert ,bracelets ,knacks ,nosegays
          3          1831  withering ,funerals ,revelling ,harshness
          4          1311  gawds ,abjure ,fruitless ,prosecute
          5           904  egeus ,conceits ,disfigure ,nun
          6           743  feigning ,cloister ,customary ,troyan
          7           530  distill ,scornful ,remote ,strongest
          8           429  disobedience ,dotes ,inconstant ,edict
          9           373  compos ,sympathy ,arrow ,doves
         10           322  vexation ,rimes ,beauties ,avouch
===========  ============  ==========================================
```

3. A table containing 20 most frequent word-pairs (bigrams). The table contains three columns: rank, word pair and frequency.

Output:


ai_lab5 - [C:\Users\krist\PycharmProjects\ai_lab5] - ...\0511_lab5.py - PyCharm 2017.1.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
ai_lab5
0511_lab5.py
Project
0511_lab5.py
Run 0511_lab5
1: Project
Z: Structure

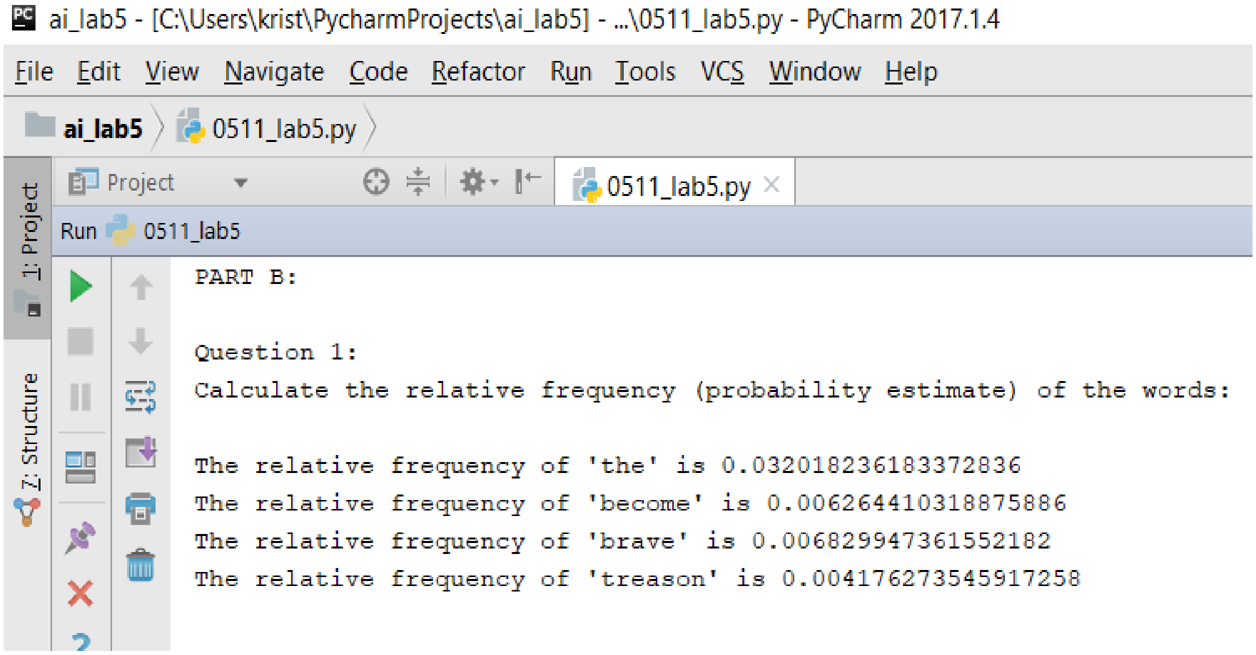
```
Question 3
A table containing 20 most frequent word-pairs (bigrams). The table contains three columns: rank, word pair and frequency.

======  ===========  ===========
  Rank  Word Pair      Frequency
======  ===========  ===========
     1  i am                1858
     2  i ll                1784
     3  my lord             1699
     4  i have              1631
     5  in the              1585
     6  i will              1582
     7  to the              1518
     8  of the              1380
     9  it is               1087
    10  to be                971
    11  that i               964
    12  and i                830
    13  i do                 829
    14  the king             784
    15  and the              728
    16  you are              724
    17  of my                696
    18  is the               692
    19  i would              674
    20  he is                658
======  ===========  ===========
```

Part B

1. Calculate the relative frequency (probability estimate) of the words:

(a) “the" (b) “become" (d) “brave" (e) “treason"

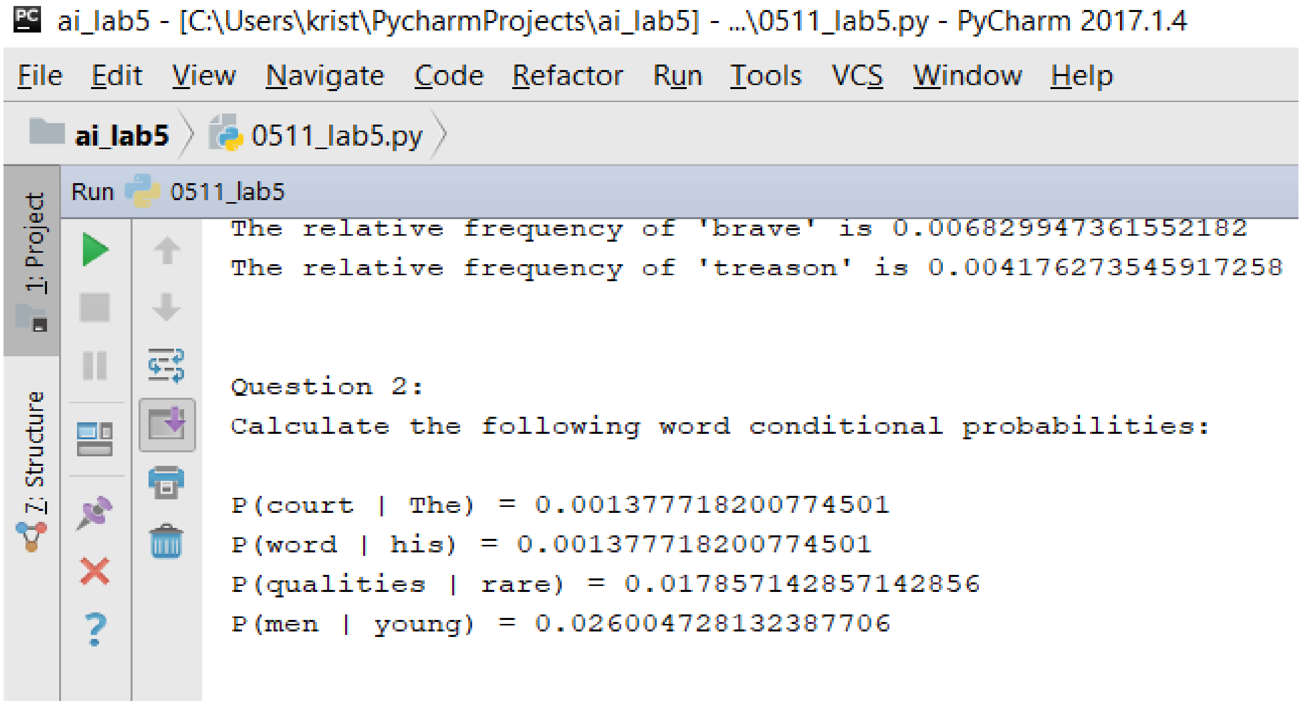
[Note: P(the) = count(the) / N . Here, count(the) is the frequency of “the" and “N" is the total word count.] **Output:**


ai_lab5 - [C:\Users\krist\PycharmProjects\ai_lab5] - ...\0511_lab5.py - PyCharm 2017.1.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
ai_lab5
0511_lab5.py
Project
0511_lab5.py
Run 0511_lab5
1: Project
Z: Structure

```
PART B:

Question 1:
Calculate the relative frequency (probability estimate) of the words:

The relative frequency of 'the' is 0.032018236183372836
The relative frequency of 'become' is 0.006264410318875886
The relative frequency of 'brave' is 0.006829947361552182
The relative frequency of 'treason' is 0.004176273545917258
```

2. Calculate the following word conditional probabilities:
(a) P(court | The) (b) P(word | his) (c) P(qualities | rare) (d) P(men | young)
[Read P(B | A) as “the probability with which word B follows word A". Note: P(B | A) = count(A;B) | count(A)]

Output:


ai_lab5 - [C:\Users\krist\PycharmProjects\ai_lab5] - ...\0511_lab5.py - PyCharm 2017.1.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
ai_lab5
0511_lab5.py
Run
0511_lab5
1: Project
7: Structure

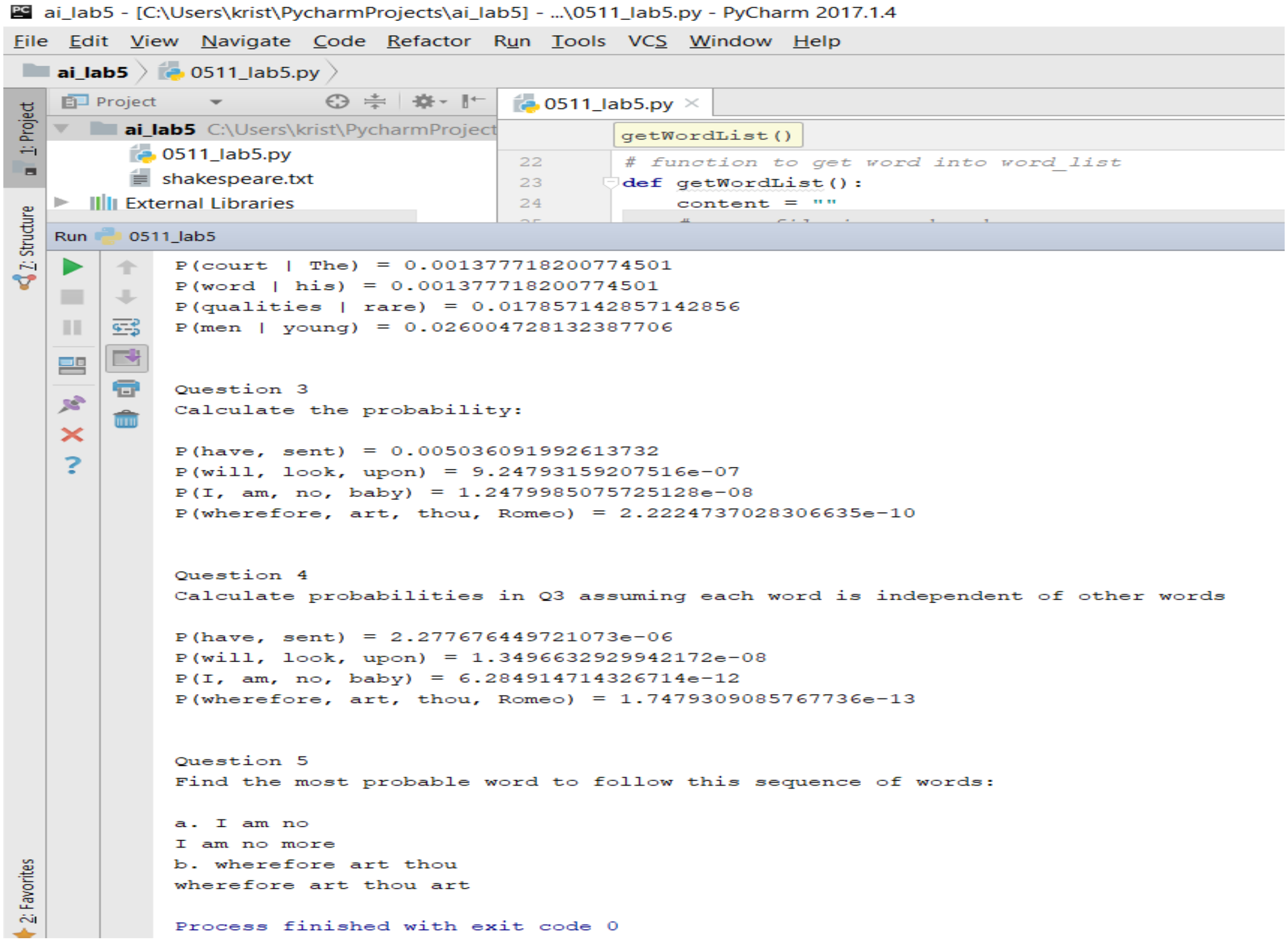
```
The relative frequency of 'brave' is 0.006829947361552182
The relative frequency of 'treason' is 0.004176273545917258


Question 2:
Calculate the following word conditional probabilities:

P(court | The) = 0.001377718200774501
P(word | his) = 0.001377718200774501
P(qualities | rare) = 0.017857142857142856
P(men | young) = 0.026004728132387706
```

3. Calculate the probability:
(a) P(have, sent) (b) P(will, look, upon) (c) P(I, am, no, baby) (d) P(wherefore, art, thou, Romeo)
Hint à use the chain rule (multiplication rule):

4. Calculate probabilities in Q3 assuming each word is independent of other words (independence assumption).

5. Find the most probable word to follow this sequence of words:
(a) I am no (b) wherefore art thou

OUTPUT:


ai_lab5 - [C:\Users\krist\PycharmProjects\ai_lab5] - ...\0511_lab5.py - PyCharm 2017.1.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
ai_lab5
0511_lab5.py
1: Project
7: Structure
2: Favorites
Project
ai_lab5 C:\Users\krist\PycharmProject
0511_lab5.py
shakespeare.txt
External Libraries
0511_lab5.py
getWordList()
Run
0511_lab5

```
# function to get word into word_list
def getWordList():
    content = ""
```

```
P(court | The) = 0.001377718200774501
P(word | his) = 0.001377718200774501
P(qualities | rare) = 0.017857142857142856
P(men | young) = 0.026004728132387706


Question 3
Calculate the probability:

P(have, sent) = 0.005036091992613732
P(will, look, upon) = 9.247931592075l6e-07
P(I, am, no, baby) = 1.247998507572512 8e-08
P(wherefore, art, thou, Romeo) = 2.2224737028306635e-10


Question 4
Calculate probabilities in Q3 assuming each word is independent of other words

P(have, sent) = 2.277676449721073e-06
P(will, look, upon) = 1.3496632929942172e-08
P(I, am, no, baby) = 6.284914714326714e-12
P(wherefore, art, thou, Romeo) = 1.747930908576773 6e-13


Question 5
Find the most probable word to follow this sequence of words:

a. I am no
I am no more
b. wherefore art thou
wherefore art thou art

Process finished with exit code 0
```

Conclusion:

Hence the file is read and the stop words are filtered out and then count was performed and also bigrams was constructed so that it can be manipulated easily in order to carry out various probability calculations in the words present in that file.