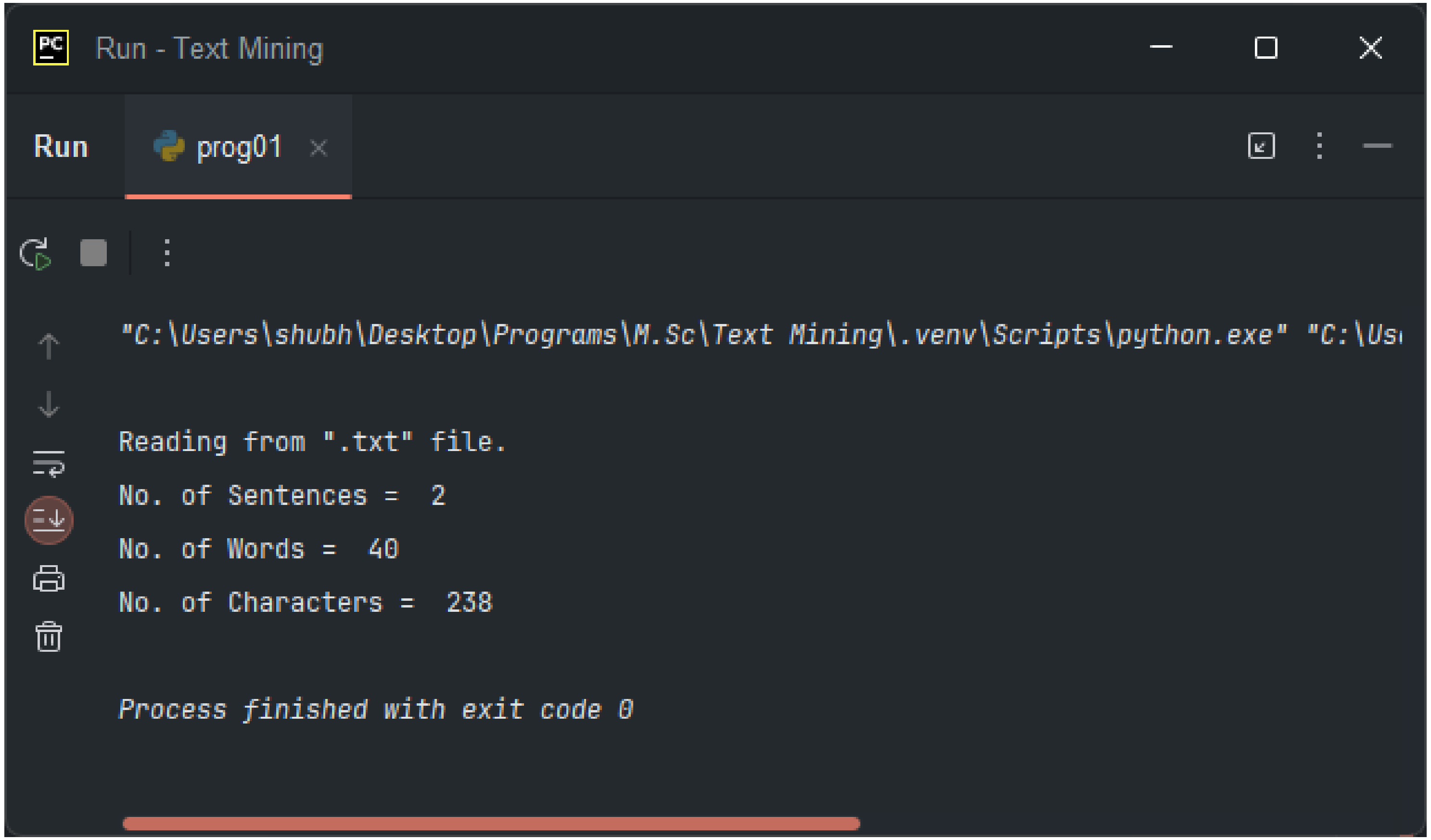
CS-313 : Text Mining (LAB)

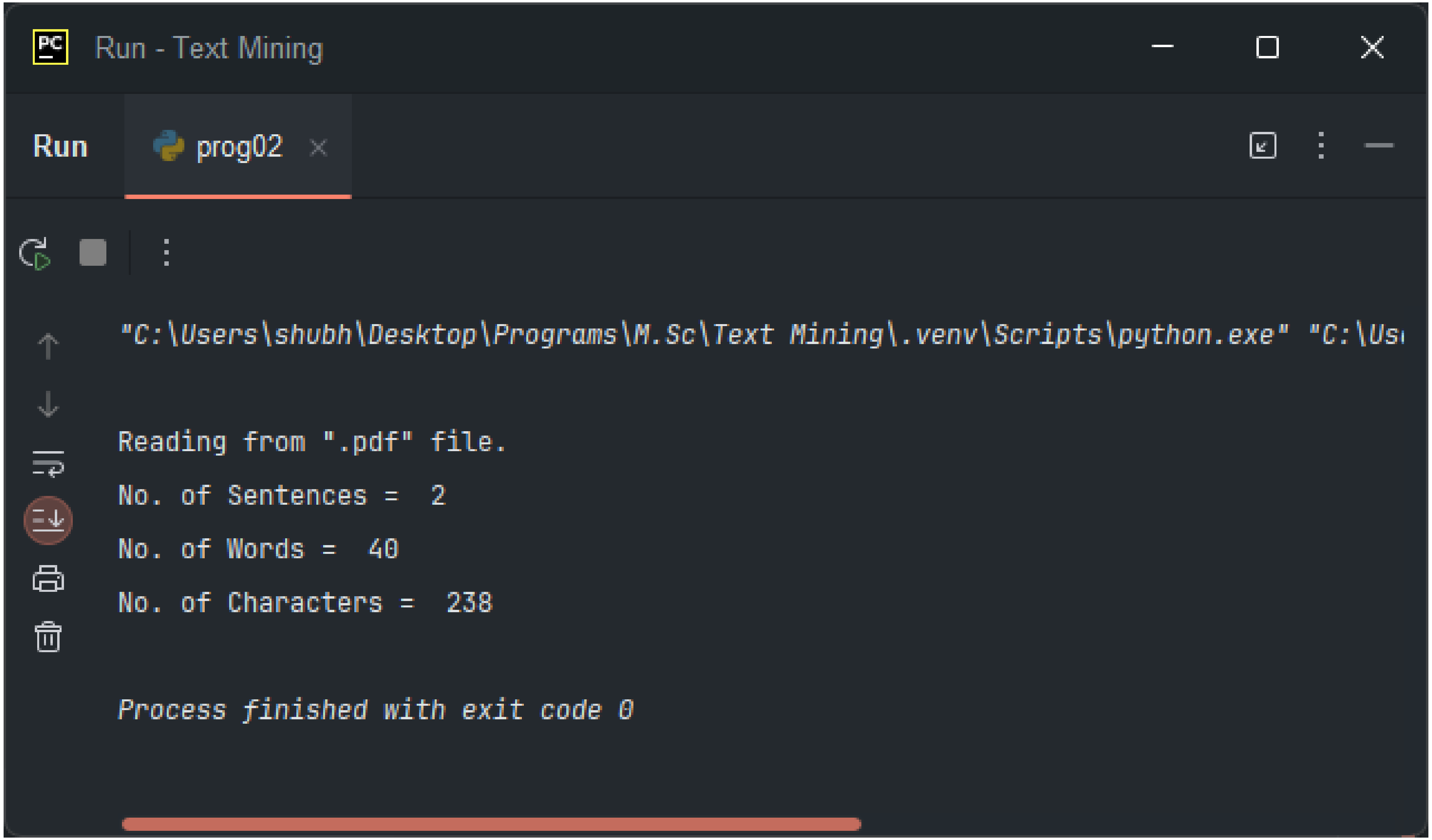
Name: Shubham Dey M.Sc. Computer Science Roll No. : 23419CMP026

Program-01 : Read from a '.txt' file and count the number of sentences, words, and characters in the file.



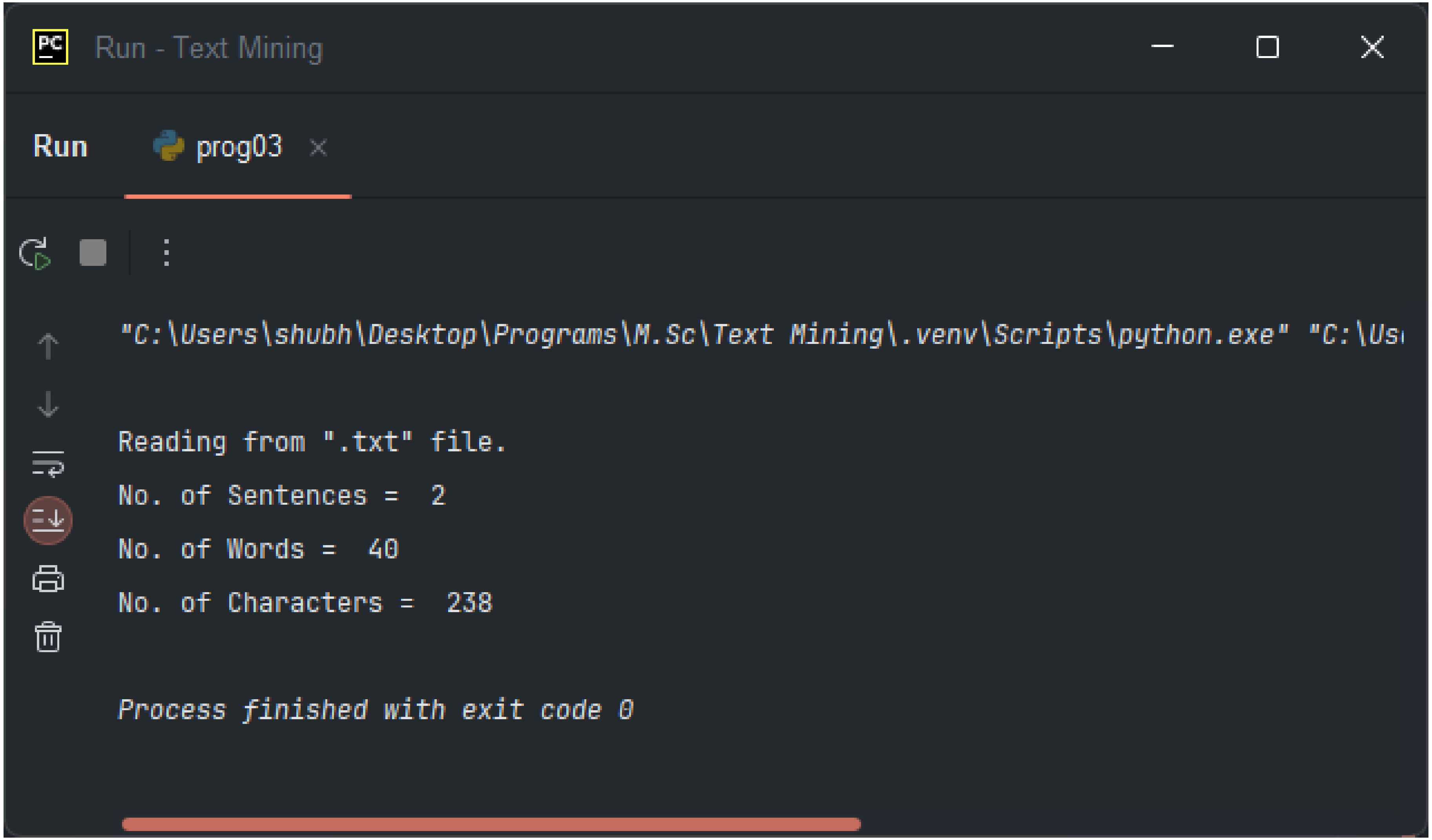
```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe" "C:\Us

Reading from ".txt" file.
No. of Sentences =  2
No. of Words =  40
No. of Characters =  238

Process finished with exit code 0
```

Program-02 : Read from a '.pdf' file and count the number of sentences, words, and characters in the file.



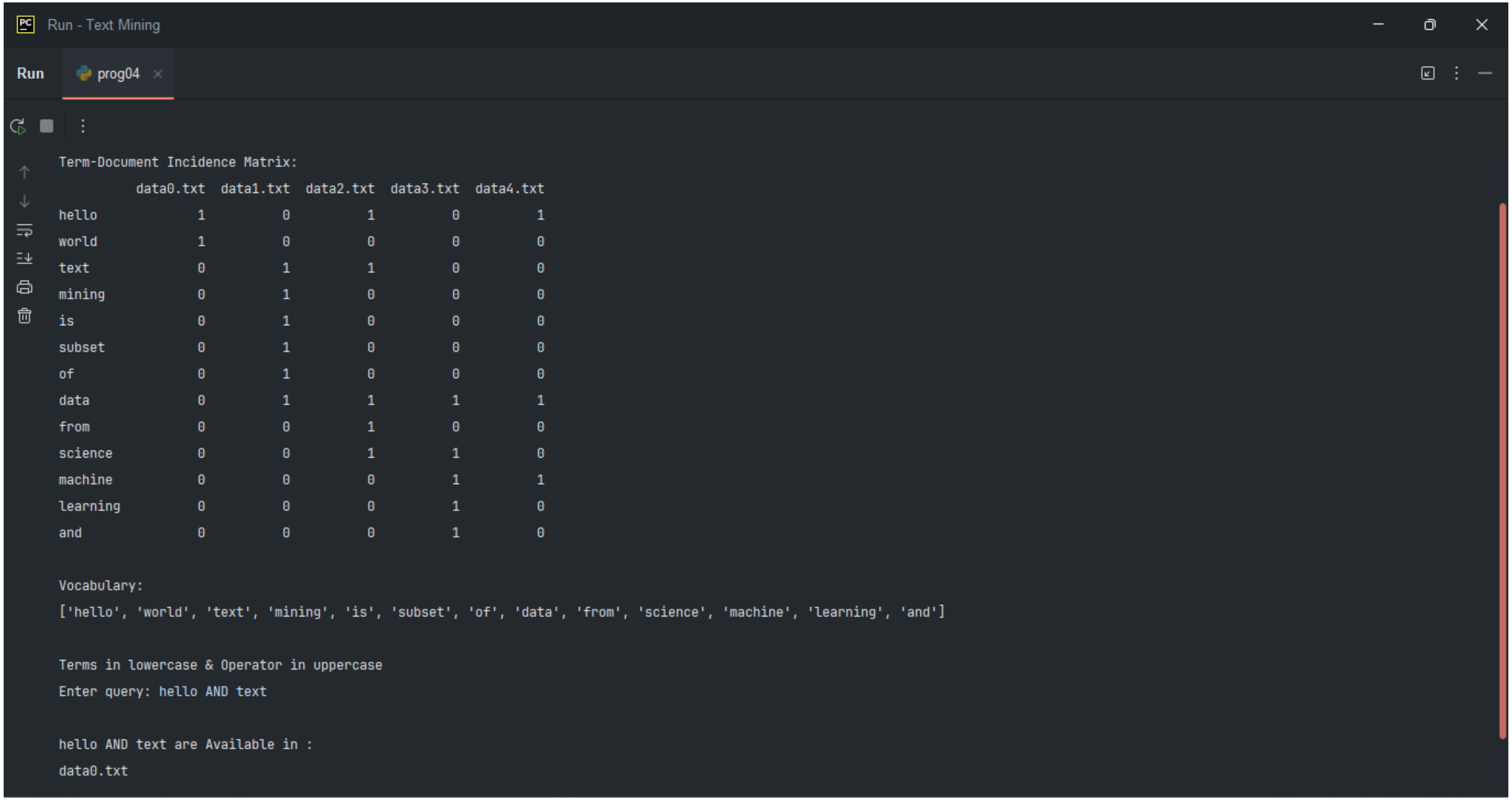
```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe" "C:\Us

Reading from ".pdf" file.
No. of Sentences =  2
No. of Words =  40
No. of Characters =  238

Process finished with exit code 0
```

Program-03 : Read from a ‘.docx’ file and count the number of sentences, words, and characters in the file.



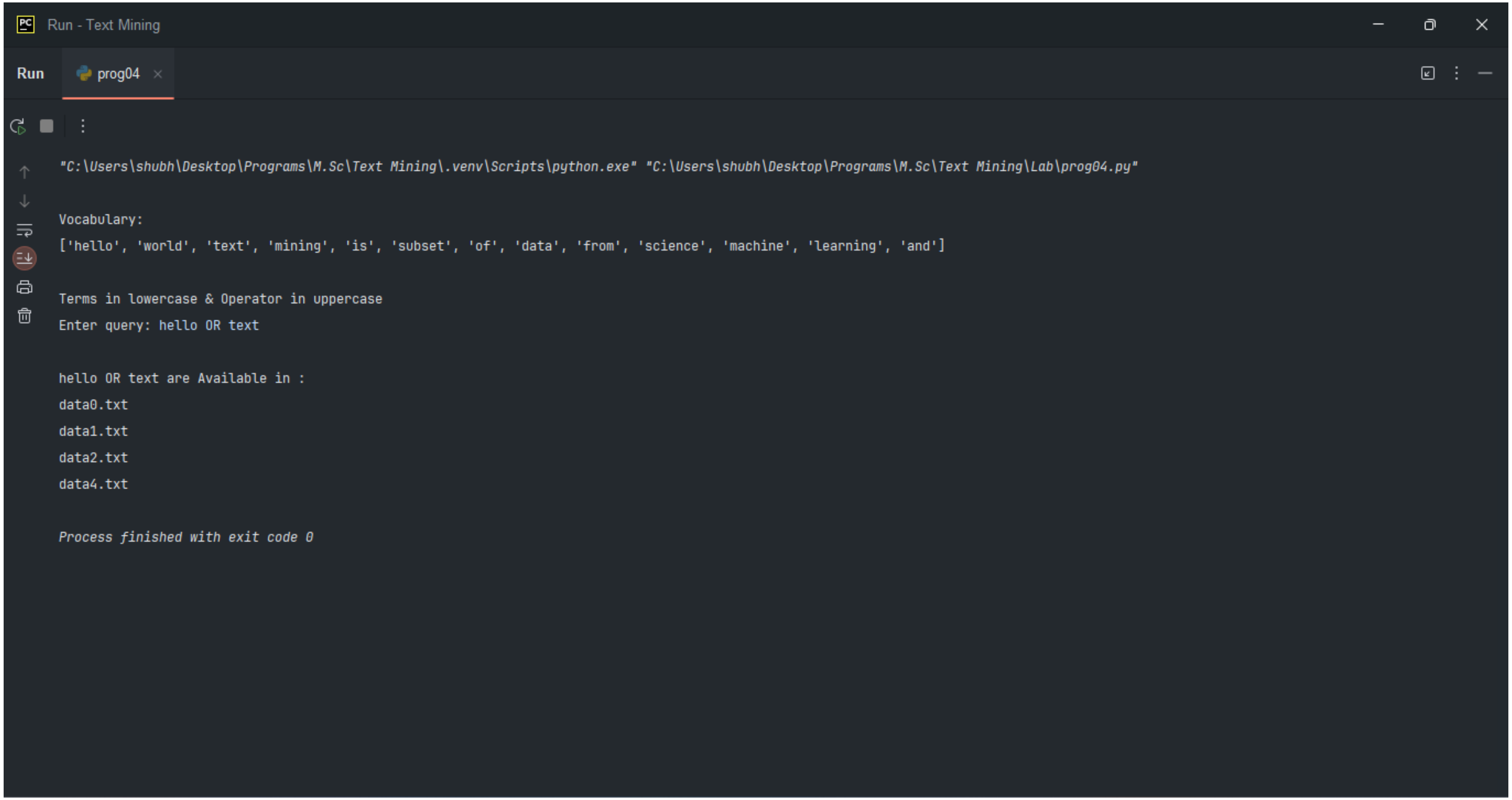
```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe" "C:\Us

Reading from ".txt" file.
No. of Sentences =  2
No. of Words =  40
No. of Characters =  238

Process finished with exit code 0
```

Program-04 : Implement Term-Document incidence matrix for boolean retrieval.



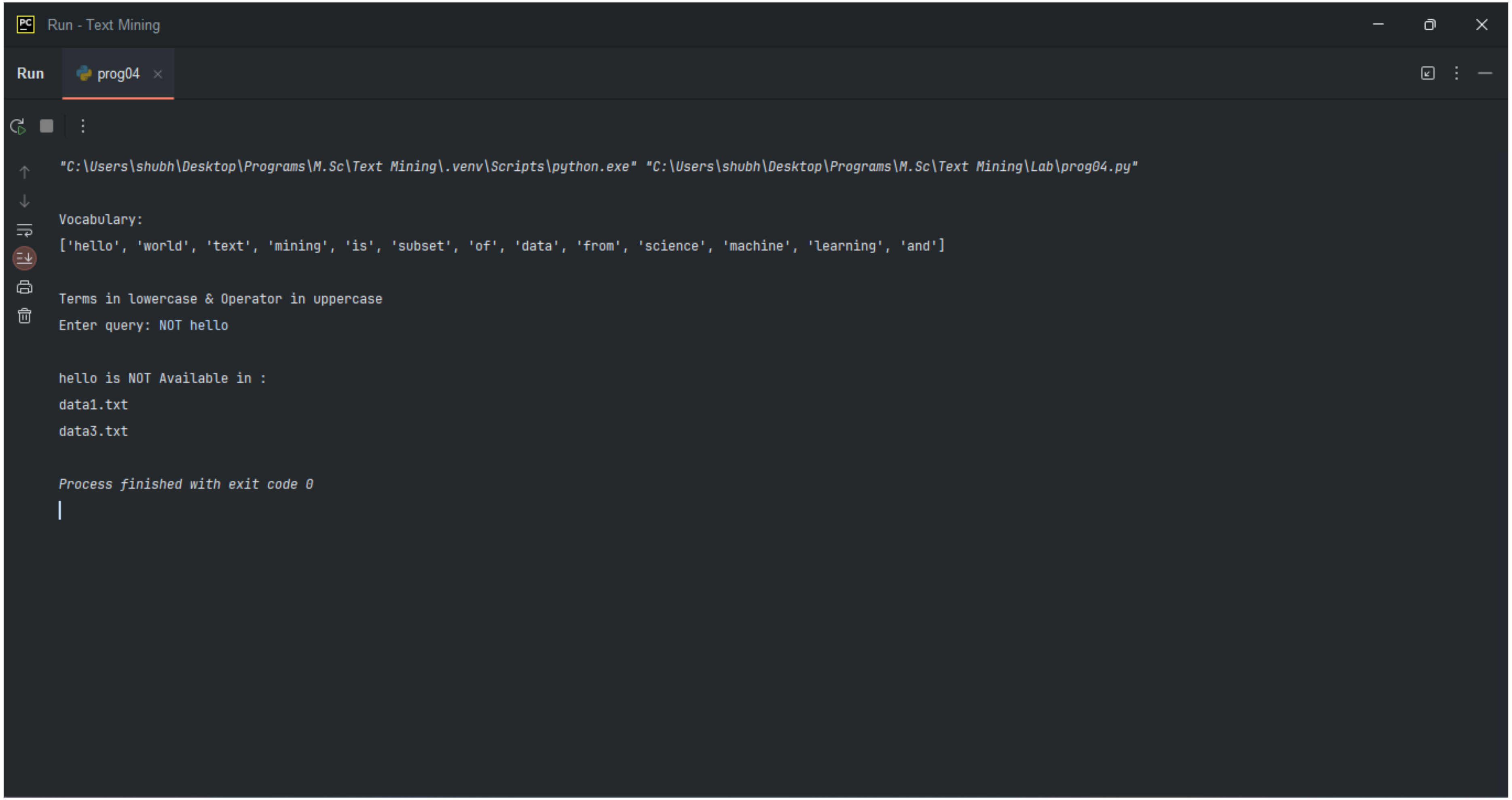
```
Term-Document Incidence Matrix:
          data0.txt  data1.txt  data2.txt  data3.txt  data4.txt
hello             1          0          1          0          1
world             1          0          0          0          0
text              0          1          1          0          0
mining            0          1          0          0          0
is                0          1          0          0          0
subset            0          1          0          0          0
of                0          1          0          0          0
data              0          1          1          1          1
from              0          0          1          0          0
science           0          0          1          1          0
machine           0          0          0          1          1
learning          0          0          0          1          0
and               0          0          0          1          0

Vocabulary:
['hello', 'world', 'text', 'mining', 'is', 'subset', 'of', 'data', 'from', 'science', 'machine', 'learning', 'and']

Terms in lowercase & Operator in uppercase
Enter query: hello AND text

hello AND text are Available in :
data0.txt
```

Query = hello AND text



```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe" "C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\Lab\prog04.py"

Vocabulary:
['hello', 'world', 'text', 'mining', 'is', 'subset', 'of', 'data', 'from', 'science', 'machine', 'learning', 'and']

Terms in lowercase & Operator in uppercase
Enter query: hello OR text

hello OR text are Available in :
data0.txt
data1.txt
data2.txt
data4.txt

Process finished with exit code 0
```

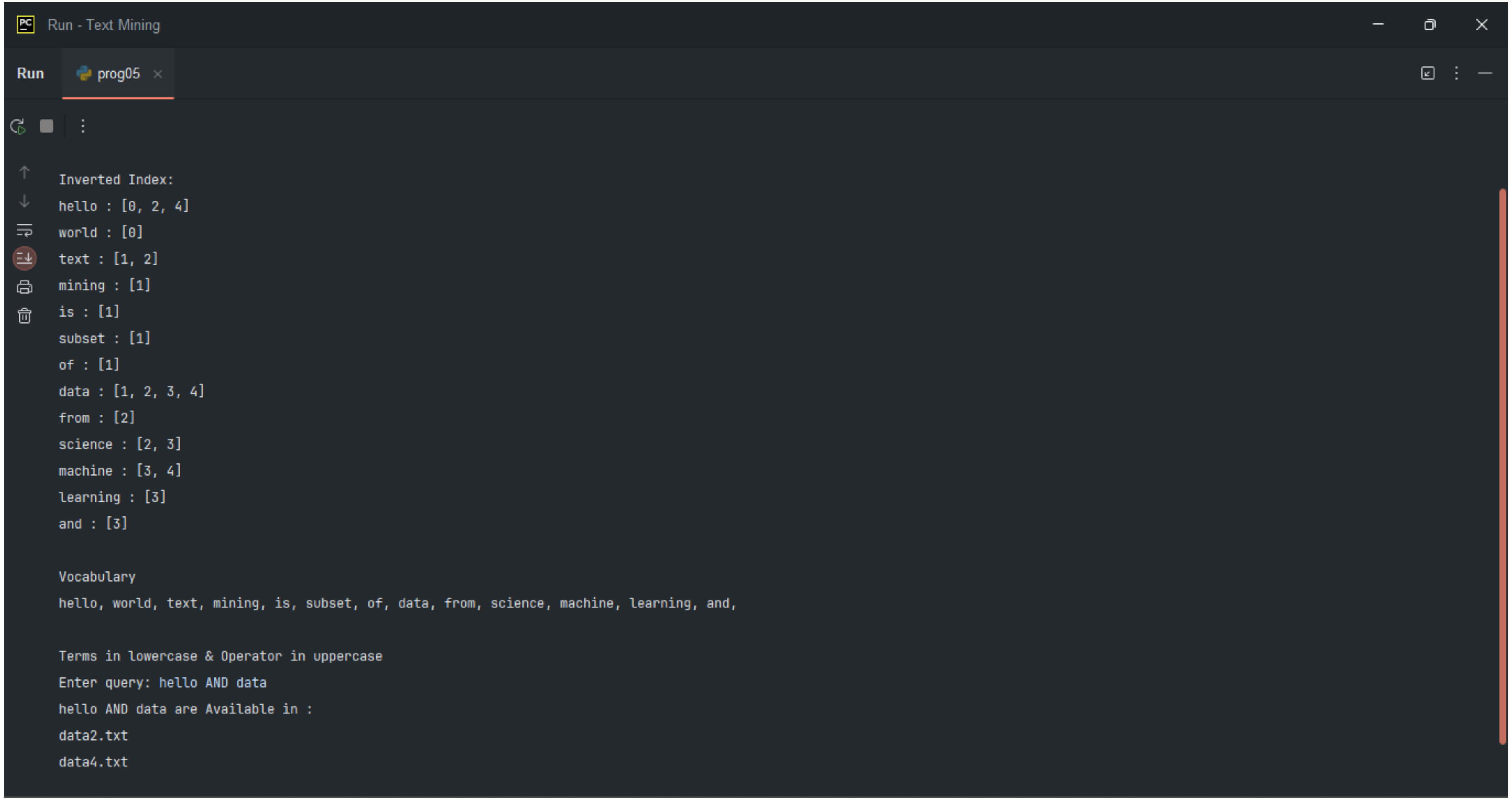
Query = hello OR text



```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe" "C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\Lab\prog04.py"

Vocabulary:
['hello', 'world', 'text', 'mining', 'is', 'subset', 'of', 'data', 'from', 'science', 'machine', 'learning', 'and']

Terms in lowercase & Operator in uppercase
Enter query: NOT hello

hello is NOT Available in :
data1.txt
data3.txt

Process finished with exit code 0
```

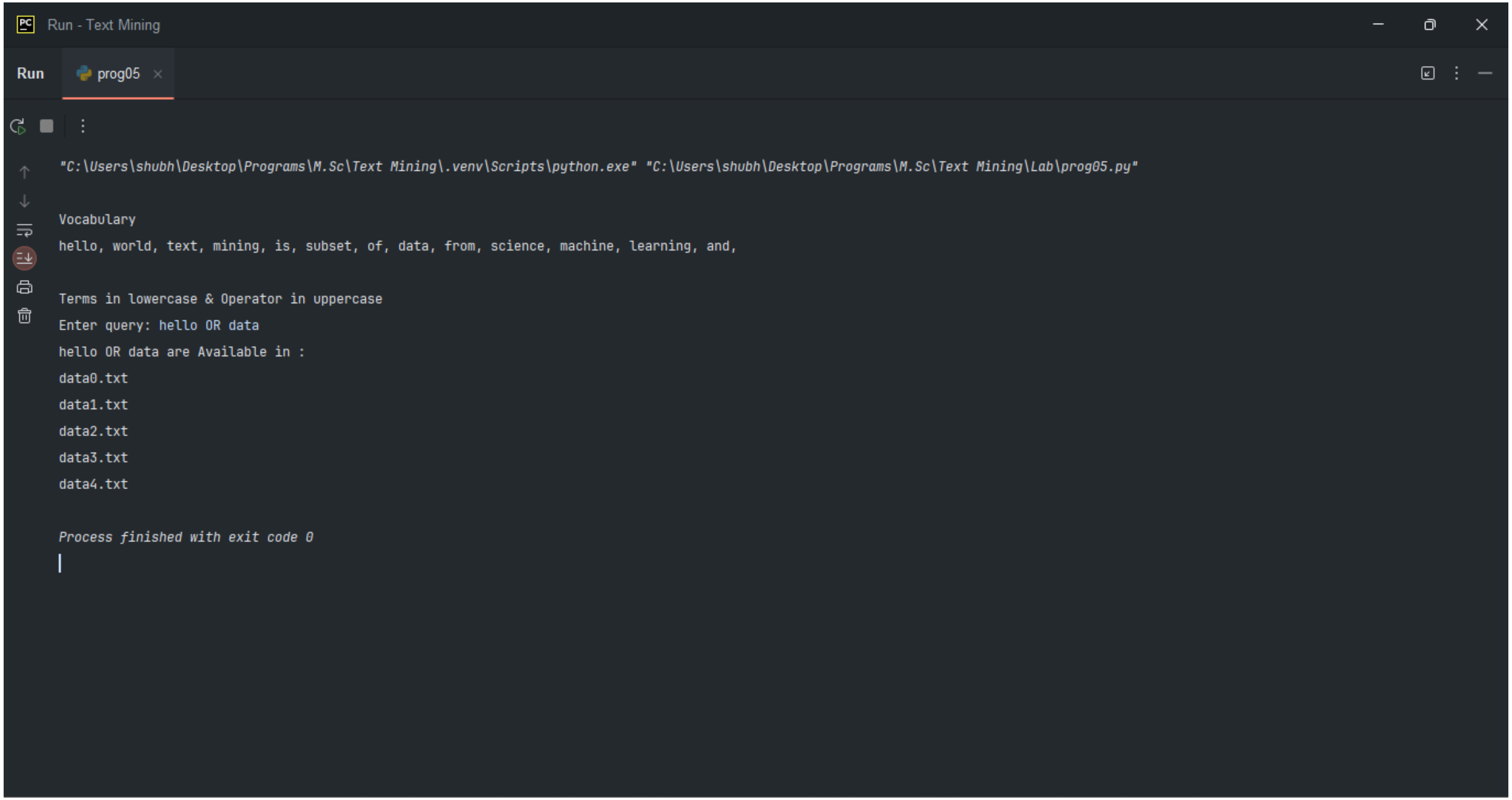
Query = NOT hello

Program-05 : Implement Inverted-Index for boolean retrieval.



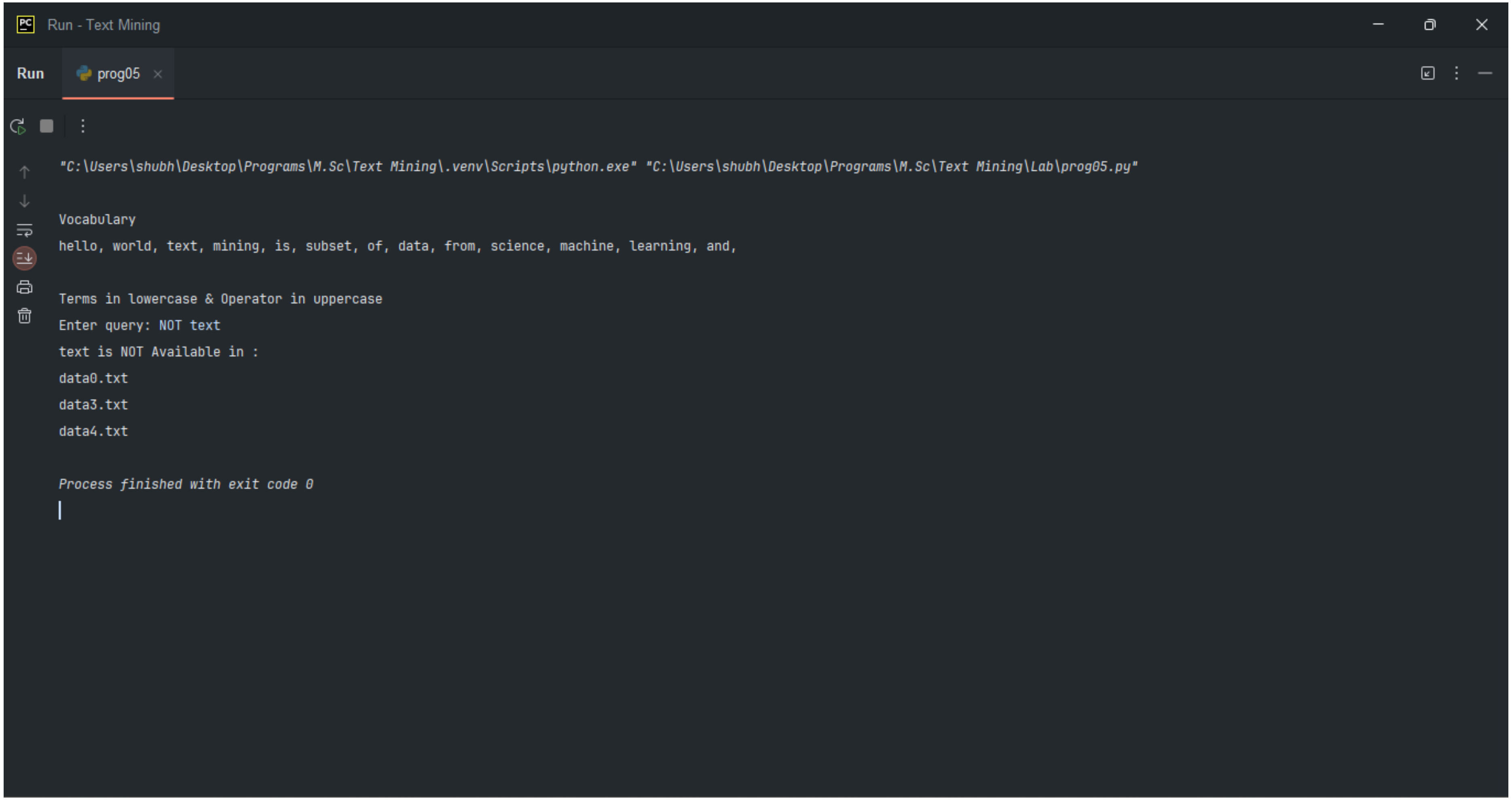
```
Inverted Index:
hello : [0, 2, 4]
world : [0]
text : [1, 2]
mining : [1]
is : [1]
subset : [1]
of : [1]
data : [1, 2, 3, 4]
from : [2]
science : [2, 3]
machine : [3, 4]
learning : [3]
and : [3]

Vocabulary
hello, world, text, mining, is, subset, of, data, from, science, machine, learning, and,

Terms in lowercase & Operator in uppercase
Enter query: hello AND data
hello AND data are Available in :
data2.txt
data4.txt
```

Query = hello AND data



```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe" "C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\Lab\prog05.py"

Vocabulary
hello, world, text, mining, is, subset, of, data, from, science, machine, learning, and,

Terms in lowercase & Operator in uppercase
Enter query: hello OR data
hello OR data are Available in :
data0.txt
data1.txt
data2.txt
data3.txt
data4.txt

Process finished with exit code 0
```

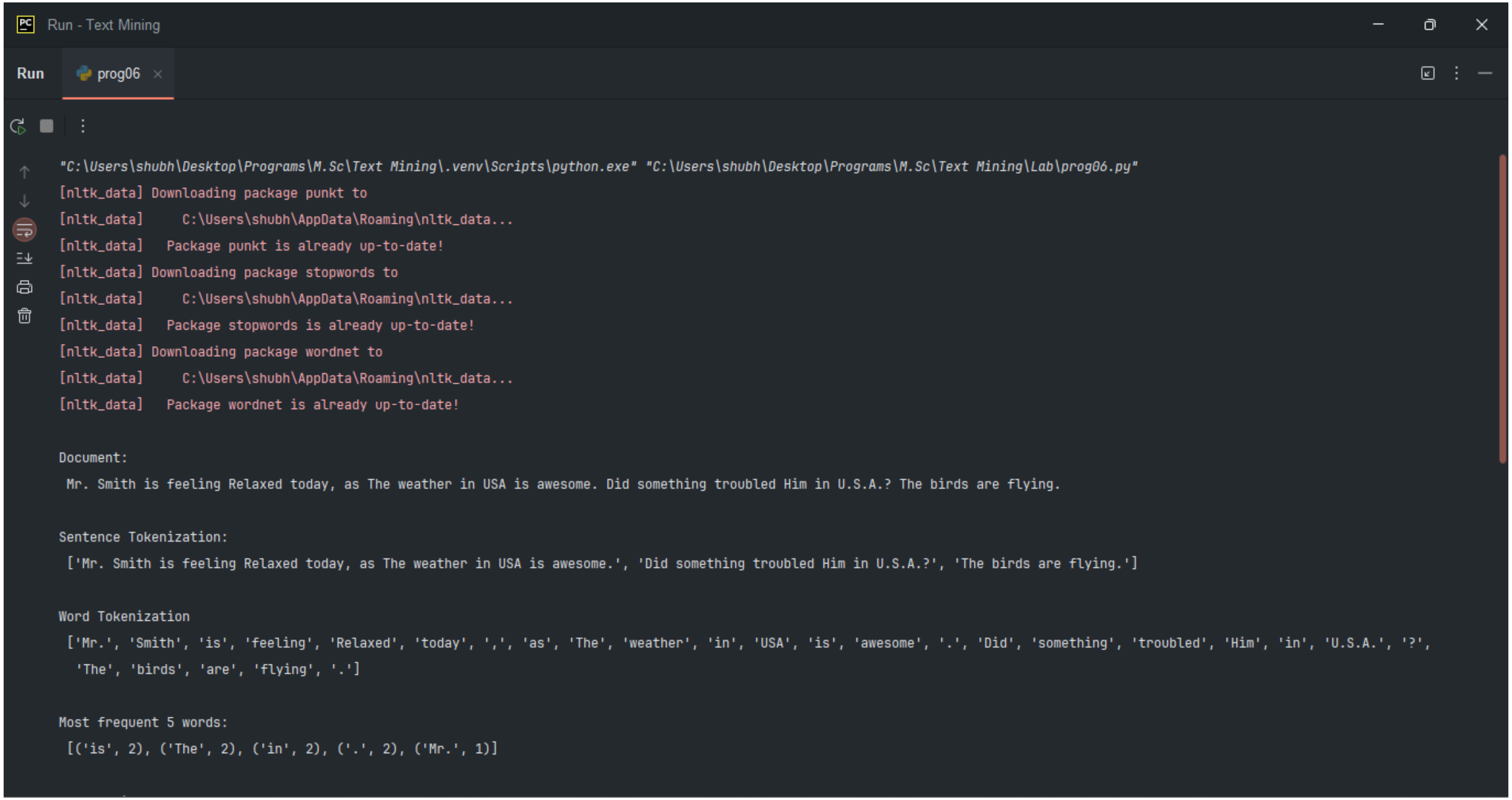
Query = hello OR data



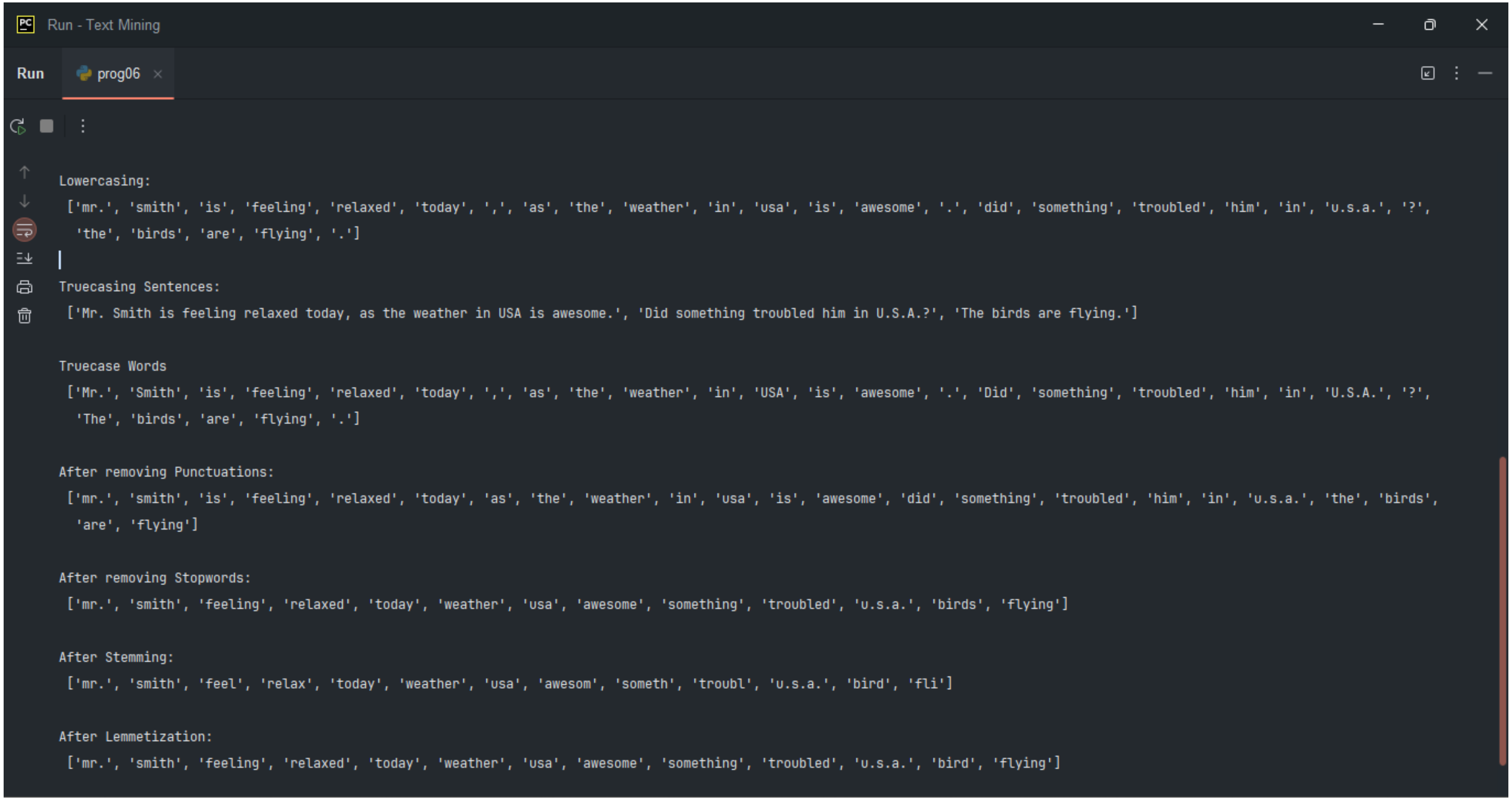
```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe" "C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\Lab\prog05.py"

Vocabulary
hello, world, text, mining, is, subset, of, data, from, science, machine, learning, and,

Terms in lowercase & Operator in uppercase
Enter query: NOT text
text is NOT Available in :
data0.txt
data3.txt
data4.txt

Process finished with exit code 0
```

Query = NOT text

Program-06 : Using NLTK perform Tokenization, Normalization, Stemming & Lemmetization.

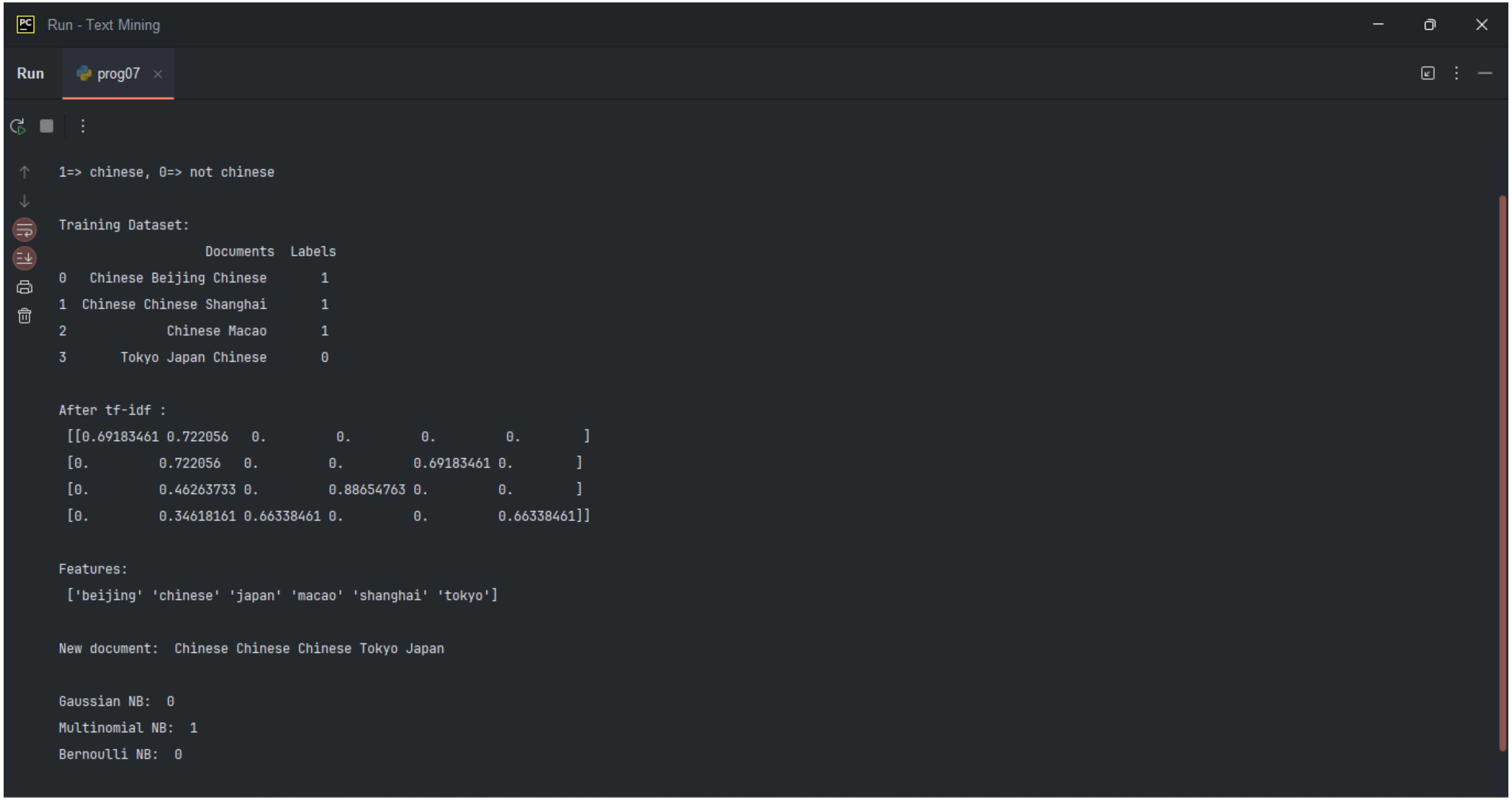


```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe" "C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\Lab\prog06.py"
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\shubh\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\shubh\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\shubh\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!

Document:
 Mr. Smith is feeling Relaxed today, as The weather in USA is awesome. Did something troubled Him in U.S.A.? The birds are flying.

Sentence Tokenization:
 ['Mr. Smith is feeling Relaxed today, as The weather in USA is awesome.', 'Did something troubled Him in U.S.A.?', 'The birds are flying.']

Word Tokenization
 ['Mr.', 'Smith', 'is', 'feeling', 'Relaxed', 'today', ',', 'as', 'The', 'weather', 'in', 'USA', 'is', 'awesome', '.', 'Did', 'something', 'troubled', 'Him', 'in', 'U.S.A.', '?',
  'The', 'birds', 'are', 'flying', '.']

Most frequent 5 words:
 [('is', 2), ('The', 2), ('in', 2), ('.', 2), ('Mr.', 1)]
```



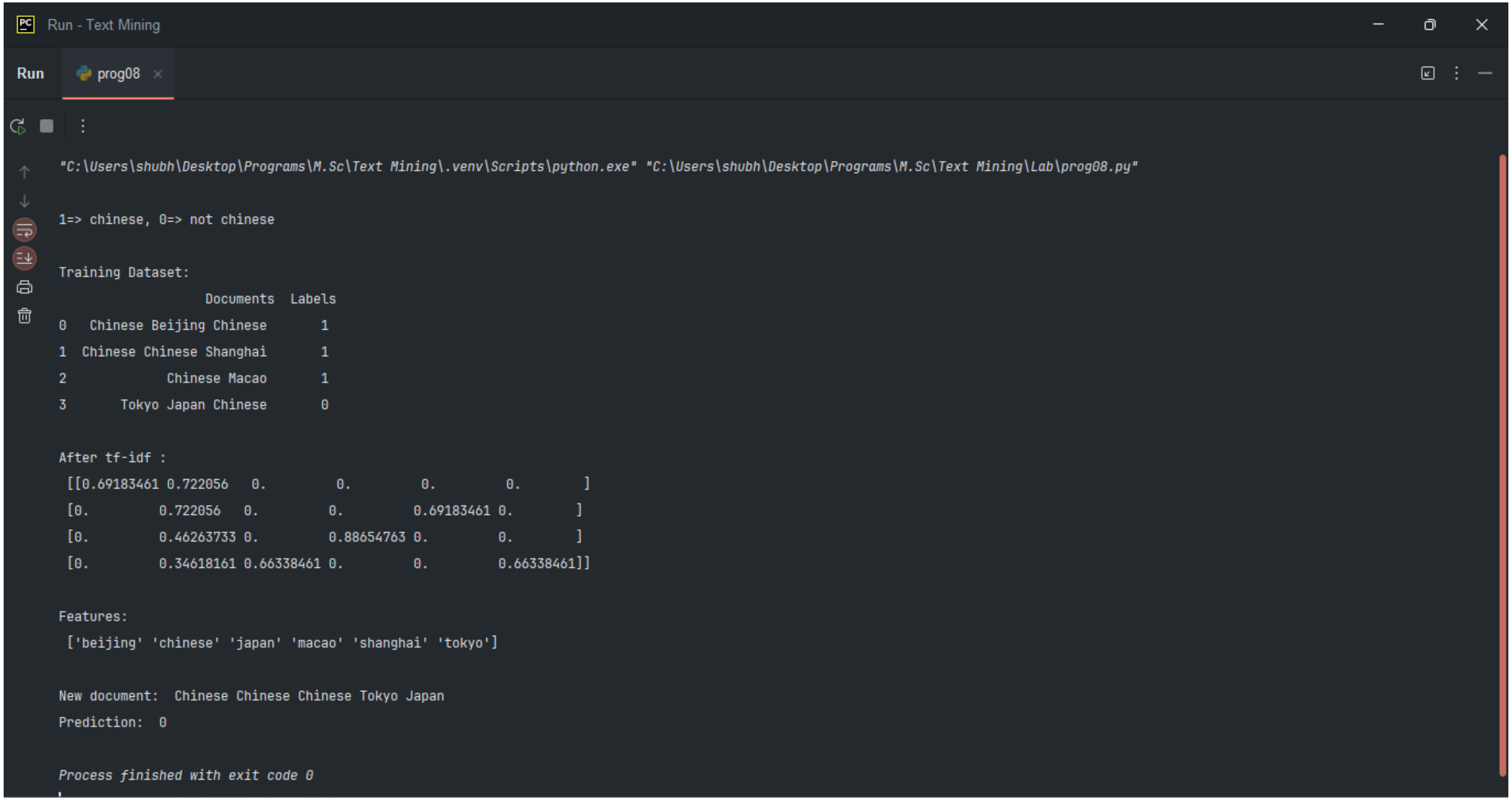
```
Lowercasing:
 ['mr.', 'smith', 'is', 'feeling', 'relaxed', 'today', ',', 'as', 'the', 'weather', 'in', 'usa', 'is', 'awesome', '.', 'did', 'something', 'troubled', 'him', 'in', 'u.s.a.', '?',
  'the', 'birds', 'are', 'flying', '.']

Truecasing Sentences:
 ['Mr. Smith is feeling relaxed today, as the weather in USA is awesome.', 'Did something troubled him in U.S.A.?', 'The birds are flying.']

Truecase Words
 ['Mr.', 'Smith', 'is', 'feeling', 'relaxed', 'today', ',', 'as', 'the', 'weather', 'in', 'USA', 'is', 'awesome', '.', 'Did', 'something', 'troubled', 'him', 'in', 'U.S.A.', '?',
  'The', 'birds', 'are', 'flying', '.']

After removing Punctuations:
 ['mr.', 'smith', 'is', 'feeling', 'relaxed', 'today', 'as', 'the', 'weather', 'in', 'usa', 'is', 'awesome', 'did', 'something', 'troubled', 'him', 'in', 'u.s.a.', 'the', 'birds',
  'are', 'flying']

After removing Stopwords:
 ['mr.', 'smith', 'feeling', 'relaxed', 'today', 'weather', 'usa', 'awesome', 'something', 'troubled', 'u.s.a.', 'birds', 'flying']

After Stemming:
 ['mr.', 'smith', 'feel', 'relax', 'today', 'weather', 'usa', 'awesom', 'someth', 'troubl', 'u.s.a.', 'bird', 'fli']

After Lemmetization:
 ['mr.', 'smith', 'feeling', 'relaxed', 'today', 'weather', 'usa', 'awesome', 'something', 'troubled', 'u.s.a.', 'bird', 'flying']
```

Program-07 : Naive Bayes classification using scikit-learn.



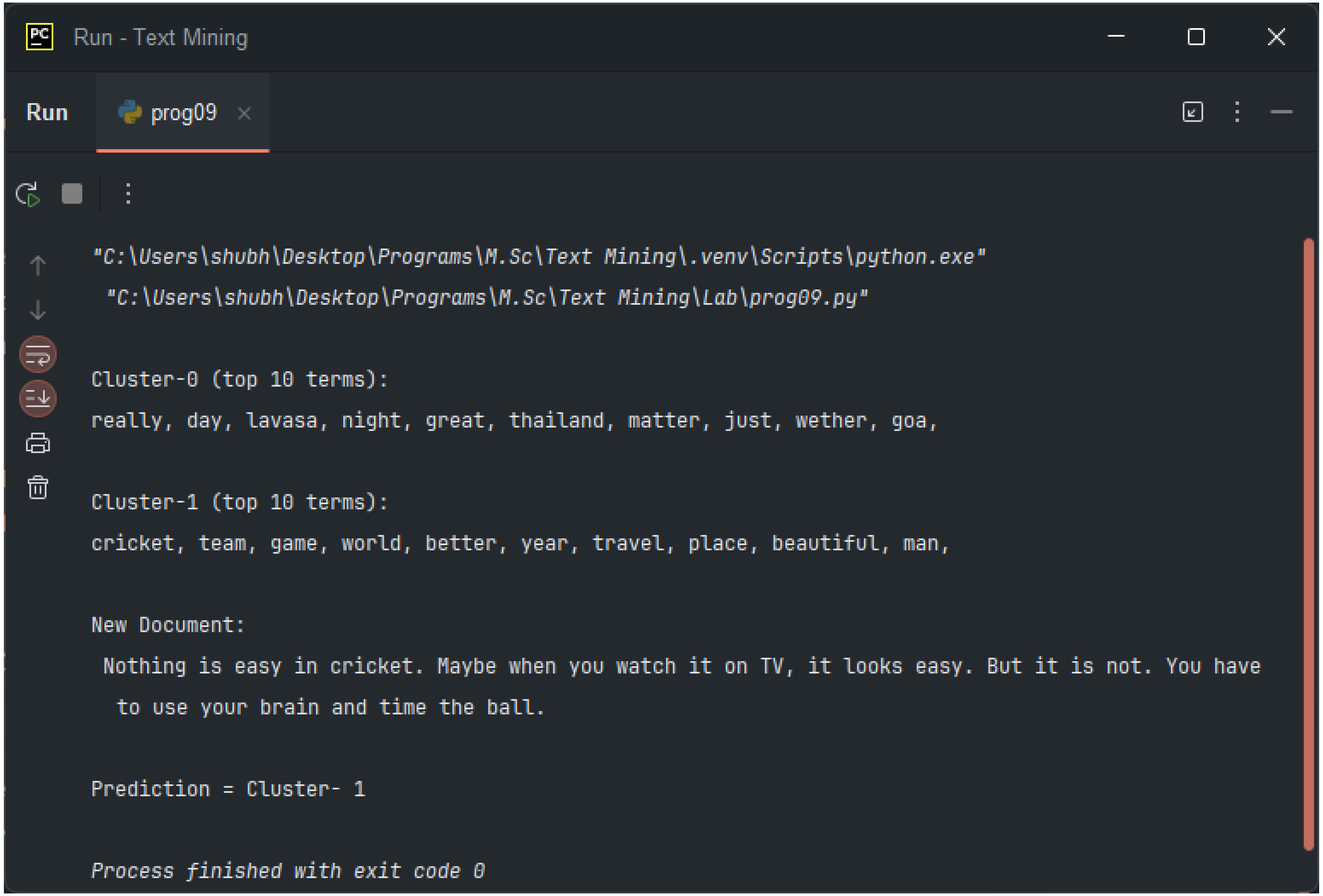
```
1=> chinese, 0=> not chinese

Training Dataset:
                  Documents  Labels
0   Chinese Beijing Chinese       1
1  Chinese Chinese Shanghai       1
2             Chinese Macao       1
3       Tokyo Japan Chinese       0

After tf-idf :
 [[0.69183461 0.722056   0.         0.         0.         0.        ]
 [0.         0.722056   0.         0.         0.69183461 0.        ]
 [0.         0.46263733 0.         0.88654763 0.         0.        ]
 [0.         0.34618161 0.66338461 0.         0.         0.66338461]]

Features:
 ['beijing' 'chinese' 'japan' 'macao' 'shanghai' 'tokyo']

New document:  Chinese Chinese Chinese Tokyo Japan

Gaussian NB:  0
Multinomial NB:  1
Bernoulli NB:  0
```

Program-08 : Rocchio classification using scikit-learn.



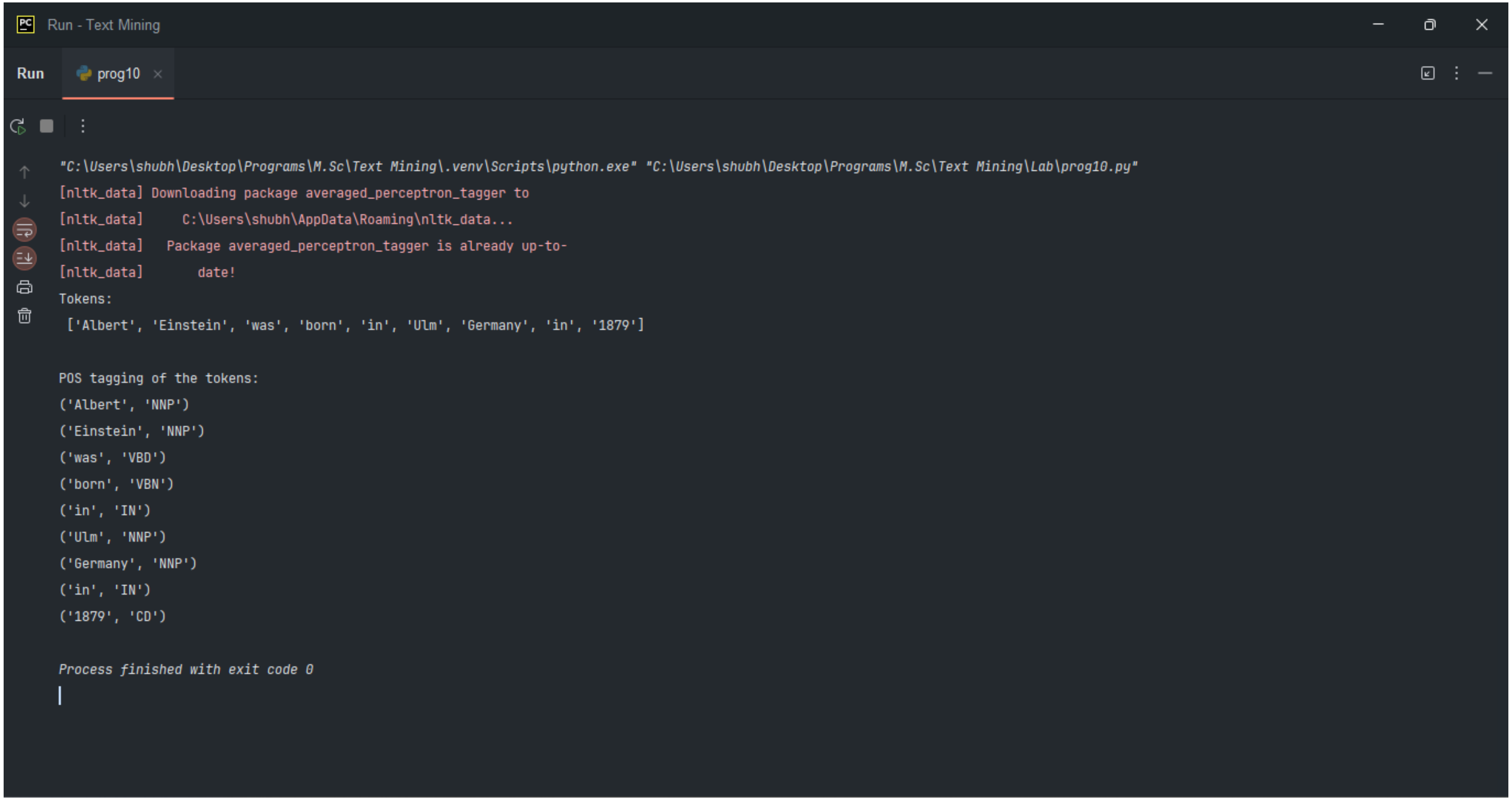
```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe" "C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\Lab\prog08.py"

1=> chinese, 0=> not chinese

Training Dataset:
                   Documents  Labels
0   Chinese Beijing Chinese       1
1  Chinese Chinese Shanghai       1
2             Chinese Macao       1
3       Tokyo Japan Chinese       0

After tf-idf :
 [[0.69183461 0.722056   0.         0.         0.         0.        ]
 [0.         0.722056   0.         0.         0.69183461 0.        ]
 [0.         0.46263733 0.         0.88654763 0.         0.        ]
 [0.         0.34618161 0.66338461 0.         0.         0.66338461]]

Features:
 ['beijing' 'chinese' 'japan' 'macao' 'shanghai' 'tokyo']

New document:  Chinese Chinese Chinese Tokyo Japan
Prediction:  0

Process finished with exit code 0
```

Program-09 : k-means Clustering using scikit-learn.



```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe"
 "C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\Lab\prog09.py"

Cluster-0 (top 10 terms):
really, day, lavasa, night, great, thailand, matter, just, wether, goa,

Cluster-1 (top 10 terms):
cricket, team, game, world, better, year, travel, place, beautiful, man,

New Document:
 Nothing is easy in cricket. Maybe when you watch it on TV, it looks easy. But it is not. You have
  to use your brain and time the ball.

Prediction = Cluster- 1

Process finished with exit code 0
```

Program-10 : Perform Parts-Of-Speech (POS) Tagging using NLTK.



```
"C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\.venv\Scripts\python.exe" "C:\Users\shubh\Desktop\Programs\M.Sc\Text Mining\Lab\prog10.py"
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     C:\Users\shubh\AppData\Roaming\nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
Tokens:
 ['Albert', 'Einstein', 'was', 'born', 'in', 'Ulm', 'Germany', 'in', '1879']

POS tagging of the tokens:
('Albert', 'NNP')
('Einstein', 'NNP')
('was', 'VBD')
('born', 'VBN')
('in', 'IN')
('Ulm', 'NNP')
('Germany', 'NNP')
('in', 'IN')
('1879', 'CD')

Process finished with exit code 0
```