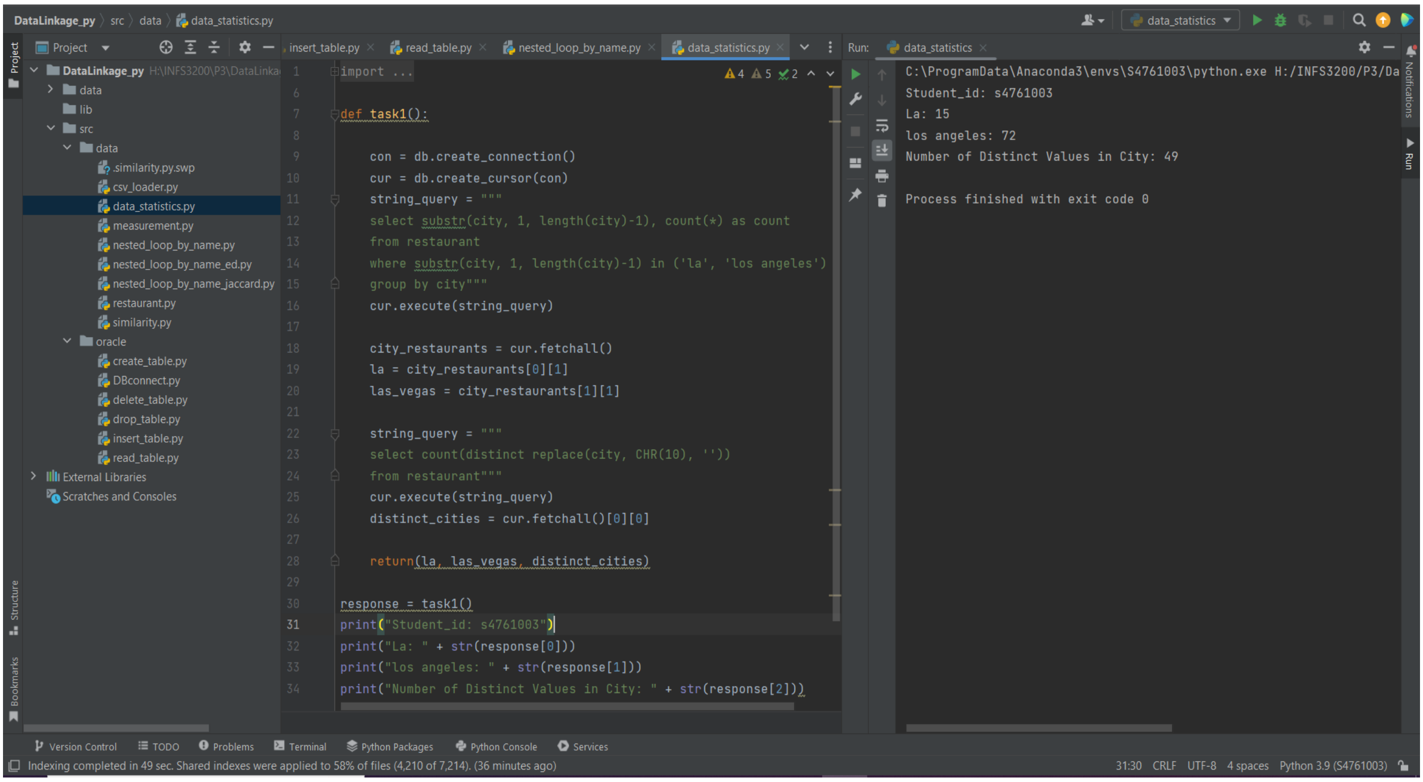
TASK 1

data_statistics.py

```
import src.oracle.DBconnect as db
from src.data.restaurant import restaurant as res
import datetime
import src.data.csv_loader as csv
import src.data.measurement as measure

def task1():

    con = db.create_connection()
    cur = db.create_cursor(con)
    string_query = """
    select substr(city, 1, length(city)-1), count(*) as count
    from restaurant
    where substr(city, 1, length(city)-1) in ('la', 'los angeles')
    group by city"""
    cur.execute(string_query)

    city_restaurants = cur.fetchall()
    la = city_restaurants[0][1]
    las_vegas = city_restaurants[1][1]

    string_query = """
    select count(distinct replace(city, CHR(10), ''))
    from restaurant"""
    cur.execute(string_query)
    distinct_cities = cur.fetchall()[0][0]

    return(la, las_vegas, distinct_cities)

response = task1()
print("Student_id: s4761003")
print("La: " + str(response[0]))
print("los angeles: " + str(response[1]))
print("Number of Distinct Values in City: " + str(response[2]))
```

Output



```
import ...

def task1():

    con = db.create_connection()
    cur = db.create_cursor(con)
    string_query = """
    select substr(city, 1, length(city)-1), count(*) as count
    from restaurant
    where substr(city, 1, length(city)-1) in ('la', 'los angeles')
    group by city"""
    cur.execute(string_query)

    city_restaurants = cur.fetchall()
    la = city_restaurants[0][1]
    las_vegas = city_restaurants[1][1]

    string_query = """
    select count(distinct replace(city, CHR(10), ''))
    from restaurant"""
    cur.execute(string_query)
    distinct_cities = cur.fetchall()[0][0]

    return(la, las_vegas, distinct_cities)

response = task1()
print("Student_id: s4761003")
print("La: " + str(response[0]))
print("los angeles: " + str(response[1]))
print("Number of Distinct Values in City: " + str(response[2]))
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/Da
Student_id: s4761003
La: 15
los angeles: 72
Number of Distinct Values in City: 49

Process finished with exit code 0
```

Indexing completed in 49 sec. Shared indexes were applied to 58% of files (4,210 of 7,214). (36 minutes ago)

TASK 2

In measurement.py,

precision = count / len(results)

recall = count / len(benchmark)

$$Precision = \frac{tp}{tp+fp} \qquad Recall = \frac{tp}{tp+fn}$$

- **count** defines the id pair of the restaurants which are correctly identified as the duplicates according to both the algorithm and the given gold standard file. Hence count is defined as the true positive which had been identified correctly by the algorithm.
- **len(results) or results.size()** is the size of the list of restaurant id pairs which have been identified by the algorithm as the duplicate records. It is the list size which contains the duplicate records that have been correctly and incorrectly classified by the algorithm. Hence it is the sum of the true positive and false positive.
- **len(benchmark) or benchmark.size()** defines the size of the list of the pairs of restaurant ids that have been given as the list of restaurants by the external gold-standard. It is the sum of the sizes of the list of restaurants correctly identified by the algorithm and the size of the list of restaurants present only in the gold standard not identified by the algorithm but is present in the gold standard. Hence it is the sum of the true positive and false negative.

True Positive: count = 75. The restaurant id pair belongs to both the algorithm and the golden standard file.

False Positive: len(results) – count = 7. The restaurant id pair belongs to the algorithm but not present in golden standard file.

True Negative: cannot be calculated as the id pair will not appear in both the algorithm and the golden standard file.

False Negative: len(benchmark) – count = 31. The restaurant id pair belongs to the golden standard file but not by the algorithm.

The **true positive rate**, also called sensitivity is calculated as **TP/TP+FN**. TPR is the probability that an actual positive will test positive. This shows the ratio of restaurant id pairs correctly identified as when compared to the golden standard file.

The **false negative rate** – also called the miss rate – is the probability that a true positive will be missed by the test. It's calculated as **FN/FN+TP.** This is also 1 – (true positive rate). This shows the ratio of restaurant id pairs incorrectly identified as when compared to the golden standard file.

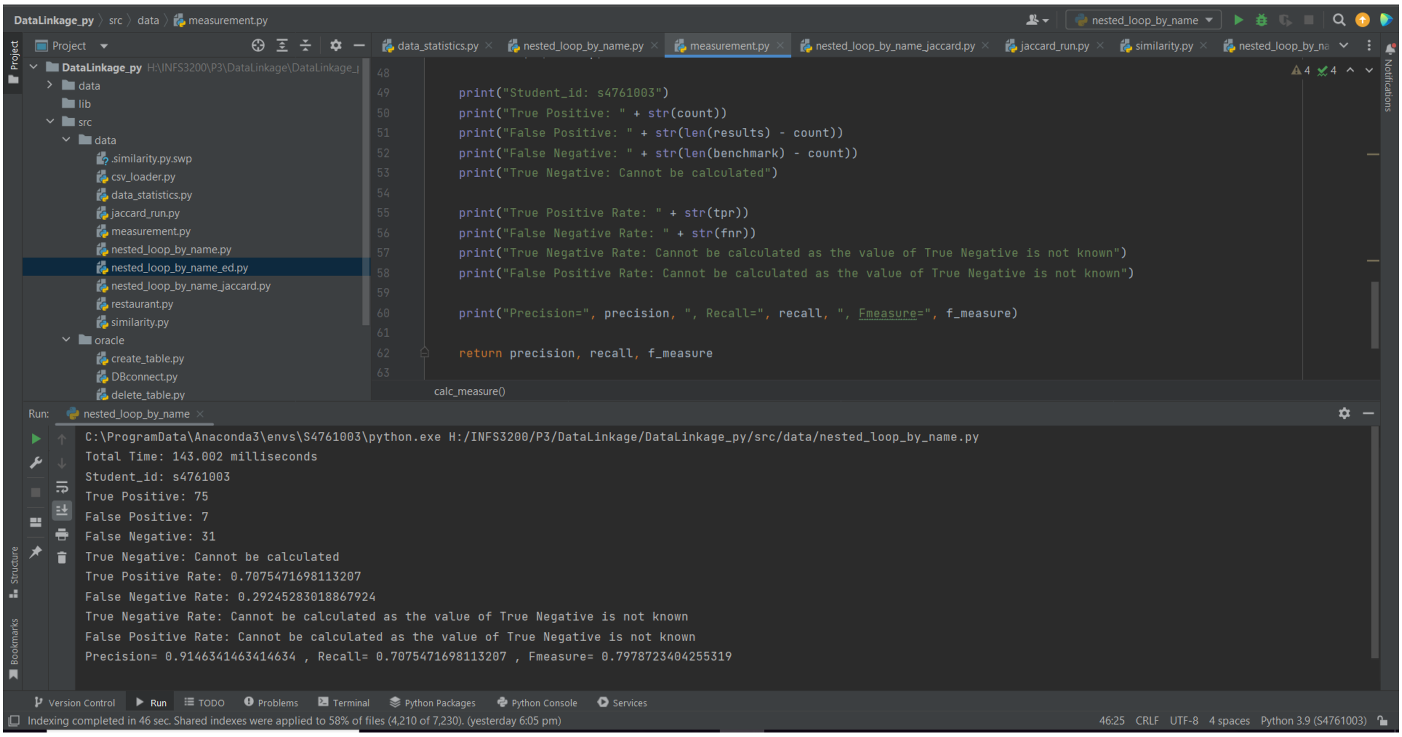
The **false positive rate** is calculated as FP/FP+TN, where FP is the number of false positives and TN is the number of true negatives (FP+TN being the total number of negatives).

The **true negative rate** (also called specificity), is calculated as **TN/TN+FP**. The ratio of the restaurants id pairs which are not in both algorithm and golden standard file to the restaurant id pairs present in neither algorithm and golden standard file and restaurant ids belonging to the algorithm but not present in golden standard file.

Precision is the ratio of true positives to the total of the true positives and false positives. It is the ratio of the number of restaurant id pairs present in both algorithm and golden standard file to the number of restaurants id pairs present in the in the list of algorithm.

Recall is the ratio of true positives to the total of the true positives and false negatives. It is the ratio of the number of restaurant id pairs present in both algorithm and golden standard file to the number of restaurants id pairs present in the in the list of golden standard file.

OUTPUT -



```
    print("Student_id: s4761003")
    print("True Positive: " + str(count))
    print("False Positive: " + str(len(results) - count))
    print("False Negative: " + str(len(benchmark) - count))
    print("True Negative: Cannot be calculated")

    print("True Positive Rate: " + str(tpr))
    print("False Negative Rate: " + str(fnr))
    print("True Negative Rate: Cannot be calculated as the value of True Negative is not known")
    print("False Positive Rate: Cannot be calculated as the value of True Negative is not known")

    print("Precision=", precision, ", Recall=", recall, ", Fmeasure=", f_measure)

    return precision, recall, f_measure
```

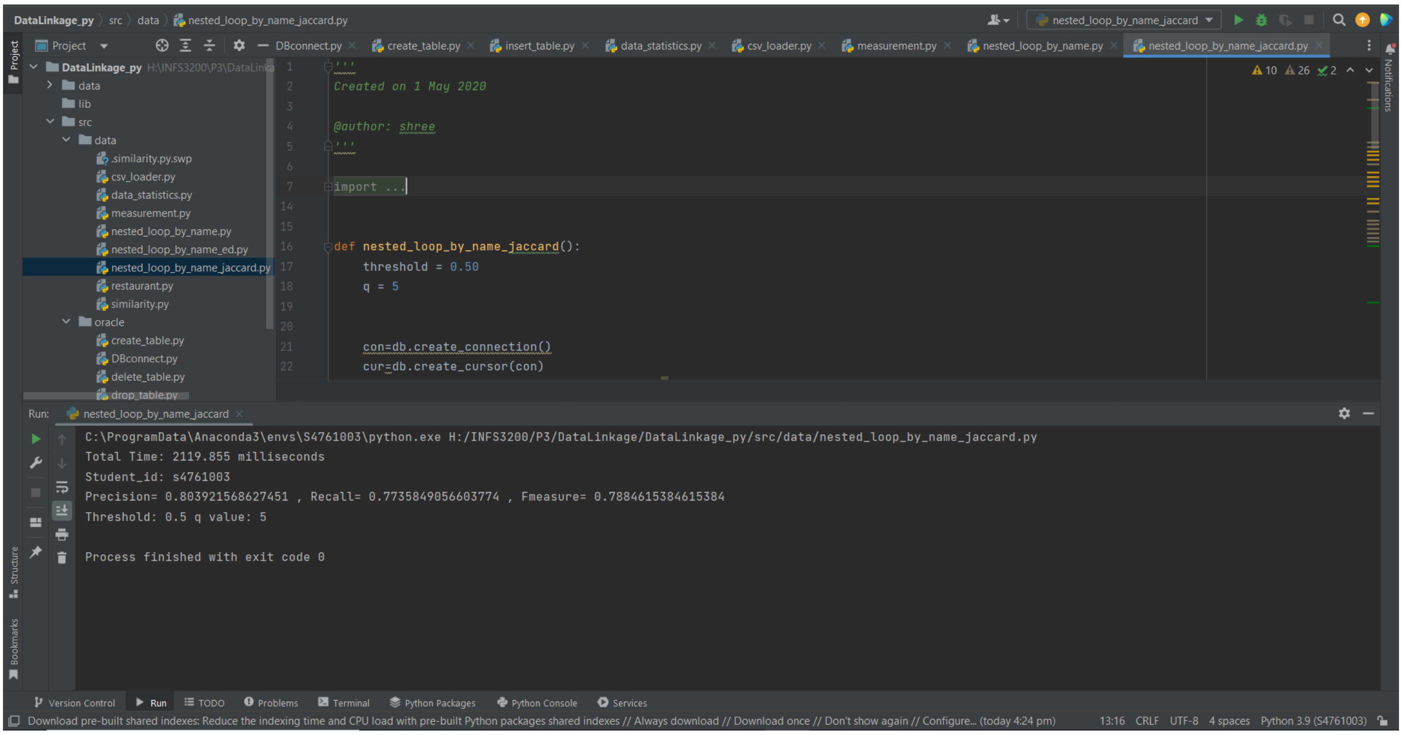
```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name.py
Total Time: 143.002 milliseconds
Student_id: s4761003
True Positive: 75
False Positive: 7
False Negative: 31
True Negative: Cannot be calculated
True Positive Rate: 0.7075471698113207
False Negative Rate: 0.29245283018867924
True Negative Rate: Cannot be calculated as the value of True Negative is not known
False Positive Rate: Cannot be calculated as the value of True Negative is not known
Precision= 0.9146341463414634 , Recall= 0.7075471698113207 , Fmeasure= 0.7978723404255319
```

Indexing completed in 46 sec. Shared indexes were applied to 58% of files (4,210 of 7,230). (yesterday 6:05 pm)

To calculate True Negative, we have to implement the algorithm to calculate the potential pairs of matched restaurant ids. Once done we have to compare with the golden standard file and if the id pair is not found in the standard, we can assume it to be a part of true negative.

TASK 3

q = 5, threshold = 0.5

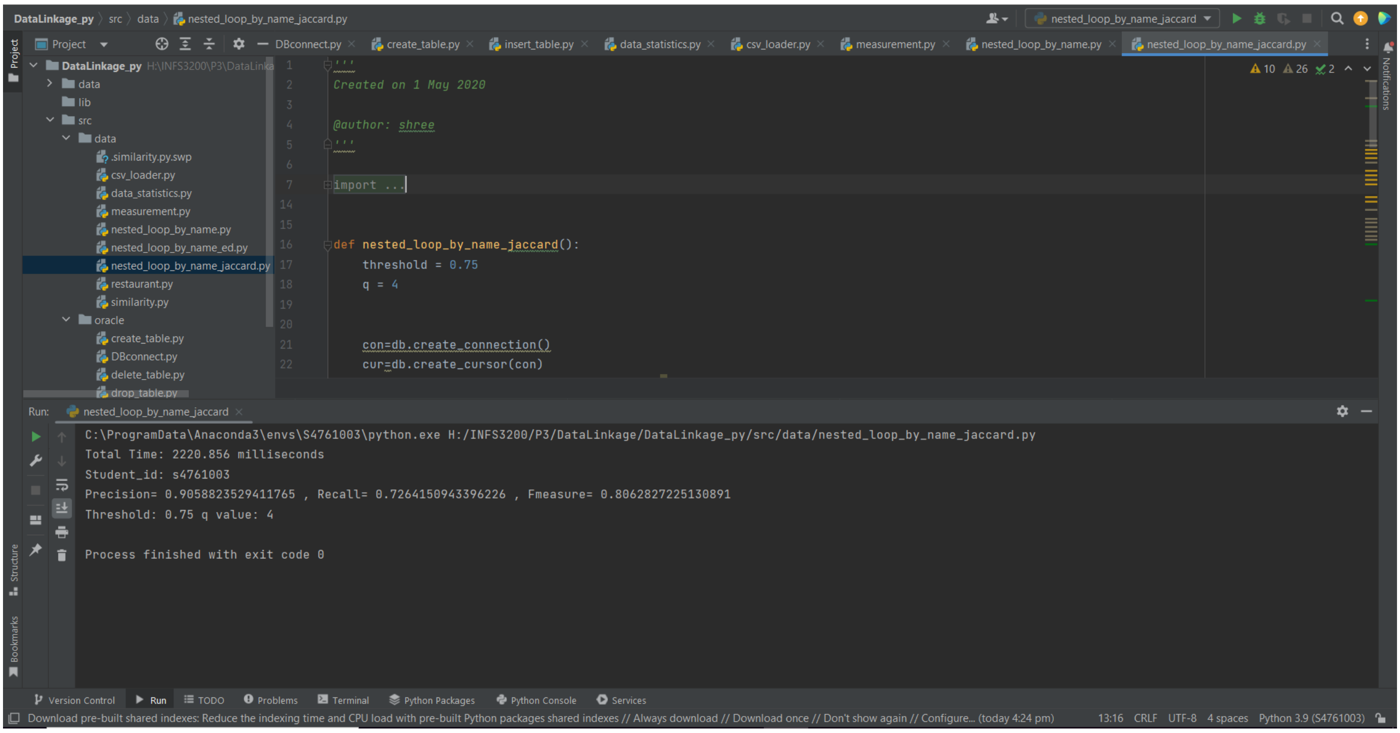


```
'''
Created on 1 May 2020

@author: shree
'''

import ...


def nested_loop_by_name_jaccard():
    threshold = 0.50
    q = 5


    con=db.create_connection()
    cur=db.create_cursor(con)
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_jaccard.py
Total Time: 2119.855 milliseconds
Student_id: s4761003
Precision= 0.803921568627451 , Recall= 0.7735849056603774 , Fmeasure= 0.7884615384615384
Threshold: 0.5 q value: 5

Process finished with exit code 0
```

Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once // Don't show again // Configure... (today 4:24 pm)

q = 4, threshold = 0.75

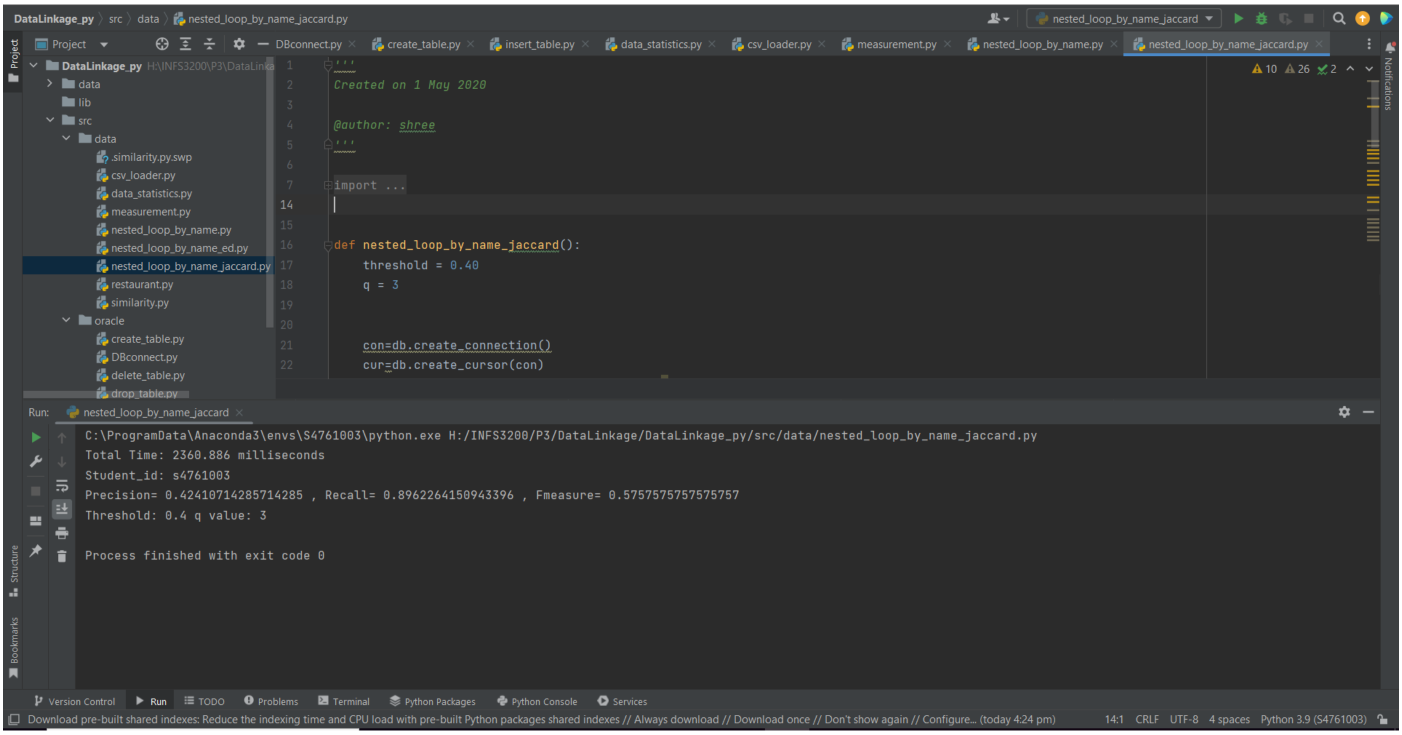


```
'''
Created on 1 May 2020

@author: shree
'''

import ...


def nested_loop_by_name_jaccard():
    threshold = 0.75
    q = 4


    con=db.create_connection()
    cur=db.create_cursor(con)
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_jaccard.py
Total Time: 2220.856 milliseconds
Student_id: s4761003
Precision= 0.9058823529411765 , Recall= 0.7264150943396226 , Fmeasure= 0.8062827225130891
Threshold: 0.75 q value: 4

Process finished with exit code 0
```

Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once // Don't show again // Configure... (today 4:24 pm)

q = 3, threshold = 0.4

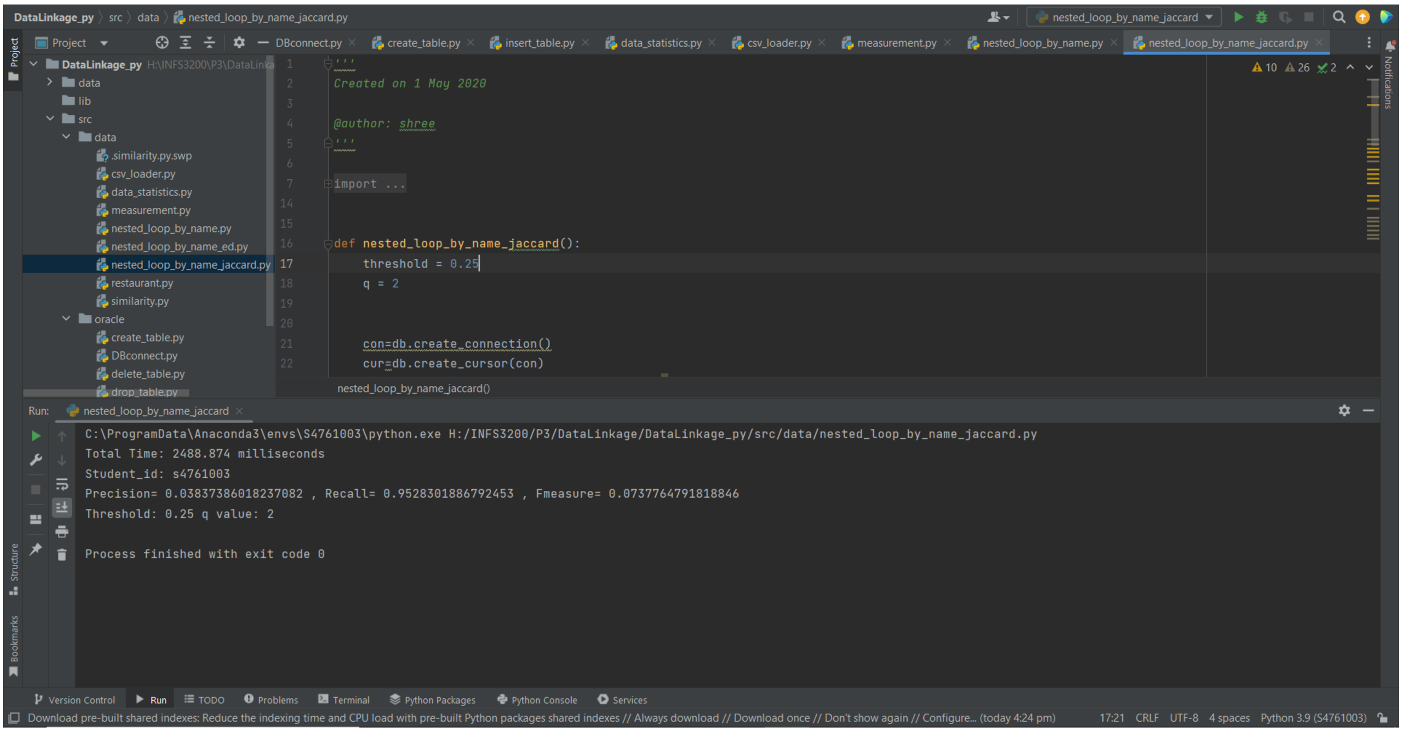


```
'''
Created on 1 May 2020

@author: shree
'''

import ...


def nested_loop_by_name_jaccard():
    threshold = 0.40
    q = 3


    con=db.create_connection()
    cur=db.create_cursor(con)
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_jaccard.py
Total Time: 2360.886 milliseconds
Student_id: s4761003
Precision= 0.42410714285714285 , Recall= 0.8962264150943396 , Fmeasure= 0.5757575757575757
Threshold: 0.4 q value: 3

Process finished with exit code 0
```

Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once // Don't show again // Configure... (today 4:24 pm)

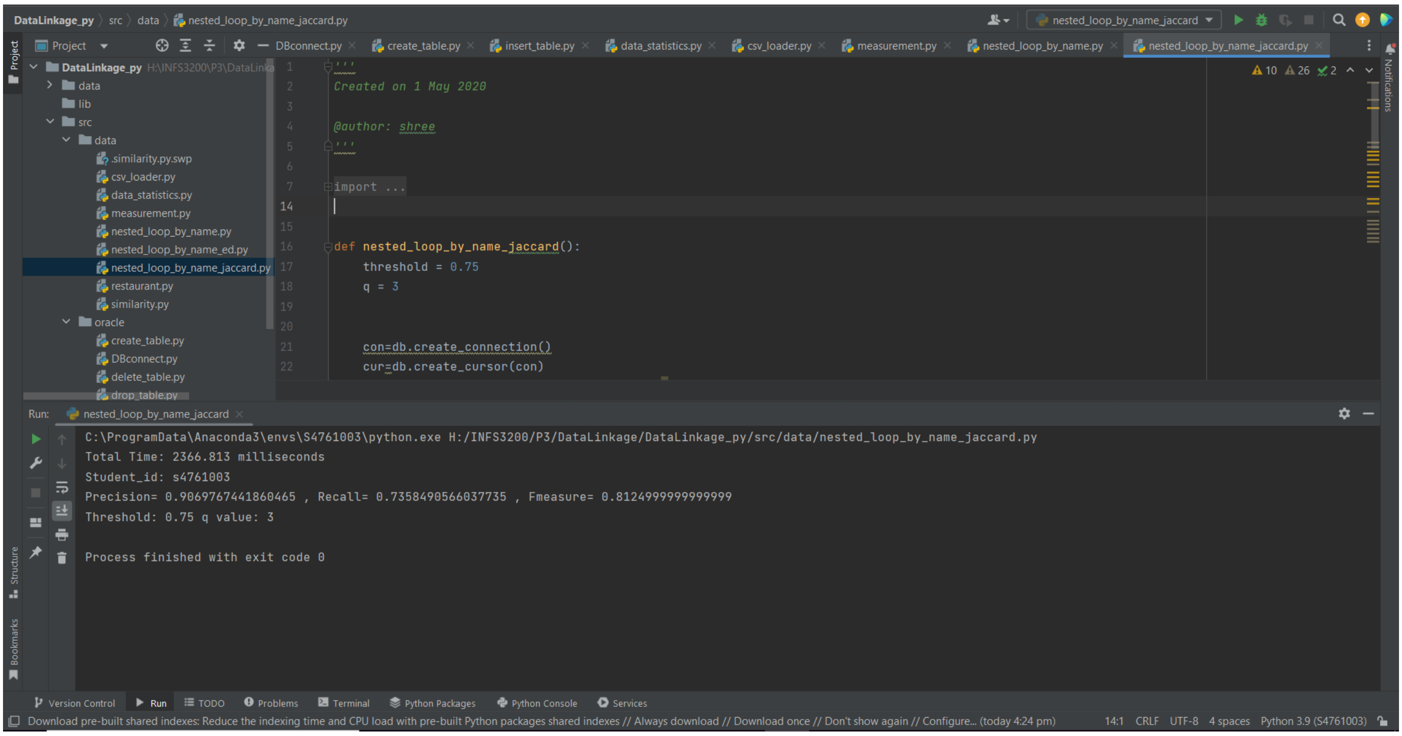
q = 2, threshold = 0.25


DataLinkage_py 〉 src 〉 data 〉 nested_loop_by_name_jaccard.py
nested_loop_by_name_jaccard
Project
DBconnect.py
create_table.py
insert_table.py
data_statistics.py
csv_loader.py
measurement.py
nested_loop_by_name.py
nested_loop_by_name_jaccard.py
DataLinkage_py H:\INFS3200\P3\DataLink
data
lib
src
data
.similarity.py.swp
csv_loader.py
data_statistics.py
measurement.py
nested_loop_by_name.py
nested_loop_by_name_ed.py
nested_loop_by_name_jaccard.py
restaurant.py
similarity.py
oracle
create_table.py
DBconnect.py
delete_table.py
drop_table.py
Run:
Version Control
Run
TODO
Problems
Terminal
Python Packages
Python Console
Services
Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once // Don't show again // Configure... (today 4:24 pm)
17:21 CRLF UTF-8 4 spaces Python 3.9 (S4761003)

```
'''
Created on 1 May 2020

@author: shree
'''

import ...


def nested_loop_by_name_jaccard():
    threshold = 0.25
    q = 2


    con=db.create_connection()
    cur=db.create_cursor(con)
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_jaccard.py
Total Time: 2488.874 milliseconds
Student_id: s4761003
Precision= 0.03837386018237082 , Recall= 0.9528301886792453 , Fmeasure= 0.0737764791818846
Threshold: 0.25 q value: 2

Process finished with exit code 0
```

q = 3, threshold = 0.75


DataLinkage_py 〉 src 〉 data 〉 nested_loop_by_name_jaccard.py
nested_loop_by_name_jaccard
Project
DBconnect.py
create_table.py
insert_table.py
data_statistics.py
csv_loader.py
measurement.py
nested_loop_by_name.py
nested_loop_by_name_jaccard.py
DataLinkage_py H:\INFS3200\P3\DataLink
data
lib
src
data
.similarity.py.swp
csv_loader.py
data_statistics.py
measurement.py
nested_loop_by_name.py
nested_loop_by_name_ed.py
nested_loop_by_name_jaccard.py
restaurant.py
similarity.py
oracle
create_table.py
DBconnect.py
delete_table.py
drop_table.py
Run:
Version Control
Run
TODO
Problems
Terminal
Python Packages
Python Console
Services
Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once // Don't show again // Configure... (today 4:24 pm)
14:1 CRLF UTF-8 4 spaces Python 3.9 (S4761003)

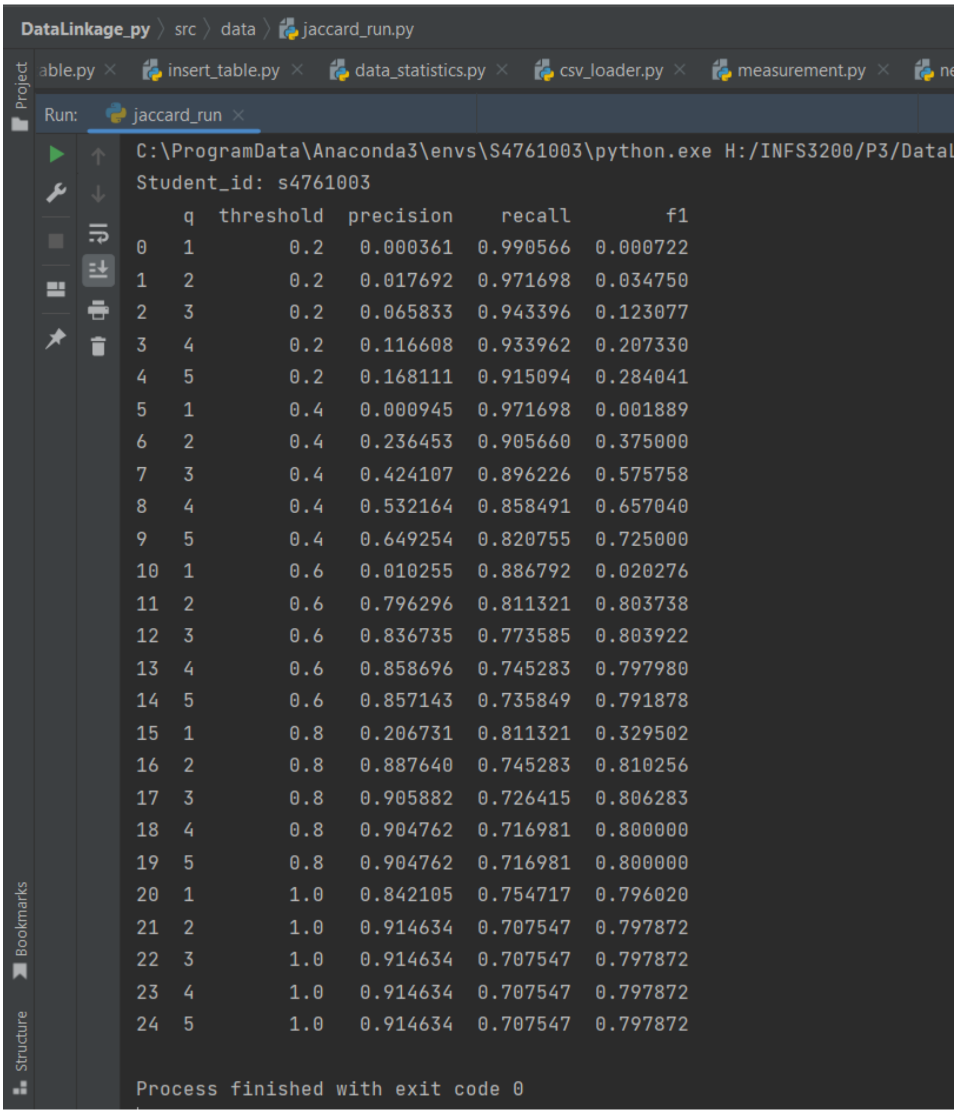
```
'''
Created on 1 May 2020

@author: shree
'''

import ...


def nested_loop_by_name_jaccard():
    threshold = 0.75
    q = 3


    con=db.create_connection()
    cur=db.create_cursor(con)
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_jaccard.py
Total Time: 2366.813 milliseconds
Student_id: s4761003
Precision= 0.9069767441860465 , Recall= 0.7358490566037735 , Fmeasure= 0.8124999999999999
Threshold: 0.75 q value: 3

Process finished with exit code 0
```

Writing a python file for automating the call, running and substituting values for q and threshold in nested_loop_by_name_jaccard.py file.

```
import src.data.nested_loop_by_name_jaccard as jaccard
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings("ignore")

df = pd.DataFrame(columns=["q", "threshold", "precision", "recall", "f1"])

q_range = range(1,6)
threshold_range = np.arange(0.2, 1.2, 0.2)

for q in q_range:
    for threshold in threshold_range:
        precision, recall, f_measure = jaccard.nested_loop_by_name_jaccard(q, threshold)
        df = df.append({"q":int(q), "threshold":threshold, "precision":precision, "recall":recall, "f1":f_measure}, ignore_index=True)

df.sort_values(by=["threshold", "q"], inplace=True)
df.reset_index(drop=True, inplace=True)
df["q"] = df["q"].astype('int64')

print("Student_id: s4761003")
print(df)
```

Output



```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataL
Student_id: s4761003
     q  threshold  precision    recall        f1
0    1        0.2   0.000361  0.990566  0.000722
1    2        0.2   0.017692  0.971698  0.034750
2    3        0.2   0.065833  0.943396  0.123077
3    4        0.2   0.116608  0.933962  0.207330
4    5        0.2   0.168111  0.915094  0.284041
5    1        0.4   0.000945  0.971698  0.001889
6    2        0.4   0.236453  0.905660  0.375000
7    3        0.4   0.424107  0.896226  0.575758
8    4        0.4   0.532164  0.858491  0.657040
9    5        0.4   0.649254  0.820755  0.725000
10   1        0.6   0.010255  0.886792  0.020276
11   2        0.6   0.796296  0.811321  0.803738
12   3        0.6   0.836735  0.773585  0.803922
13   4        0.6   0.858696  0.745283  0.797980
14   5        0.6   0.857143  0.735849  0.791878
15   1        0.8   0.206731  0.811321  0.329502
16   2        0.8   0.887640  0.745283  0.810256
17   3        0.8   0.905882  0.726415  0.806283
18   4        0.8   0.904762  0.716981  0.800000
19   5        0.8   0.904762  0.716981  0.800000
20   1        1.0   0.842105  0.754717  0.796020
21   2        1.0   0.914634  0.707547  0.797872
22   3        1.0   0.914634  0.707547  0.797872
23   4        1.0   0.914634  0.707547  0.797872
24   5        1.0   0.914634  0.707547  0.797872

Process finished with exit code 0
```

From the output Dataframe, we can see that after keeping the value of threshold the same while circling through different values of q. There is an increase in the value of the precision while a steady decrease in the value of recall is observed. This pattern is seen as the size of individual substrings in the tokenized list increases, the length of the tokenized list increase. Based on this, the value of the similarity ratio keeps on increasing. More the similarity score, more the restaurant id pairs are being sent for result calculation of precision and recall. Hence, the groups being found at same threshold will see big jumps in precision for big value of q as compared to small value of q at the same defined threshold.

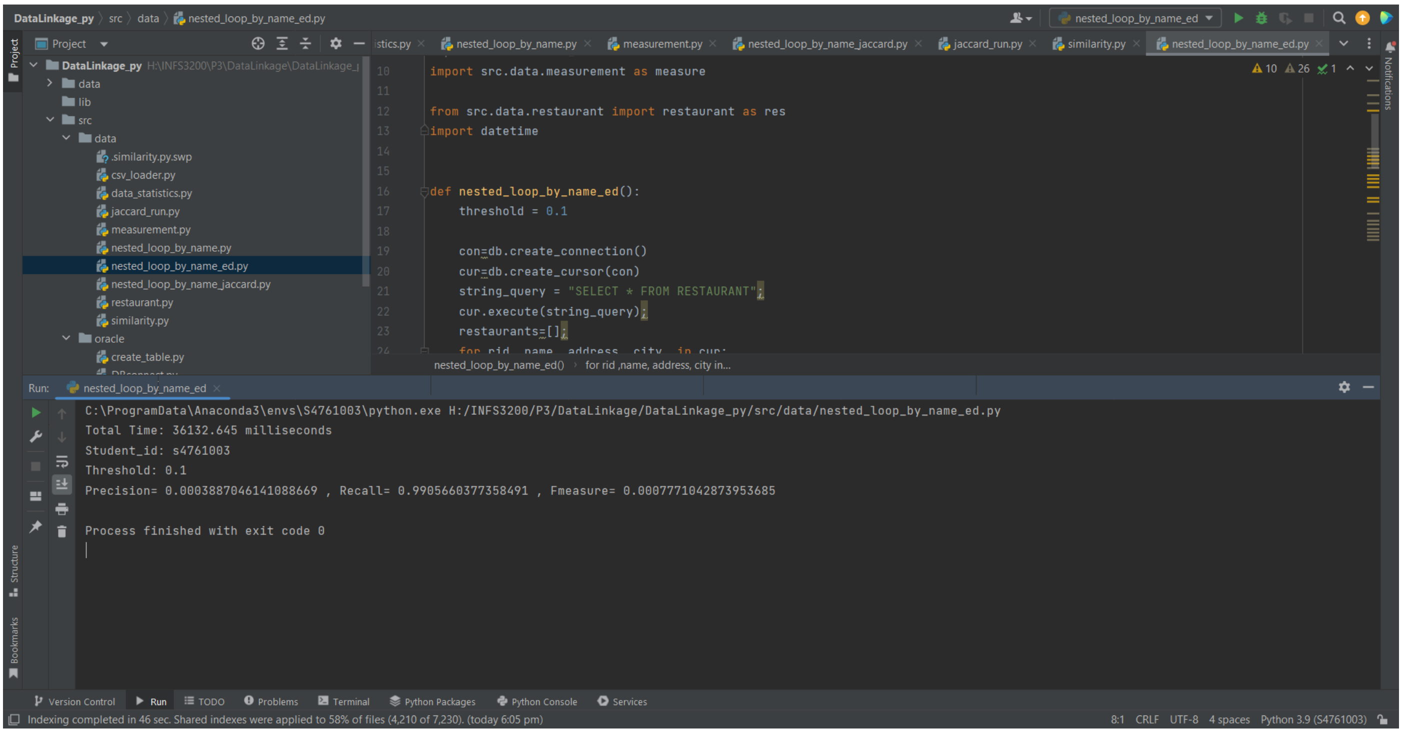
After keeping the value of q the same and going through different values of threshold, there is a very small increase in the value of precision and a very small decrease in the value of recall. This is because the number of restaurant id pairs being found after tokenization will differ minutely but the threshold percentage will keep on increasing. Hence a very small increase in precision is seen as ratio of the number of restaurants names matching (jaccard coefficient) will not be greater for the increasing threshold percentage.

TASK 4

Implementing Edit distance in similarity.py file

```
def calc_ed(str1, str2):
    ed = 0

    if (len(str1) == 0):
        ed = len(str2)
    elif (len(str2) == 0):
        ed = len(str1)
    else:
        len_source = len(str1) + 1
        len_target = len(str2) + 1

        arr = []
        for i in range(len_target):
            arr.append([0] * len_source)

        for i in range(len_target):
            for j in range(len_source):
                if (i == 0):
                    arr[i][j] = j
                elif (j == 0):
                    arr[i][j] = i
                elif (str2[i - 1] == str1[j - 1]):
                    arr[i][j] = arr[i - 1][j - 1]
                else:
                    arr[i][j] = 1 + min(arr[i][j - 1], arr[i - 1][j], arr[i - 1][j - 1])
        ed = arr[len_target - 1][len_source - 1]

    return ed
```

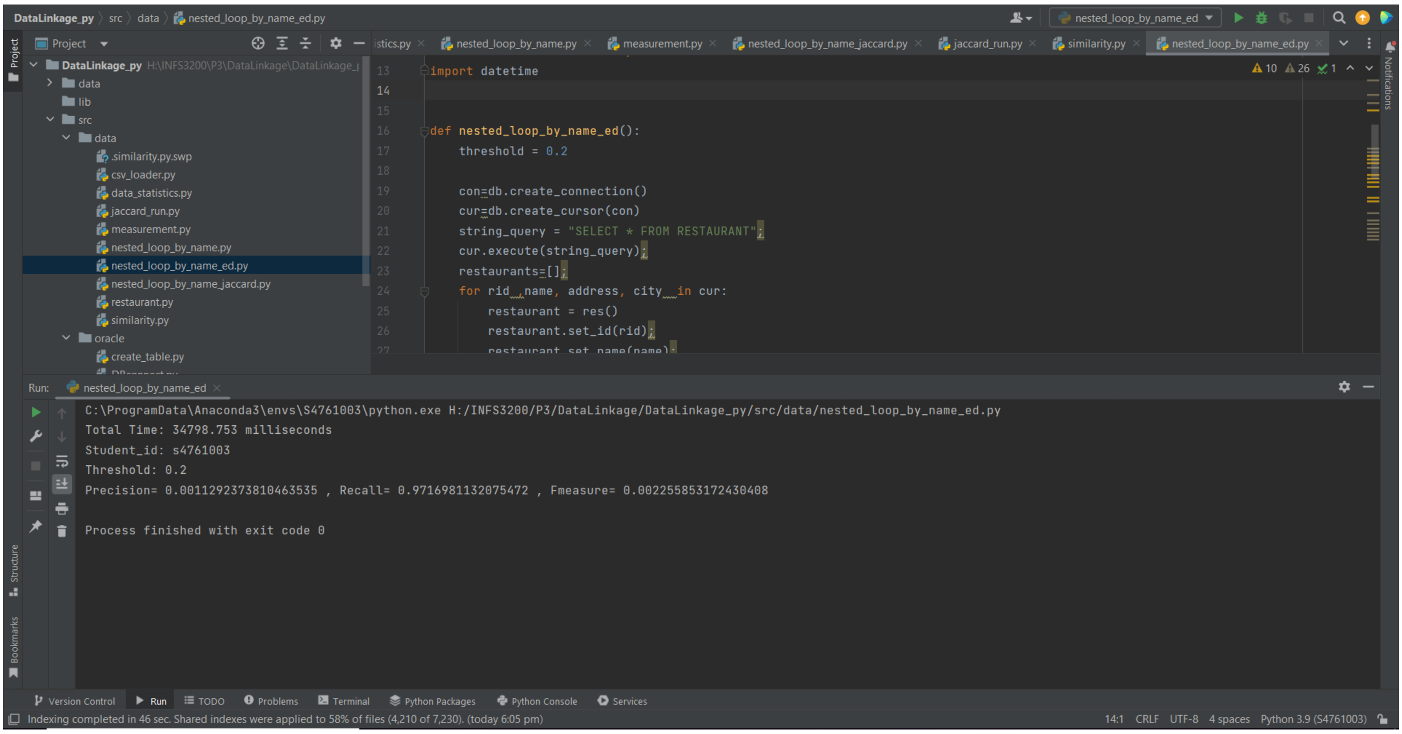
Threshold = 0.1


DataLinkage_py › src › data › nested_loop_by_name_ed.py
Project
DataLinkage_py H:\INFS3200\P3\DataLinkage\DataLinkage_
data
lib
src
data
.similarity.py.swp
csv_loader.py
data_statistics.py
jaccard_run.py
measurement.py
nested_loop_by_name.py
nested_loop_by_name_ed.py
nested_loop_by_name_jaccard.py
restaurant.py
similarity.py
oracle
create_table.py
istics.py
nested_loop_by_name.py
measurement.py
nested_loop_by_name_jaccard.py
jaccard_run.py
similarity.py
nested_loop_by_name_ed.py
nested_loop_by_name_ed
nested_loop_by_name_ed() › for rid ,name, address, city in...
Run:
Version Control
Run
TODO
Problems
Terminal
Python Packages
Python Console
Services
Indexing completed in 46 sec. Shared indexes were applied to 58% of files (4,210 of 7,230). (today 6:05 pm)
8:1 CRLF UTF-8 4 spaces Python 3.9 (S4761003)

```
import src.data.measurement as measure

from src.data.restaurant import restaurant as res
import datetime


def nested_loop_by_name_ed():
    threshold = 0.1

    con=db.create_connection()
    cur=db.create_cursor(con)
    string_query = "SELECT * FROM RESTAURANT";
    cur.execute(string_query);
    restaurants=[];
    for rid ,name, address, city  in cur:
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_ed.py
Total Time: 36132.645 milliseconds
Student_id: s4761003
Threshold: 0.1
Precision= 0.0003887046141088669 , Recall= 0.9905660377358491 , Fmeasure= 0.0007771042873953685

Process finished with exit code 0
```

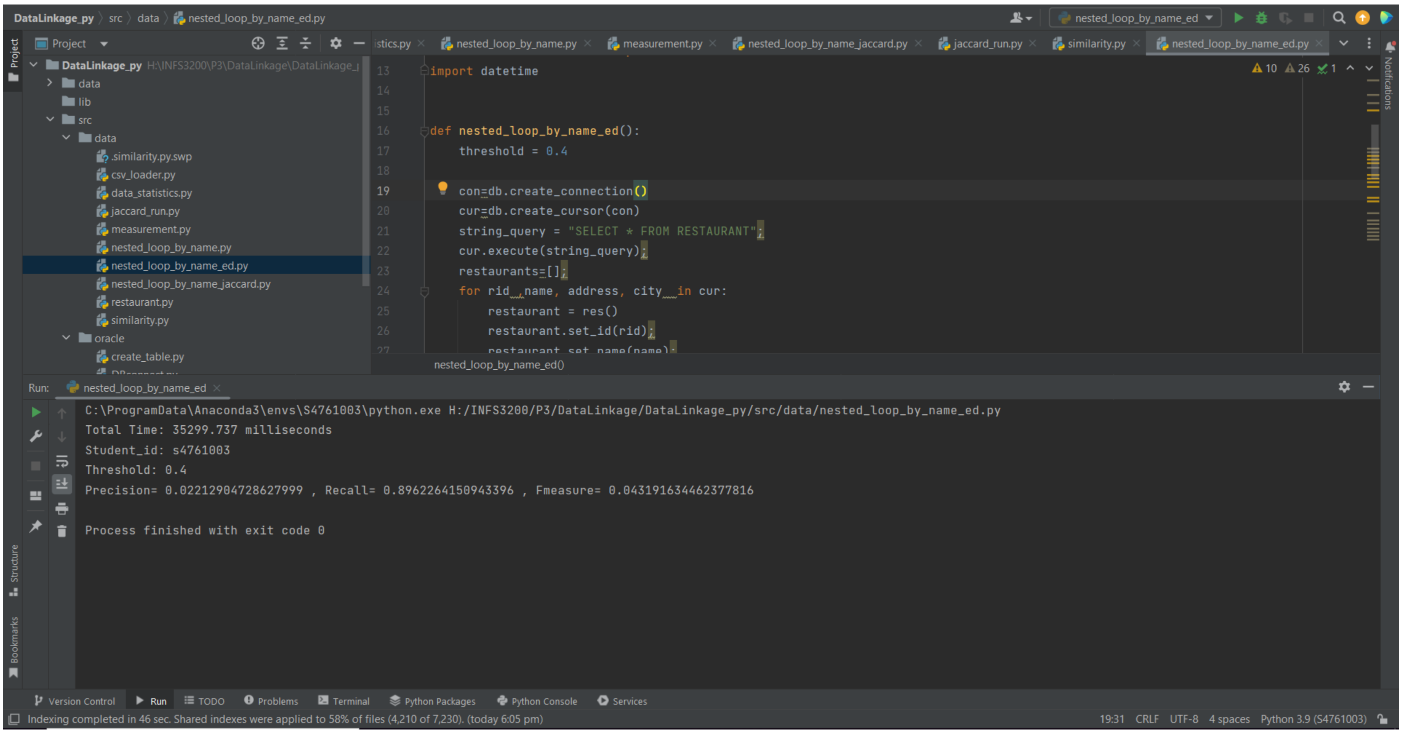
Threshold = 0.2


DataLinkage_py › src › data › nested_loop_by_name_ed.py
Project
DataLinkage_py H:\INFS3200\P3\DataLinkage\DataLinkage_
data
lib
src
data
.similarity.py.swp
csv_loader.py
data_statistics.py
jaccard_run.py
measurement.py
nested_loop_by_name.py
nested_loop_by_name_ed.py
nested_loop_by_name_jaccard.py
restaurant.py
similarity.py
oracle
create_table.py
istics.py
nested_loop_by_name.py
measurement.py
nested_loop_by_name_jaccard.py
jaccard_run.py
similarity.py
nested_loop_by_name_ed.py
nested_loop_by_name_ed
Run:
Version Control
Run
TODO
Problems
Terminal
Python Packages
Python Console
Services
Indexing completed in 46 sec. Shared indexes were applied to 58% of files (4,210 of 7,230). (today 6:05 pm)
14:1 CRLF UTF-8 4 spaces Python 3.9 (S4761003)

```
import datetime


def nested_loop_by_name_ed():
    threshold = 0.2

    con=db.create_connection()
    cur=db.create_cursor(con)
    string_query = "SELECT * FROM RESTAURANT";
    cur.execute(string_query);
    restaurants=[];
    for rid ,name, address, city  in cur:
        restaurant = res()
        restaurant.set_id(rid);
        restaurant.set_name(name);
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_ed.py
Total Time: 34798.753 milliseconds
Student_id: s4761003
Threshold: 0.2
Precision= 0.0011292373810463535 , Recall= 0.9716981132075472 , Fmeasure= 0.002255853172430408

Process finished with exit code 0
```

Threshold = 0.4

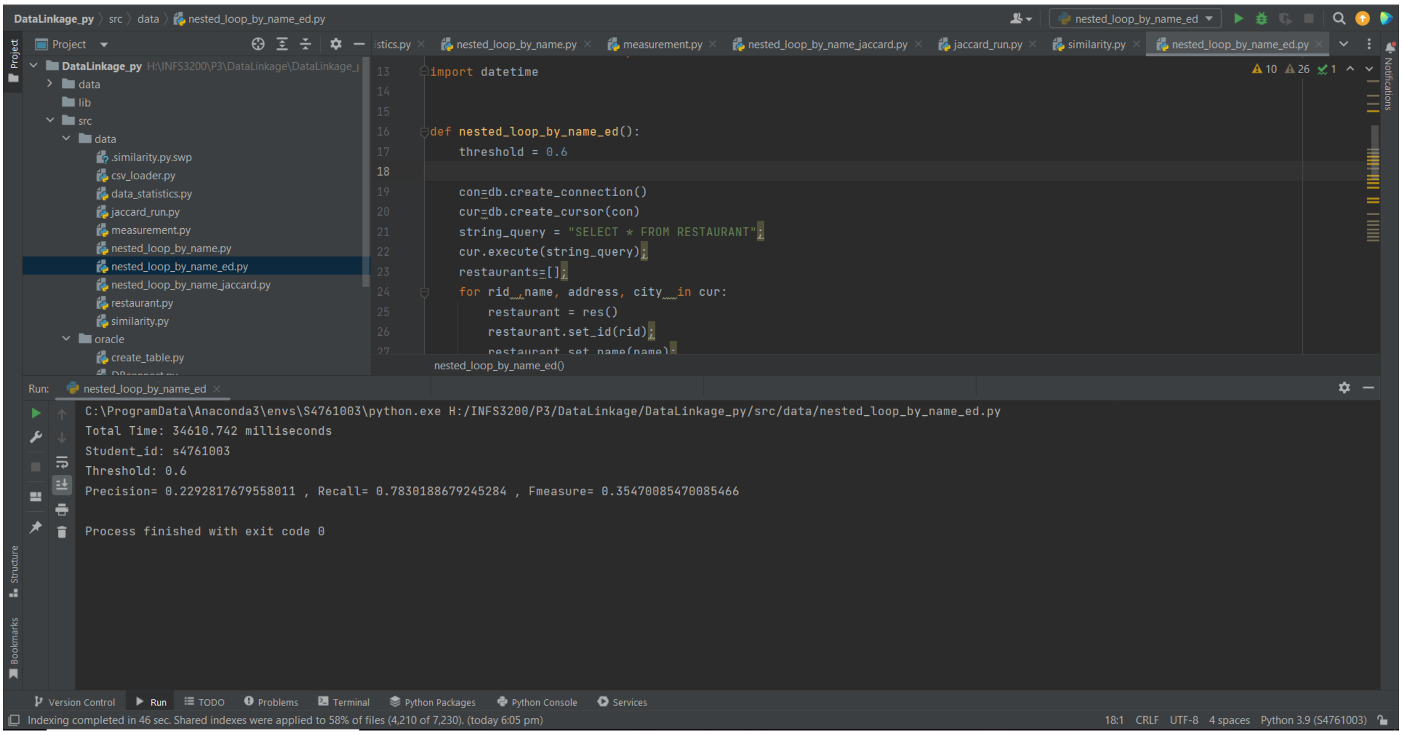


```python
import datetime


def nested_loop_by_name_ed():
    threshold = 0.4

    con=db.create_connection()
    cur=db.create_cursor(con)
    string_query = "SELECT * FROM RESTAURANT";
    cur.execute(string_query);
    restaurants=[];
    for rid ,name, address, city  in cur:
        restaurant = res()
        restaurant.set_id(rid);
        restaurant.set_name(name);
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_ed.py
Total Time: 35299.737 milliseconds
Student_id: s4761003
Threshold: 0.4
Precision= 0.02212904728627999 , Recall= 0.8962264150943396 , Fmeasure= 0.043191634462377816

Process finished with exit code 0
```

Indexing completed in 46 sec. Shared indexes were applied to 58% of files (4,210 of 7,230). (today 6:05 pm)

Threshold = 0.6

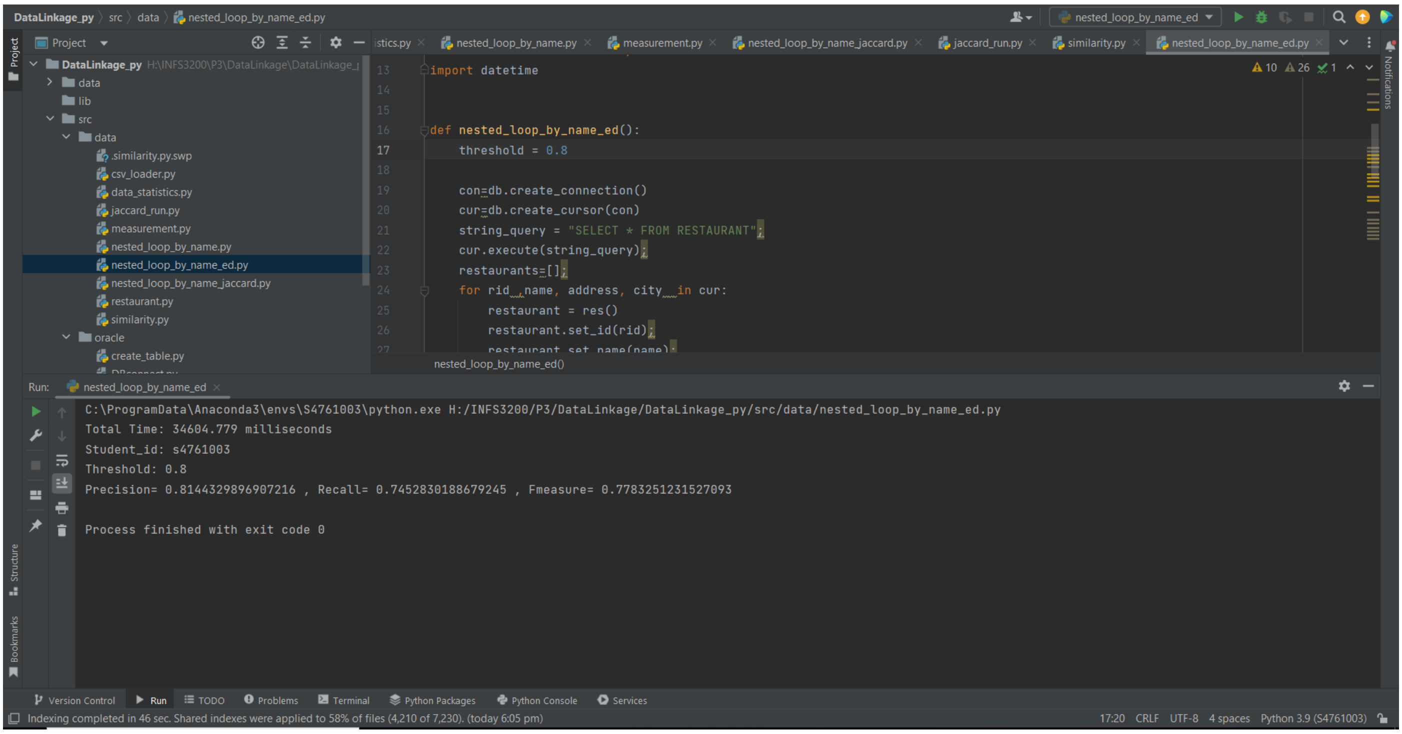


```python
import datetime


def nested_loop_by_name_ed():
    threshold = 0.6

    con=db.create_connection()
    cur=db.create_cursor(con)
    string_query = "SELECT * FROM RESTAURANT";
    cur.execute(string_query);
    restaurants=[];
    for rid ,name, address, city  in cur:
        restaurant = res()
        restaurant.set_id(rid);
        restaurant.set_name(name);
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_ed.py
Total Time: 34610.742 milliseconds
Student_id: s4761003
Threshold: 0.6
Precision= 0.2292817679558011 , Recall= 0.7830188679245284 , Fmeasure= 0.35470085470085466

Process finished with exit code 0
```

Indexing completed in 46 sec. Shared indexes were applied to 58% of files (4,210 of 7,230). (today 6:05 pm)

Threshold = 0.8

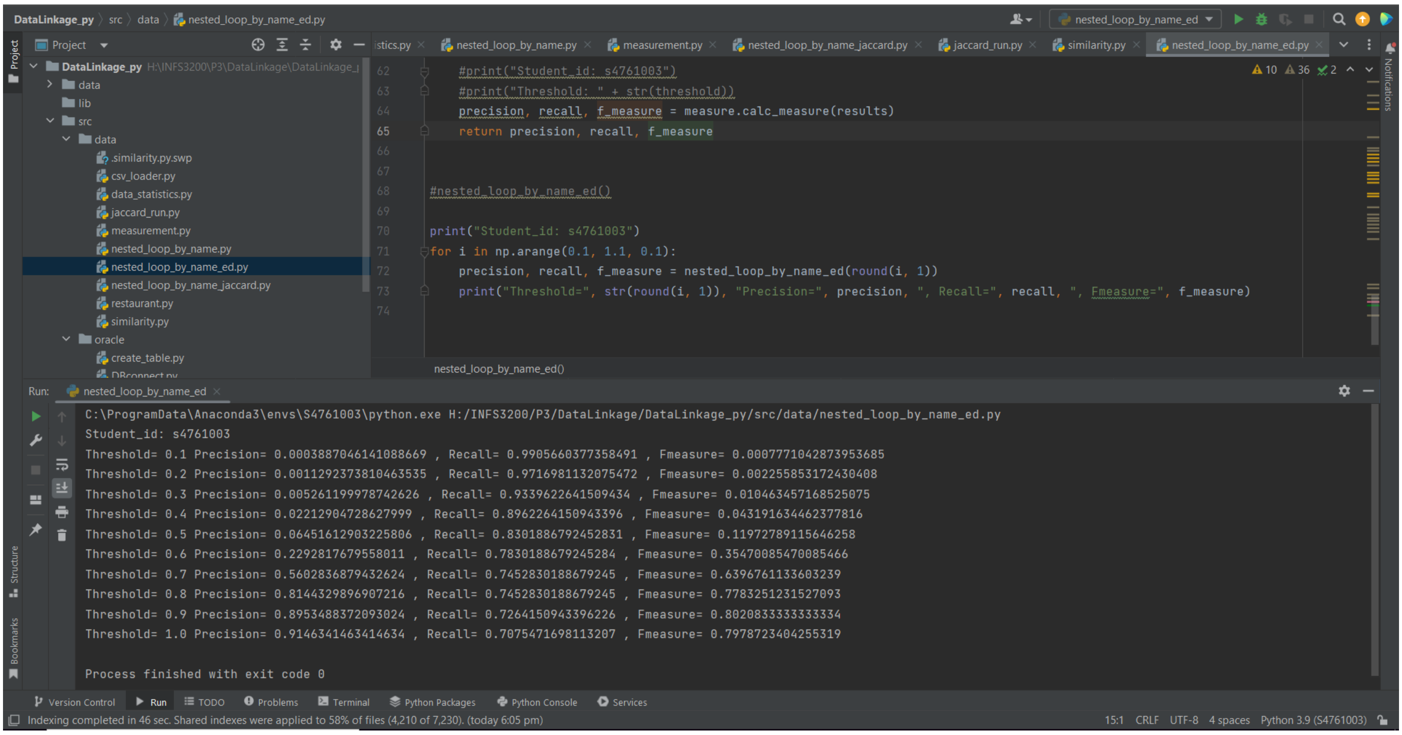


```
import datetime


def nested_loop_by_name_ed():
    threshold = 0.8

    con=db.create_connection()
    cur=db.create_cursor(con)
    string_query = "SELECT * FROM RESTAURANT";
    cur.execute(string_query);
    restaurants=[];
    for rid ,name, address, city  in cur:
        restaurant = res()
        restaurant.set_id(rid);
        restaurant.set_name(name);
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_ed.py
Total Time: 34604.779 milliseconds
Student_id: s4761003
Threshold: 0.8
Precision= 0.8144329896907216 , Recall= 0.7452830188679245 , Fmeasure= 0.7783251231527093

Process finished with exit code 0
```

DataLinkage_py 〉 src 〉 data 〉 nested_loop_by_name_ed.py
Project
nested_loop_by_name_ed.py
Run: nested_loop_by_name_ed
Version Control
Run
TODO
Problems
Terminal
Python Packages
Python Console
Services
Indexing completed in 46 sec. Shared indexes were applied to 58% of files (4,210 of 7,230). (today 6:05 pm)
17:20 CRLF UTF-8 4 spaces Python 3.9 (S4761003)

Writing for loop to get values in a single format -



```
    #print("Student_id: s4761003")
    #print("Threshold: " + str(threshold))
    precision, recall, f_measure = measure.calc_measure(results)
    return precision, recall, f_measure


#nested_loop_by_name_ed()

print("Student_id: s4761003")
for i in np.arange(0.1, 1.1, 0.1):
    precision, recall, f_measure = nested_loop_by_name_ed(round(i, 1))
    print("Threshold=", str(round(i, 1)), "Precision=", precision, ", Recall=", recall, ", Fmeasure=", f_measure)
```

```
C:\ProgramData\Anaconda3\envs\S4761003\python.exe H:/INFS3200/P3/DataLinkage/DataLinkage_py/src/data/nested_loop_by_name_ed.py
Student_id: s4761003
Threshold= 0.1 Precision= 0.0003887046141088669 , Recall= 0.9905660377358491 , Fmeasure= 0.0007771042873953685
Threshold= 0.2 Precision= 0.0011292373810463535 , Recall= 0.9716981132075472 , Fmeasure= 0.002255853172430408
Threshold= 0.3 Precision= 0.005261199978742626 , Recall= 0.9339622641509434 , Fmeasure= 0.010463457168525075
Threshold= 0.4 Precision= 0.02212904728627999 , Recall= 0.8962264150943396 , Fmeasure= 0.043191634462377816
Threshold= 0.5 Precision= 0.06451612903225806 , Recall= 0.8301886792452831 , Fmeasure= 0.11972789115646258
Threshold= 0.6 Precision= 0.2292817679558011 , Recall= 0.7830188679245284 , Fmeasure= 0.35470085470085466
Threshold= 0.7 Precision= 0.5602836879432624 , Recall= 0.7452830188679245 , Fmeasure= 0.6396761133603239
Threshold= 0.8 Precision= 0.8144329896907216 , Recall= 0.7452830188679245 , Fmeasure= 0.7783251231527093
Threshold= 0.9 Precision= 0.8953488372093024 , Recall= 0.7264150943396226 , Fmeasure= 0.8020833333333334
Threshold= 1.0 Precision= 0.9146341463414634 , Recall= 0.7075471698113207 , Fmeasure= 0.7978723404255319

Process finished with exit code 0
```

DataLinkage_py 〉 src 〉 data 〉 nested_loop_by_name_ed.py
Project
nested_loop_by_name_ed.py
Run: nested_loop_by_name_ed
Version Control
Run
TODO
Problems
Terminal
Python Packages
Python Console
Services
Indexing completed in 46 sec. Shared indexes were applied to 58% of files (4,210 of 7,230). (today 6:05 pm)
15:1 CRLF UTF-8 4 spaces Python 3.9 (S4761003)

As the value of threshold increases, the precision increases and recall decreases. Here we can see an inverse relationship between Precision and Recall. Here the ratio of the similarity between the 2 restaurant id pairs is returned. If the similarity ratio is greater than the threshold value the restaurant pair is added to results and sent for calculation of the precision and recall. More the value of similarity ratio in comparison to threshold, more restaurant id pairs are being sent for testing in measure.py file will be in golden standard file. And greater will be the value of the precision calculated.