

DPS-Project Report 1

Reflection:

As part of project1 we are asked to find the business based on city and location. Here I have attached the screenshot of the code for the same and have explained the approach to accomplish in below steps:

```
[ ] import math

def FindBusinessBasedOnCity(cityToSearch, saveLocation1, collection):
    business_list = []
    for index in range(len(collection)):
        business = collection.fetch(index)

        if business['city'] == cityToSearch:
            business_list.append((business['name'], business['full_address'], business['city'], business['state']))

    f = open(saveLocation1, 'w')
    for value in business_list:
        f.write('$'.join(str(s) for s in value))
        f.write('\n')
    f.close()

def FindBusinessBasedOnLocation(categoriesToSearch, myLocation, maxDistance, saveLocation2, collection):
    business_names = []
    latitude1 = myLocation[0]
    longitude1 = myLocation[1]

    for index in range(len(collection.all())):
        business = collection.fetch(index)
        latitude2 = business['latitude']
        longitude2 = business['longitude']

        distance = DistanceFunction(latitude2, longitude2, latitude1, longitude1)
        if distance <= maxDistance:
            for category in categoriesToSearch:
                if category in business['categories']:
                    business_names.append(business['name'])
                    break

    f = open(saveLocation2, 'w')
    for name in business_names:
        f.write(name + '\n')
    f.close()

def DistanceFunction(latitude2, longitude2, latitude1, longitude1):
    R = 3959
    phi1 = math.radians(latitude1)
    phi2 = math.radians(latitude2)
    delta_phi = math.radians(latitude2 - latitude1)
    delta_lambda = math.radians(longitude2 - longitude1)
    a = math.sin(delta_phi/2) * math.sin(delta_phi/2) + math.cos(phi1) * math.cos(phi2) * math.sin(delta_lambda/2) * math.sin(delta_lambda/2)
    c = 2 * math.atan2(math.sqrt(a), math.sqrt(1 - a))
    d = R * c
    return d
```

1. In the Sample template file that's given as part of the FindBusinessBasedOnCity function this is the approach I have done
 - a. I have gone through over the collection data and looked for the matching city parameter in business data to the city to search that's given as input and appended the list with name, full address, city, state to the result data.
 - b. I have opened a file with Name saveLocation1 and by traversing over the result data to append the fields as given in the problem statement
2. In the Sample template file that's given as part of the FindBusinessBasedOnLocation function this is the approach I have done
 - a. Initially have written a separate function named Distance Function which takes the input of two locations in the form of latitude and longitude and calculates the distance between them based on the logic given in problem statement .

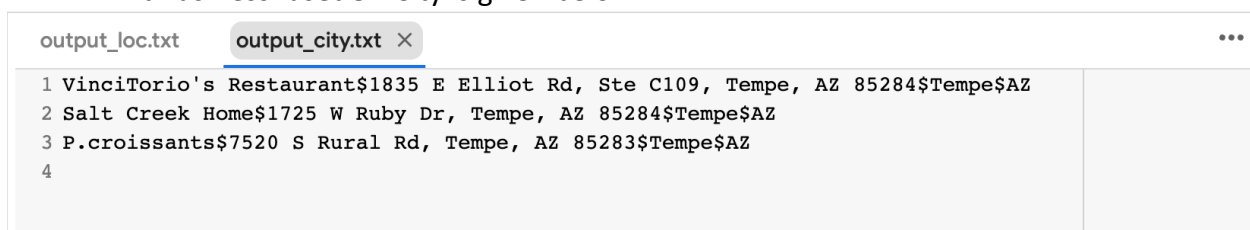
- b. By going through the collection data, for every business tuple, we will calculate the distance between the business location and the inputlocation and if the distance is less then or equal to maxDistance given,
- c. Then we will traverse over the list of categories to search and if any category matches to the category in the business tuple we will add the businessName to the resultant list.
- d. At the end, opened a file with Name saveLocation2 and by traversing over the resultant list have appended all the busineesName to the file.

Lessons learned:

1. Learnt and understood about the Unqlite which is a light weighted NoSQL Database and also where data is store is JSON document store and its installation for the project.
2. Studied and understood the schema and architecture of the data stored in the NoSQL database in the form of key-Value pairs in the form of JSON.
3. Researched and validated the retrieval of data using python notebooks for searching and retrieval of data from the location search and the key concepts in searching for the relevant data from the dataset.
4. I have also validated the different edge cases that should be taken care while getting and comparing the results as part of which I have appended the results in the output tab.

Output:

1. The output which is saved in the saveLocation1 for the data done on the function FindBusinessBasedOn City is given below:

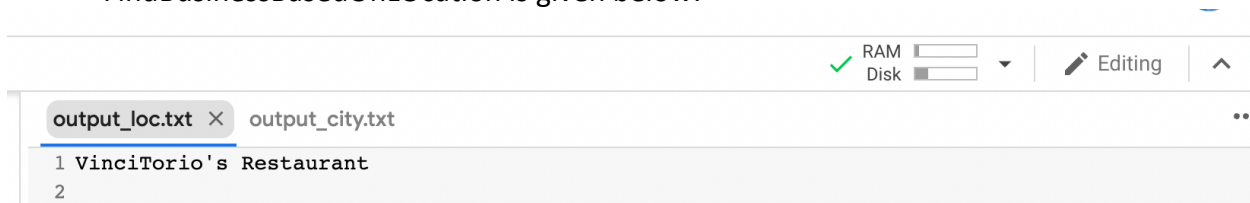


```

output_loc.txt  output_city.txt X
1 VinciTorio's Restaurant$1835 E Elliot Rd, Ste C109, Tempe, AZ 85284$Tempe$AZ
2 Salt Creek Home$1725 W Ruby Dr, Tempe, AZ 85284$Tempe$AZ
3 P.croissants$7520 S Rural Rd, Tempe, AZ 85283$Tempe$AZ
4

```

2. The output which is saved in the saveLocation2 for the data done on the function FindBusinessBasedOnLOcation is given below:



```

output_loc.txt X  output_city.txt
1 VinciTorio's Restaurant
2

```

Result:

Below I have attached the screenshot of the working code for the given test cases in the project where it passed the cases “correct, the given function passes the test cases.

```
13 true_results = ["VinciTorio's Restaurants1835 E Elliot Rd, Ste C109, Tempe, AZ 85284$Tempe$AZ", "P.croissants$7520 S Rural Rd, Tempe, AZ 85283$Tempe$AZ", "Salt Creek Home$1725 W Ruby Dr, Tempe, AZ 85284$Tempe$AZ"]
14
15 try:
16     FindBusinessBasedOnCity('Tempe', 'output_city.txt', data)
17 except NameError as e:
18     print ('The FindBusinessBasedOnCity function is not defined! You must run the cell containing the function before running this evaluation cell.')
19 except TypeError as e:
20     print ("The FindBusinessBasedOnCity function is supposed to accept three arguments. Yours does not!")
21
22 try:
23     opf = open('output_city.txt', 'r')
24 except FileNotFoundError as e:
25     print ("The FindBusinessBasedOnCity function does not write data to the correct location.")
26
27 lines = opf.readlines()
28 if len(lines) != 3:
29     print ("The FindBusinessBasedOnCity function does not find the correct number of results, should be 3.")
30
31 lines = [line.strip() for line in lines]
32 if sorted(lines) == sorted(true_results):
33     print ("Correct! You FindBusinessByCity function passes these test cases. This does not cover all possible test edge cases, however, so make sure that your function covers them before submitting!")
34
35 Correct! You FindBusinessByCity function passes these test cases. This does not cover all possible test edge cases, however, so make sure that your function covers them before submitting!
36
37 true_results = ["VinciTorio's Restaurant"]
38
39 try:
40     FindBusinessBasedOnLocation(['Buffets'], [33.3482589, -111.9088346], 10, 'output_loc.txt', data)
41 except NameError as e:
42     print ('The FindBusinessBasedOnLocation function is not defined! You must run the cell containing the function before running this evaluation cell.')
43 except TypeError as e:
44     print ("The FindBusinessBasedOnLocation function is supposed to accept five arguments. Yours does not!")
45
46 try:
47     opf = open('output_loc.txt', 'r')
48 except FileNotFoundError as e:
49     print ("The FindBusinessBasedOnLocation function does not write data to the correct location.")
50
51 lines = opf.readlines()
52 if len(lines) != 1:
53     print ("The FindBusinessBasedOnLocation function does not find the correct number of results, should be only 1.")
54
55 if lines[0].strip() == true_results[0]:
56     print ("Correct! Your FindBusinessBasedOnLocation function passes these test cases. This does not cover all possible edge cases, so make sure your function does before submitting.")
57
58 Correct! Your FindBusinessBasedOnLocation function passes these test cases. This does not cover all possible edge cases, so make sure your function does before submitting.
```

As part of the test cases I have also validated the additional test cases which has been added to the assignment with commented code and have made sure the code is working fine.