

Project 1 - NoSQL

Vivek Sahukar ASURite ID: 1230067360

Reflection

For this project, I broke the problem in 2 parts: retrieving the data based on geographical location and calculating the distances between locations. I wrote two functions:

`FindBusinessBasedOnCity` and `FindBusinessBasedOnLocation`, which connect with NoSQL database to retrieve business information based on city search and within a certain radius of a given location respectively. For distance calculations, I used the algorithm given in the assignment to write the `DistanceFunction`, which calculates the distance between two points on the earth given their latitude and longitude, that helped in filtering the businesses within the given radius.

```
# Graded Cell, PartID: 01flK

def FindBusinessBasedOnCity(cityToSearch, saveLocation1, collection):
    records = collection.all()
    filtered_businesses = [record for record in records if record['city']
== cityToSearch]
    with open(saveLocation1, 'w') as file:
        for business in filtered_businesses:
            name = business.get('name', '')
            full_address = business.get('full_address', '')
            city = business.get('city', '')
            state = business.get('state', '')
            line = f"{name}${full_address}${city}${state}\n"
            file.write(line)
        print(f"Data saved to {saveLocation1}")

def DistanceFunction(lat2, lon2, lat1, lon1):
    R = 3959
    phi1 = math.radians(lat1)
    phi2 = math.radians(lat2)
    delta_phi = math.radians(lat2 - lat1)
    delta_lambda = math.radians(lon2 - lon1)
```

```

    a = math.sin(delta_phi / 2) ** 2 + math.cos(phi1) * math.cos(phi2) *
math.sin(delta_lambda / 2) ** 2
    c = 2 * math.atan2(math.sqrt(a), math.sqrt(1 - a))
    d = R * c
    return d

def FindBusinessBasedOnLocation(categoriesToSearch, myLocation,
maxDistance, saveLocation2, collection):
    records = collection.all()
    filtered_businesses = []
    for record in records:
        categories = record.get('categories', [])
        if any(category in categoriesToSearch for category in categories):
            business_location = (record.get('latitude', 0),
record.get('longitude', 0))
            distance = DistanceFunction(myLocation[0], myLocation[1],
business_location[0], business_location[1])
            if distance <= maxDistance:
                filtered_businesses.append(record.get('name', ''))

    with open(saveLocation2, 'w') as file:
        for business_name in filtered_businesses:
            file.write(business_name + '\n')

```

Lessons Learned

In this project, I learned how to use NoSQL databases especially UnQLite, and how to interact with them using Python. I got experience with geospatial data processing, particularly how to search and filter data based on the given geographical constraints and save the final results in a required format. Moreover, I learned how to debug the code and write a clean code that's easier for others to understand and modify for future requirements. I also learned about error handling and debugging techniques by using the test cases provided.

Output

```
vs-Assignment+5+NoSQL.ipynb  output_city.txt X
home > vs > Dropbox (ASU) > asu sem 1 courses > cse511-data-processing-at-scale > project-1 > output_city.txt
1 Penny's Restaurant$1330 S Power Rd, Mesa, AZ 85206$Mesa$AZ
2 Bikram Yoga$1940 W 8th St, Ste 111, Mesa, AZ 85202$Mesa$AZ
3 Southeast Valley Medical Group$1950 S Country Club Dr, Mesa, AZ 85210$Mesa$AZ
4 The Seafood Market$1910 S Gilbert Rd, Mesa, AZ 85204$Mesa$AZ
5 Diamondback Gymnastics$7211 E Southern Avenue, Mesa, AZ 85209$Mesa$AZ
6 Arizona Exterminating Co.$521 E Broadway Rd, Mesa, AZ 85204$Mesa$AZ
7 Spa Pima$2150 S Power Rd, Mesa, AZ 85209$Mesa$AZ
8

vs-Assignment+5+NoSQL.ipynb  output_loc.txt X
home > vs > Dropbox (ASU) > asu sem 1 courses > cse511-data-processing-at-scale > project-1 > output_loc.txt
1 The Seafood Market
2 P.croissants
3
```

Result

Data saved to output_city.txt

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!

Correct! Your FindBusinessBasedOnLocation function passes these test cases. This does not cover all possible edge cases, so make sure your function does before submitting.