Introduction

Implement the functions provided in the Jupyter Notebook to perform the operations as listed below. You may use the included sample.db to test your functions with the Python code provided or you may use the Jupyter Notebook environment, which has the sample database embedded within.

**A. FindBusinessBasedOnCity(cityToSearch, saveLocation1, collection) –** This function searches the ‘collection’ given to find all the business present in the city provided in ‘cityToSearch’ and save it to ‘saveLocation1’. For each business you found, you should store the name, full address, city, and state of the business in the following format. Each line of the saved file will contain: Name$FullAddress$City$State. ($ is the separator and must be present.)

**B. FindBusinessBasedOnLocation(categoriesToSearch, myLocation, maxDistance, saveLocation2, collection) –** This function searches the ‘collection’ given to find the name of all the businesses present in the ‘maxDistance’ from the given ‘myLocation’ (please use the distance algorithm given below) and save them to ‘saveLocation2’. Each line of the output file will contain the name of the business only.

**sample**DB File

[Download file](https://d3c33hcgiwev3.cloudfront.net/8qR4HZEMEeiHxhIKzPUaYA_f2e9e510910c11e89a76a93431138c0c_sample.db?Expires=1639353600&Signature=DvfsCD4eoE7Nwm2tBUuL1LaviQ9LwbeduzmasTg2X-b1fziKla4yQguYlvFWBiaGrUG~URHADEkJALwYRe0Eb7rO6cFLBxUccLnEPIbJ1gEtQ8pL1NCK8rB4uEcHf0jTIdS3acUSIRExyOyH5CZIOlCOvQvVNQpRNhsvYP2LIZI_&Key-Pair-Id=APKAJLTNE6QMUY6HBC5A" \t "_blank)

**Distance Algorithm**

A distance algorithm will need to be used. Given two pairs of latitude and longitude as [lat2, lon2] and [lat1, lon1], you can calculate the distance between them using the formula given below:

DistanceFunction(lat2, lon2, lat1, lon1):

* var R = 3959; // miles
* var φ1 = lat1.toRadians();
* var φ2 = lat2.toRadians();
* var Δφ = (lat2-lat1).toRadians();
* var Δλ = (lon2-lon1).toRadians();
* var a = Math.sin(Δφ/2) \* Math.sin(Δφ/2) + Math.cos(φ1) \* Math.cos(φ2) \* Math.sin(Δλ/2) \* Math.sin(Δλ/2);
* var c = 2 \* Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
* var d = R \* c;
* d is the distance between the given pair of latitude and longitude. The distance is in **miles.**

Reference: <http://www.movable-type.co.uk/scripts/latlong.html>

Test Cases

Test cases are provided in the file below.

**Project1-AdditionalTestCases**DOCX File

[Download file](https://d3c33hcgiwev3.cloudfront.net/6ISi1tieTjSEotbYno40gQ_fdf84719ff614c5186bc9cbef0b031f1_Project1-AdditionalTestCases.docx?Expires=1639353600&Signature=latJo2EiB-PcdptjpfkBwg3zVpWMSFB9F43k59dAejNh-fEek5Uq8GzksGqKnUcuDe8eWNu69JLULoMgZ~tu6SVrZCsVEa4u~aR5~Cpei38KNrc-qhHg4K47cAh23ELn-iHFKT5bhZR1reMhjylu7Ks0EpBXF6T~iVqhGa6vKrA_&Key-Pair-Id=APKAJLTNE6QMUY6HBC5A" \t "_blank)

Feedback

There is one test for a total of one point. If some part of your data is incorrect, you will get partial scores of 0.25, 0.50, or 0.75. **If the submission fails, you will see the corresponding error logs that indicate *where* the error occurred.**

**Submission:** Submit your .ipynb file to the Programming Assignment.

**Common Errors**

1. Error: Submission was not a well-formed Jupyter Notebook file.
2. Save the file as IPYNB directly from Jupiter Notebook rather than saving it in other format and then replacing their properties as it makes the file unstable.
3. Runtime errors happen in your submission invalid syntax (<string>, line 22)
4. Error: Distance function, FindBusinessBasedOnCity, FindBusinessBasedOnLocation
5. Your submission did not define the FindBusinessBasedOnLocation function, but passed all tests for FindBusinessBasedOnCity.
6. Runtime errors happen in your submission name 'data' is not defined.
7. The original function needs to remain untouched and students need to write code inside the function definition block.

**Additional Common Errors**

A. **Unqlite** treats everything as bytes, not strings so things that work in python 3 will probably break in python 2. Grader works with python 2. So they need to use Python 2 for this project. B. **Malformed feedback** error. Solutions: - Grader won't accept new functions to be defined, so the code should be within the already mentioned functions. use normal libraries in 2.7. - You need to import any library in the cell with function code. - Delete all the self-defined 'print' functions, - Also please make sure that all your code is in the predefined cell.

C. The following error message: ***Your submission did not define the FindBusinessBasedOnLocation function, but passed all tests for FindBusinessBasedOnCity.*** - Think like an auto-grader. Only the graded cell is examined and the other cells are ignored.