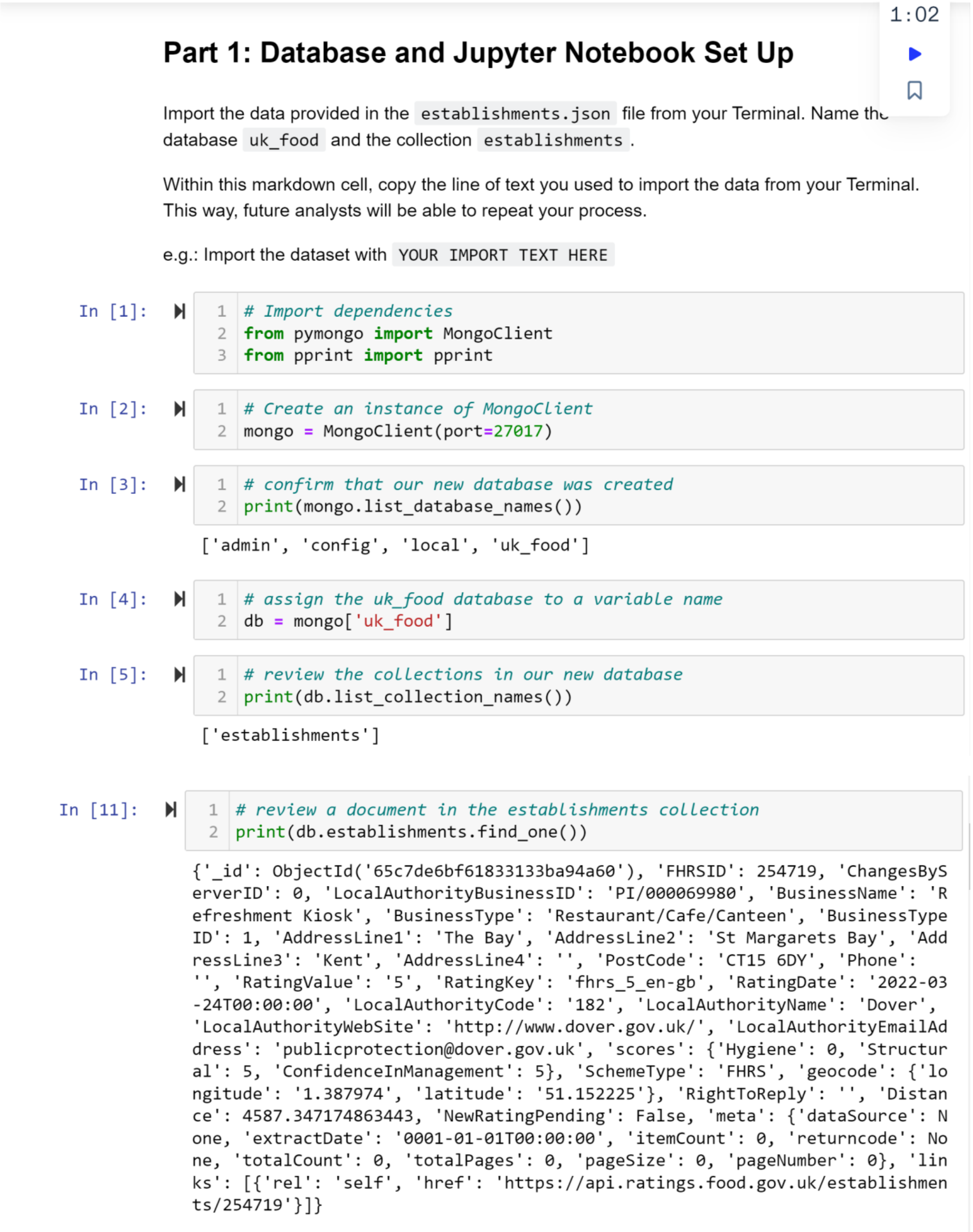


Part 1: Database and Jupyter Notebook Set Up

Import the data provided in the `establishments.json` file from your Terminal. Name the database `uk_food` and the collection `establishments`.

Within this markdown cell, copy the line of text you used to import the data from your Terminal. This way, future analysts will be able to repeat your process.

e.g.: Import the dataset with `YOUR IMPORT TEXT HERE`

In [1]:

```python
# Import dependencies
from pymongo import MongoClient
from pprint import pprint
```

In [2]:

```python
# Create an instance of MongoClient
mongo = MongoClient(port=27017)
```

In [3]:

```python
# confirm that our new database was created
print(mongo.list_database_names())
```

```
['admin', 'config', 'local', 'uk_food']
```

In [4]:

```python
# assign the uk_food database to a variable name
db = mongo['uk_food']
```

In [5]:

```python
# review the collections in our new database
print(db.list_collection_names())
```

```
['establishments']
```

In [11]:

```python
# review a document in the establishments collection
print(db.establishments.find_one())
```

```
{'_id': ObjectId('65c7de6bf61833133ba94a60'), 'FHRSID': 254719, 'ChangesByServerID': 0, 'LocalAuthorityBusinessID': 'PI/000069980', 'BusinessName': 'Refreshment Kiosk', 'BusinessType': 'Restaurant/Cafe/Canteen', 'BusinessTypeID': 1, 'AddressLine1': 'The Bay', 'AddressLine2': 'St Margarets Bay', 'AddressLine3': 'Kent', 'AddressLine4': '', 'PostCode': 'CT15 6DY', 'Phone': '', 'RatingValue': '5', 'RatingKey': 'fhrs_5_en-gb', 'RatingDate': '2022-03-24T00:00:00', 'LocalAuthorityCode': '182', 'LocalAuthorityName': 'Dover', 'LocalAuthorityWebSite': 'http://www.dover.gov.uk/', 'LocalAuthorityEmailAddress': 'publicprotection@dover.gov.uk', 'scores': {'Hygiene': 0, 'Structural': 5, 'ConfidenceInManagement': 5}, 'SchemeType': 'FHRS', 'geocode': {'longitude': '1.387974', 'latitude': '51.152225'}, 'RightToReply': '', 'Distance': 4587.347174863443, 'NewRatingPending': False, 'meta': {'dataSource': None, 'extractDate': '0001-01-01T00:00:00', 'itemCount': 0, 'returncode': None, 'totalCount': 0, 'totalPages': 0, 'pageSize': 0, 'pageNumber': 0}, 'links': [{'rel': 'self', 'href': 'https://api.ratings.food.gov.uk/establishments/254719'}]}
```

In [13]:

```
# assign the collection to a variable
establishments = db['establishments']
```

Part 2: Update the Database

1. An exciting new halal restaurant just opened in Greenwich, but hasn't been rated yet. The magazine has asked you to include it in your analysis. Add the following restaurant "Penang Flavours" to the database.

In [14]:

```
establishments.find_one({})
```

Out[14]:

```
{'_id': ObjectId('65c7de6bf61833133ba94a60'),
 'FHRSID': 254719,
 'ChangesByServerID': 0,
 'LocalAuthorityBusinessID': 'PI/000069980',
 'BusinessName': 'Refreshment Kiosk',
 'BusinessType': 'Restaurant/Cafe/Canteen',
 'BusinessTypeID': 1,
 'AddressLine1': 'The Bay',
 'AddressLine2': 'St Margarets Bay',
 'AddressLine3': 'Kent',
 'AddressLine4': '',
 'PostCode': 'CT15 6DY',
 'Phone': '',
 'RatingValue': '5',
 'RatingKey': 'fhrs_5_en-gb',
 'RatingDate': '2022-03-24T00:00:00',
 'LocalAuthorityCode': '182',
 'LocalAuthorityName': 'Dover',
 'LocalAuthorityWebSite': 'http://www.dover.gov.uk/',
 'LocalAuthorityEmailAddress': 'publicprotection@dover.gov.uk',
 'scores': {'Hygiene': 0, 'Structural': 5, 'ConfidenceInManagement': 5},
 'SchemeType': 'FHRS',
 'geocode': {'longitude': '1.387974', 'latitude': '51.152225'},
```

In [15]:

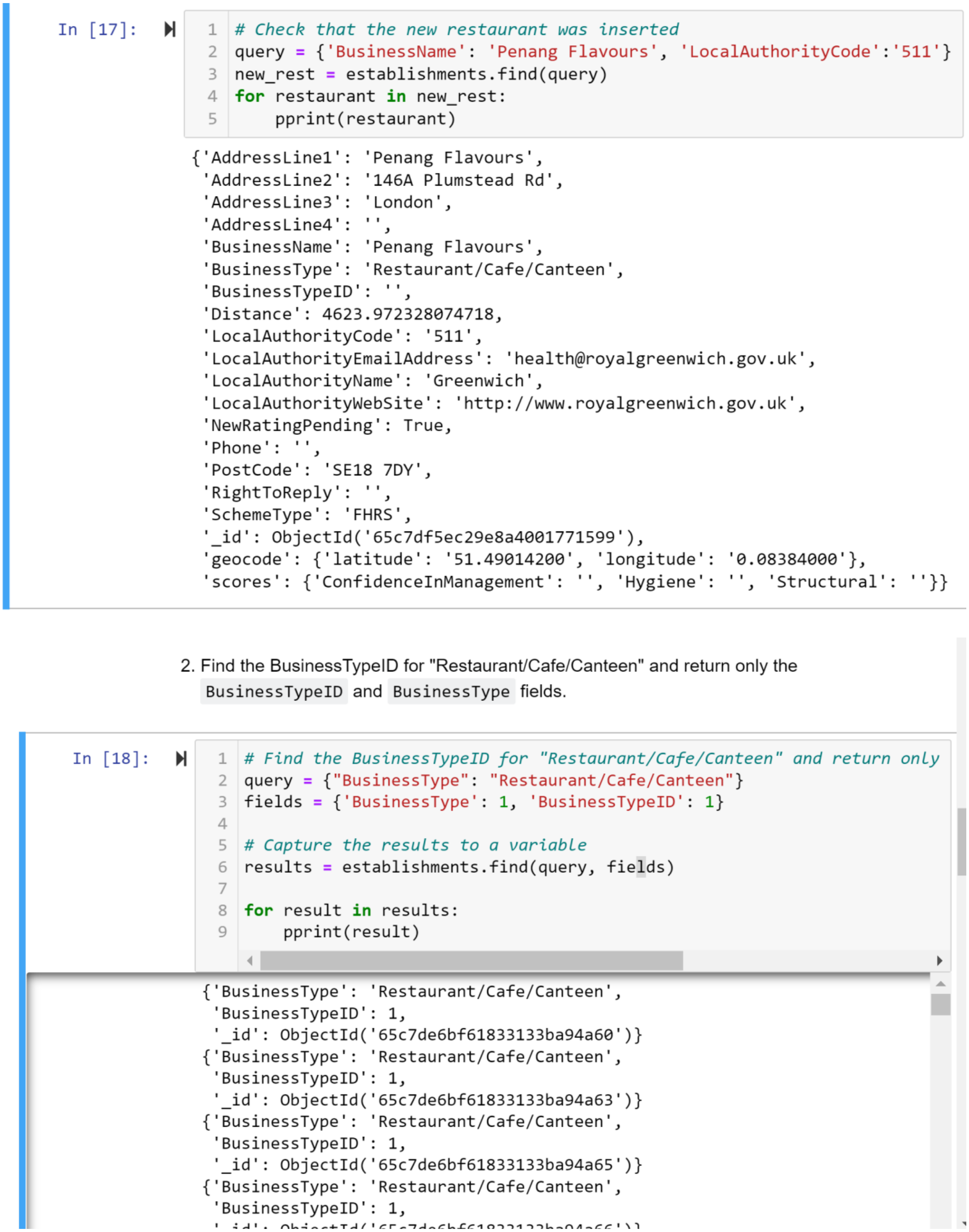
```
# Create a dictionary for the new restaurant data
new_rest = {
    "BusinessName":"Penang Flavours",
    "BusinessType":"Restaurant/Cafe/Canteen",
    "BusinessTypeID":"",
    "AddressLine1":"Penang Flavours",
    "AddressLine2":"146A Plumstead Rd",
    "AddressLine3":"London",
    "AddressLine4":"",
    "PostCode":"SE18 7DY",
    "Phone":"",
    "LocalAuthorityCode":"511",
    "LocalAuthorityName":"Greenwich",
    "LocalAuthorityWebSite":"http://www.royalgreenwich.gov.uk",
    "LocalAuthorityEmailAddress":"health@royalgreenwich.gov.uk",
    "scores":{
        "Hygiene":"",
        "Structural":"",
        "ConfidenceInManagement":""
    },
    "SchemeType":"FHRS",
    "geocode":{
        "longitude":"0.08384000",
        "latitude":"51.49014200"
    },
    "RightToReply":"",
    "Distance":4623.9723280747176,
    "NewRatingPending":True
}

```

In [16]:

```
# Insert the new restaurant into the collection
establishments.insert_one(new_rest)
```

Out[16]:

```
InsertOneResult(ObjectId('65c7df5ec29e8a4001771599'), acknowledged=True)
```



In [17]:

```python
# Check that the new restaurant was inserted
query = {'BusinessName': 'Penang Flavours', 'LocalAuthorityCode':'511'}
new_rest = establishments.find(query)
for restaurant in new_rest:
    pprint(restaurant)
```

```
{'AddressLine1': 'Penang Flavours',
 'AddressLine2': '146A Plumstead Rd',
 'AddressLine3': 'London',
 'AddressLine4': '',
 'BusinessName': 'Penang Flavours',
 'BusinessType': 'Restaurant/Cafe/Canteen',
 'BusinessTypeID': '',
 'Distance': 4623.972328074718,
 'LocalAuthorityCode': '511',
 'LocalAuthorityEmailAddress': 'health@royalgreenwich.gov.uk',
 'LocalAuthorityName': 'Greenwich',
 'LocalAuthorityWebSite': 'http://www.royalgreenwich.gov.uk',
 'NewRatingPending': True,
 'Phone': '',
 'PostCode': 'SE18 7DY',
 'RightToReply': '',
 'SchemeType': 'FHRS',
 '_id': ObjectId('65c7df5ec29e8a4001771599'),
 'geocode': {'latitude': '51.49014200', 'longitude': '0.08384000'},
 'scores': {'ConfidenceInManagement': '', 'Hygiene': '', 'Structural': ''}}
```

2. Find the BusinessTypeID for "Restaurant/Cafe/Canteen" and return only the `BusinessTypeID` and `BusinessType` fields.

In [18]:

```python
# Find the BusinessTypeID for "Restaurant/Cafe/Canteen" and return only
query = {"BusinessType": "Restaurant/Cafe/Canteen"}
fields = {'BusinessType': 1, 'BusinessTypeID': 1}

# Capture the results to a variable
results = establishments.find(query, fields)

for result in results:
    pprint(result)
```

```
{'BusinessType': 'Restaurant/Cafe/Canteen',
 'BusinessTypeID': 1,
 '_id': ObjectId('65c7de6bf61833133ba94a60')}
{'BusinessType': 'Restaurant/Cafe/Canteen',
 'BusinessTypeID': 1,
 '_id': ObjectId('65c7de6bf61833133ba94a63')}
{'BusinessType': 'Restaurant/Cafe/Canteen',
 'BusinessTypeID': 1,
 '_id': ObjectId('65c7de6bf61833133ba94a65')}
{'BusinessType': 'Restaurant/Cafe/Canteen',
 'BusinessTypeID': 1,
 '_id': ObjectId('65c7de6bf61833133ba94a66')}
```

3. Update the new restaurant with the `BusinessTypeID` you found.

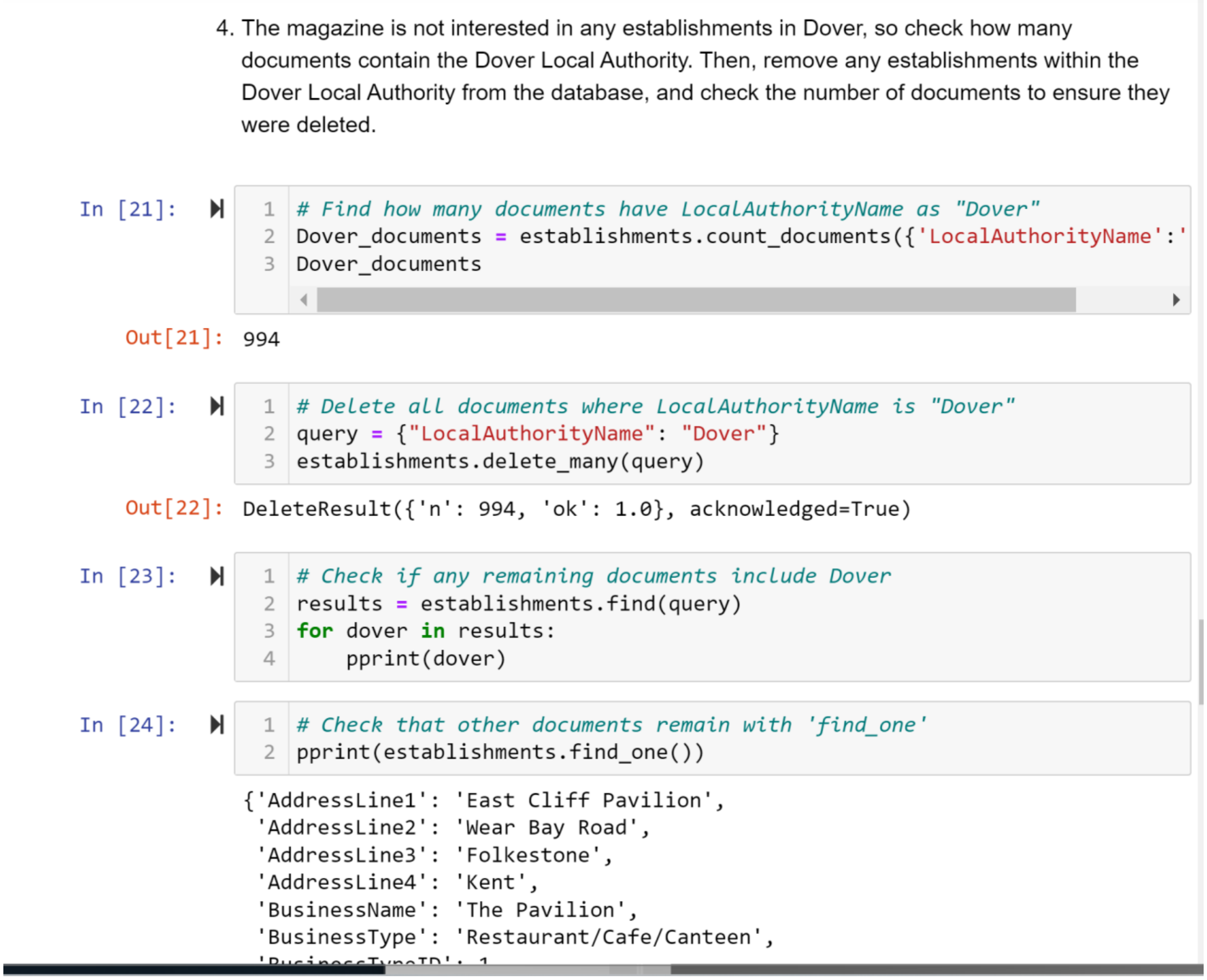
In [19]:

```
# Update the new restaurant with the correct BusinessTypeID
query = {'BusinessTypeID': 1}
results = establishments.update_one(
    {'BusinessName': 'Penang Flavours'},
    {'$set':{"BusinessTypeID":1}
    }
)
```

In [20]:

```
# Confirm that the new restaurant was updated
query = {'BusinessName': 'Penang Flavours', 'LocalAuthorityCode':'511'}
new_rest = establishments.find(query)
for restaurant in new_rest:
    pprint(restaurant)
```

```
{'AddressLine1': 'Penang Flavours',
 'AddressLine2': '146A Plumstead Rd',
 'AddressLine3': 'London',
 'AddressLine4': '',
 'BusinessName': 'Penang Flavours',
 'BusinessType': 'Restaurant/Cafe/Canteen',
 'BusinessTypeID': 1,
 'Distance': 4623.972328074718,
 'LocalAuthorityCode': '511',
 'LocalAuthorityEmailAddress': 'health@royalgreenwich.gov.uk',
 'LocalAuthorityName': 'Greenwich',
 'LocalAuthorityWebSite': 'http://www.royalgreenwich.gov.uk',
 'NewRatingPending': True,
 'Phone': '',
```



4. The magazine is not interested in any establishments in Dover, so check how many documents contain the Dover Local Authority. Then, remove any establishments within the Dover Local Authority from the database, and check the number of documents to ensure they were deleted.

In [21]:

```
# Find how many documents have LocalAuthorityName as "Dover"
Dover_documents = establishments.count_documents({'LocalAuthorityName':'
Dover_documents
```

Out[21]:

```
994
```

In [22]:

```
# Delete all documents where LocalAuthorityName is "Dover"
query = {"LocalAuthorityName": "Dover"}
establishments.delete_many(query)
```

Out[22]:

```
DeleteResult({'n': 994, 'ok': 1.0}, acknowledged=True)
```

In [23]:

```
# Check if any remaining documents include Dover
results = establishments.find(query)
for dover in results:
    pprint(dover)
```

In [24]:

```
# Check that other documents remain with 'find_one'
pprint(establishments.find_one())
```

```
{'AddressLine1': 'East Cliff Pavilion',
 'AddressLine2': 'Wear Bay Road',
 'AddressLine3': 'Folkestone',
 'AddressLine4': 'Kent',
 'BusinessName': 'The Pavilion',
 'BusinessType': 'Restaurant/Cafe/Canteen',
 'BusinessTypeID': 1
```

5. Some of the number values are stored as strings, when they should be stored as numbers.

Use `update_many` to convert `latitude` and `longitude` to decimal numbers.

In [25]:

```
# Change the data type from String to Decimal for longitude and latitude
establishments.update_many({}, [ {'$set':{ "geocode.longitude" : {'$toDe
                                           "geocode.latitude" : {'$toDeci
```

Out[25]:

```
UpdateResult({'n': 38786, 'nModified': 38786, 'ok': 1.0, 'updatedExisting':
True}, acknowledged=True)
```

Use `update_many` to convert `RatingValue` to integer numbers.

In [26]:

```
# Set non 1-5 Rating Values to Null
non_ratings = ["AwaitingInspection", "Awaiting Inspection", "AwaitingPub
establishments.update_many({"RatingValue": {"$in": non_ratings}}, [ {'$s
```

Out[26]:

```
UpdateResult({'n': 4091, 'nModified': 4091, 'ok': 1.0, 'updatedExisting': T
rue}, acknowledged=True)
```

In [27]:

```
# Change the data type from String to Integer for RatingValue
establishments.update_many({}, [ {'$set':{ "RatingValue" : {'$toInt': "$
```

Out[27]:

```
UpdateResult({'n': 38786, 'nModified': 34695, 'ok': 1.0, 'updatedExisting':
True}, acknowledged=True)
```

In [30]:

```
# Check that the coordinates and rating value are now numbers
pprint(establishments.find_one())
```

```
{'AddressLine1': 'East Cliff Pavilion',
 'AddressLine2': 'Wear Bay Road',
 'AddressLine3': 'Folkestone',
 'AddressLine4': 'Kent',
 'BusinessName': 'The Pavilion',
 'BusinessType': 'Restaurant/Cafe/Canteen',
 'BusinessTypeID': 1,
 'ChangesByServerID': 0,
 'Distance': 4591.765489457773,
 'FHRSID': 1043695,
 'LocalAuthorityBusinessID': 'PI/000073616',
 'LocalAuthorityCode': '188',
 'LocalAuthorityEmailAddress': 'foodteam@folkestone-hythe.gov.uk',
 'LocalAuthorityName': 'Folkestone and Hythe',
 'LocalAuthorityWebSite': 'http://www.folkestone-hythe.gov.uk',
 'NewRatingPending': False,
 'Phone': '',
 'PostCode': 'CT19 6BL',
 'RatingDate': '2018-04-04T00:00:00',
 'RatingKey': 'fhrs_5_en-gb',
```

In [ ]: