documentdb

March 26, 2021

[8]:
```
from pathlib import Path
import json
import os
import pandas as pd

from tinydb import TinyDB

current_dir = Path(os.getcwd()).absolute()
results_dir = current_dir.joinpath('results')
kv_data_dir = results_dir.joinpath('kvdb')
kv_data_dir.mkdir(parents=True, exist_ok=True)

class DocumentDB(object):
    def __init__(self, db_path):

        # Get json file paths from previous assignment
        people_json = kv_data_dir.joinpath('people.json')
        visited_json = kv_data_dir.joinpath('visited.json')
        sites_json = kv_data_dir.joinpath('sites.json')
        measurements_json = kv_data_dir.joinpath('measurements.json')

        self._db_path = Path(db_path)
        self._db = None

        # Load json files as dicts
        with open(people_json) as file:
            people_dict = json.load(file)
        with open(visited_json) as file:
            visited_dict = json.load(file)
        with open(sites_json) as file:
            sites_dict = json.load(file)
        with open(measurements_json) as file:
            measurements_dict = json.load(file)

        self._load_db()

        # Iterate through people dictionary to create record and add to database
```

```
        for people_k, people_v in people_dict.items():

            # Create a list of visits
            people_v['visits'] = []
            for visited_k, visited_v in visited_dict.items():
                for sites_k, sites_v in sites_dict.items():

                    # If site_id in site_v dict matches site_id in visited_v
                    # dict, store site_v in current visited_v dict as site
                    if sites_v['site_id'] == visited_v['site_id']:
                        visited_v['site'] = sites_v

                        # Create measurements list for current person and visit
                        visited_v['measurements'] = []

                        for measurements_k,\
                            measurements_v in measurements_dict.items():

                            # If visit_id and person_id match,
                            # add the measurements to the current visited_v
                            # measurements list
                            if (measurements_v['visit_id']\
                                == visited_v['visit_id'])\
                            & (measurements_v['person_id']\
                               == people_v['person_id']):
                                visited_v['measurements'].append(measurements_v)

                        # If there are measurements in the list,
                        # add the visit by appending the people_v visits list
                        if len(visited_v['measurements']) != 0:
                            people_v['visits'].append(visited_v)

            # Add person to the db
            self._db.insert(people_v)


    def _load_db(self):
        # Load db with prettified formatting
        self._db = TinyDB(self._db_path, indent=4, separators=(',', ': '))
```

[9]:
```
db_path = results_dir.joinpath('patient-info.json')
if db_path.exists():
    os.remove(db_path)

db = DocumentDB(db_path)
```

[ ]:
```

```