

2.2

June 16, 2021

```
[3]: from pathlib import Path
import json
import os

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)

[4]: class DocumentDB(object):
    def __init__(self, db_path):
        ## You can use the code from the previous exmaple if you would like
        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 in the jsons as dicts based on Teams thread feedback
        with open('results/kvdb/people.json', 'r') as file:
            people_dict = json.load(file)
        with open('results/kvdb/visited.json', 'r') as file:
            visited_dict = json.load(file)
        with open('results/kvdb/sites.json', 'r') as file:
            sites_dict = json.load(file)
        with open('results/kvdb/measurements.json', 'r') as file:
            measurements_dict = json.load(file)
        self._load_db()

        # Create records by people dictionary
        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():
                # Matching up site_id by sites and visits. Then storing in
                ↪visited_v as the site
                if sites_v['site_id'] == visited_v['site_id']:
                    visited_v['site'] = sites_v

                    # Create measurements list
                    visited_v['measurements'] = []
                    for measurements_k, measurements_v in measurements_dict.
                    ↪items():
                        # Matching visit_id and person_id with measurements
                        ↪and adding to measurements_v
                        if measurements_v['visit_id'] ==
                        ↪visited_v['visit_id'] and measurements_v['person_id'] ==
                        ↪people_v['person_id']:
                            visited_v['measurements'].append(measurements_v)
                            # If there are measurements, add it to visit using
                            ↪people_v visits list
                            if len(visited_v['measurements']) != 0:
                                people_v['visits'].append(visited_v)

                    # Once person record is complete add it to the db
                    self._db.insert(people_v)

    def _load_db(self):
        self._db = TinyDB(self._db_path, indent=4, separators=(',', ': '))

```

```

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

    db = DocumentDB(db_path)

```

```

[ ]:

```