

Assign_2_1

March 29, 2021

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

import pandas as pd
import s3fs

def read_cluster_csv(file_path, endpoint_url='https://storage.budsc.
↳midwest-datascience.com'):
    s3 = s3fs.S3FileSystem(
        anon=True,
        client_kwargs={
            'endpoint_url': endpoint_url
        }
    )
    return pd.read_csv(s3.open(file_path, mode='rb'))

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)

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')
```

```
[6]: class KVDB(object):
    def __init__(self, db_path):
        self._db_path = Path(db_path)
        self._db = {}
        self._load_db()

    def _load_db(self):
        if self._db_path.exists():
            with open(self._db_path) as f:
```

```

        self._db = json.load(f)

    def get_value(self, key):
        return self._db.get(key)

    def set_value(self, key, value):
        self._db[key] = value

    def save(self):
        with open(self._db_path, 'w') as f:
            json.dump(self._db, f, indent=2)

```

```

[7]: def create_sites_kvdb():
    db = KVDB(sites_json)
    df = read_cluster_csv('data/external/tidynomicon/site.csv')
    for site_id, group_df in df.groupby('site_id'):
        db.set_value(site_id, group_df.to_dict(orient='records')[0])
    db.save()

def create_people_kvdb():
    db = KVDB(people_json)

    df = read_cluster_csv('data/external/tidynomicon/person.csv')
    for person_id, group_df in df.groupby('person_id'):
        db.set_value(person_id, group_df.to_dict(orient='records')[0])

    db.save()

def create_visits_kvdb():
    db = KVDB(visited_json)

    df = read_cluster_csv('data/external/tidynomicon/visited.csv')
    for visit_id, group_df in df.groupby('visit_id'):
        db.set_value(visit_id, group_df.to_dict(orient='records')[0])

    db.save()

def create_measurements_kvdb():
    db = KVDB(measurements_json)

    df = read_cluster_csv('data/external/tidynomicon/measurements.csv')

    for key, group_df in df.groupby(['person_id', 'visit_id', 'quantity']):
        db.set_value(str(key), group_df.to_dict(orient = 'records')[0])

```

```
db.save()
```

```
[8]: create_sites_kvdb()  
      create_people_kvdb()  
      create_visits_kvdb()  
      create_measurements_kvdb()
```