

```

import json
from pathlib import Path
import os
import pandas as pd

# Define the directory path where your CSV files are located
csv_data_dir = r'C:\Users\PN174MM\OneDrive - EY\Desktop\Personal Projects\650\dsc65

def read_local_csv(file_path):
    return pd.read_csv(file_path)

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')

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)

def create_sites_kvdb():
    db = KVDB(sites_json)
    df = read_local_csv(os.path.join(csv_data_dir, 'site.csv'))
    for site_id, group_df in df.groupby('site_id'):
        key = str(site_id) # Convert the tuple to a string
        db.set_value(key, group_df.to_dict(orient='records')[0])
    db.save()

```

```
def create_people_kvdb():
    db = KVDB(people_json)
    df = read_local_csv(os.path.join(csv_data_dir, 'person.csv'))
    for person_id, group_df in df.groupby('person_id'):
        key = str(person_id) # Convert the tuple to a string
        db.set_value(key, group_df.to_dict(orient='records')[0])
    db.save()

def create_visits_kvdb():
    db = KVDB(visited_json)
    df = read_local_csv(os.path.join(csv_data_dir, 'visited.csv'))

    # Initialize an empty dictionary to store the data
    visits_data = {}

    for index, row in df.iterrows():
        key = f"{row['visit_id']}_{row['site_id']}" # Combine values into a single
        if not pd.isnull(row['visit_id']): # Check for missing value
            visits_data[key] = {
                'visit_id': int(row['visit_id']),
                'site_id': str(row['site_id']),
                'visit_date': str(row['visit_date'])
            }

    # Set the entire visits_data dictionary as the value for visited_json
    db.set_value('visits_data', visits_data)
    db.save()

def create_measurements_kvdb():
    db = KVDB(measurements_json)
    df = read_local_csv(os.path.join(csv_data_dir, 'measurements.csv'))
    for index, row in df.iterrows():
        key = f"{row['visit_id']}_{row['person_id']}_{row['quantity']}" # Combine
        db.set_value(key, row.to_dict())
    db.save()

create_sites_kvdb()
create_people_kvdb()
create_visits_kvdb()
create_measurements_kvdb()
```

