import pandas as pd

from faker import Faker

import random

from pymongo import MongoClient

fake = Faker()

# Define the 20 top colleges in Pune (replace with your actual list)

top\_pune\_colleges = [

    "Fergusson College",

    "Symbiosis College of Arts and Commerce",

    "College of Engineering (COEP)",

    "MIT World Peace University (MIT-WPU)",

    "Brihan Maharashtra College of Commerce (BMCC)",

    "Sir Parashurambhau College (SP College)",

    "Nowrosjee Wadia College",

    "Ness Wadia College of Commerce",

    "Pune Institute of Computer Technology (PICT)",

    "Vishwakarma Institute of Technology (VIT)",

    "Cummins College of Engineering for Women",

    "Dr. D. Y. Patil Institute of Technology, Pimpri",

    "Modern College of Arts, Science & Commerce",

    "Deccan College Post Graduate and Research Institute",

    "ILS Law College",

    "Armed Forces Medical College (AFMC)",

    "Maharashtra Institute of Technology (MIT)",

    "Sinhgad College of Engineering",

    "AISSMS College of Engineering",

    "PICT Model College"

]

# Define the complete list of amenities

all\_amenities = ["Food", "Laundry", "WiFi", "AC", "Security", "Hot Water / Geyser",

                 "Attached Bathroom", "CCTV", "Drinking Water (RO)", "Parking",

                 "Cleaning", "Fridge", "Warden"]

hostel\_data = []

for college in top\_pune\_colleges:

    for i in range(1, 21):  # Generate 20 hostels per college

        hostel = {

            "hostel\_name": f"{college[:4]} Hostel {i}",

            "college": college,

            "address": fake.address(),

            "contact\_number": fake.phone\_number(),

            "rent": random.randint(1000, 10000),

            "room\_type": random.choice(["Single", "Double", "Shared"]),

            "amenities": random.sample(all\_amenities, k=random.randint(6, len(all\_amenities))),

            "safety\_priority": random.choice(["High", "Medium", "Low"]),

            "rating": round(random.uniform(3.5, 5.0), 1),

            "distance\_to\_college": f"{round(random.uniform(0.1, 5.0), 1)} km",

            "hostel\_type": random.choice(["PG", "Hostel"])

        }

        hostel\_data.append(hostel)

hostels\_df = pd.DataFrame(hostel\_data)

hostels\_df.to\_csv("pune\_hostels.csv", index=False)  # Save to CSV (optional)

print(f"Generated synthetic dataset of {len(hostels\_df)} hostels for 20 colleges.")

# --- MongoDB Atlas Insertion ---

try:

    # Replace with your MongoDB Atlas connection string

    CONNECTION\_STRING = "mongodb+srv://sakshi:gaurinde@cluster0.vpbqv.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0"

    client = MongoClient(CONNECTION\_STRING)

    db = client['lrm']  # Database name

    hostels\_collection = db['hinfo']  # Collection name

    hostels\_data\_for\_mongo = hostels\_df.to\_dict(orient='records')

    hostels\_collection.insert\_many(hostels\_data\_for\_mongo)

    print("Data successfully inserted into MongoDB Atlas (lrm.hinfo).")

except Exception as e:

    print(f"An error occurred while connecting to or inserting data into MongoDB Atlas: {e}")

finally:

    if 'client' in locals():

        client.close()