```
# File: Project3.py
# Student: Anusha Mittal
# UT EID: am98968
# Course Name: CS303E
#
# Date Created: 17th April
# Date Last Modified: 19th April
# Description of Program: Finding Data from Cities Data Set

import csv

# Reading the data from the file and create a dictionary
city_data = {}
census2020_total = 0
estimated2023_total = 0
with open('citiesData.csv', newline='') as csvfile:
    reader = csv.reader(csvfile)
    for row in reader:
        if row[0].startswith('#'):
            continue
        city_name = row[3]
        census2020 = int(row[1])
        estimated2023 = int(row[0])
        city_data[city_name] = (census2020, estimated2023)
        census2020_total += census2020
        estimated2023_total += estimated2023

# total population for Texas
city_data['Texas'] = (census2020_total, estimated2023_total)

# Creating a list of all city names in alphabetical order
city_names = list(city_data.keys())
city_names.remove('Texas')
city_names.sort()

#query processing system
print("\033[1mWelcome to the Texas Cities Population Dashboard.\033[0m")
print("This provides census data from the 2020 census and")
print("estimated population data in Texas as of 2023.")
print("\nCreating dictionary from file: citiesData.csv\n")
print("Enter any of the following commands:")
print("\033[1mHelp\033[0m - list available commands;")
print("\033[1mQuit\033[0m - exit this dashboard;")
print("\033[1mCities\033[0m - list all Texas cities;")
print("\033[1mCensus <cityName>/Texas\033[0m - population in 2020 census by
specified city or statewide;")
print("\033[1mEstimated <cityName>/Texas\033[0m - estimated population in 2023 by
specified city or statewide.")
print("\033[1mGrowth <cityName>/Texas\033[0m - percent change from 2020 to 2023, by
city or statewide.")

while True:
    command = input("\nEnter a command: ")
    if command.lower() == 'help':
        print("Enter any of the following commands:")
        print("\033[1mHelp\033[0m - list available commands;")
        print("\033[1mQuit\033[0m - exit this dashboard;")
        print("\033[1mCities\033[0m - list all Texas cities;")
        print("\033[1mCensus <cityName>/Texas\033[0m - population in 2020 census by
specified city or statewide;")
        print("\033[1mEstimated <cityName>/Texas\033[0m - estimated population in
2023 by specified city or statewide.")
        print("\033[1mGrowth <cityName>/Texas\033[0m - percent change from 2020 to
2023, by city or statewide.")
        pass

    elif command.lower() == 'quit':
        print("\033[1mThank you for using the Texas Cities Population Database
Dashboard.  Goodbye!\033")
        break

    elif command.lower() == 'cities':
        #Implementing the Cities command
        print("Texas cities:")
        city_count = 0
        for city in city_names:
            if city_count % 10 == 0:
                print()
            print(city, end=", ")
            city_count += 1
        print()
        pass


    elif command.lower().startswith('census'):
            input_list = command.split()
            census_index = input_list.index("census")
            city_name = input_list[census_index + 1]
            if city_name.lower() == 'texas':
                pop = city_data['Texas'][0]
                print(f"The population of Texas in the 2020 census was
{pop:,}".replace(",", ""))
            elif city_name in city_data:
                pop = city_data[city_name][0]
                print(f"The population of {city_name} in the 2020 census was
{pop:,}".replace(",", ""))
            else:
                print(f"City {city_name} is not recognized.")
                pass



    elif command.lower().startswith('estimated'):
        input_list = command.split()
        est_index = input_list.index("estimated")
        city_name = input_list[est_index + 1]
        if city_name.lower() == 'texas':
            pop = city_data['Texas'][1]
            print(f"Texas' estimated population in 2023: {pop:,}".replace(",", ""))
        elif city_name in city_data:
            pop = city_data[city_name][1]
            print(f"{city_name}'s estimated population in 2023:
{pop:,}".replace(",", ""))
        else:
            print(f"City {city_name} is not recognized.")
            pass

    elif command.lower().startswith('growth'):
        input_list = command.split()
        growth_index = input_list.index("growth")
        city_name = input_list[growth_index + 1]
        if city_name.lower() == 'texas':
            pop_c = (((city_data['Texas'][1]) -
(city_data['Texas'][0]))/(city_data['Texas'][1]))*100
            pop_c = round(pop_c, 2)
            print(f"Texas had a percent population change from 2020 to 2023:
{pop_c:.2f}%")
        elif city_name in city_data:
            pop_c = (((city_data[city_name][1]) -
(city_data[city_name][0]))/(city_data[city_name][1]))*100
            pop_c = round(pop_c, 2)
            print(f"{city_name} had a percent population change from 2020 to 2023:
{pop_c:.2f}%")
        else:
            print(f"City {city_name} is not recognized.")

    else:
        print("Command is not recognized.  Try again!")
```