

```
import pandas as pd
import statistics as stat
import matplotlib.pyplot as plt

In [3]: def print_stats(county,data, state, data_type, county):
    Prints mean, median, and mode about the
    given dataset.
    param data: A list containing the data.
    param county: The name of the country
    param state: The name of the country
    param data_type: What type of data is
    contained in data.
    """
    print("%s %s county weekly %s mean: %5.2f" % (state, county, data_type, stat.mean(data)))
    print("%s %s county weekly %s median: %5.2f" % (state, county, data_type, stat.median(data)))
    print("%s %s county weekly %s mode: %5.2f" % (state, county, data_type, stat.mode(data)))

In [3]: def print_stats(data, country, data_type):
    Prints mean, median, and mode about the
    given dataset.
    param data: A list containing the data.
    param country: The name of the country
    param data_type: What type of data is
    contained in data.
    """
    print("%s weekly %s mean: %5.2f" % (country, data_type, stat.mean(data)))
    print("%s weekly %s median: %5.2f" % (country, data_type, stat.median(data)))
    print("%s weekly %s mode: %5.2f" % (country, data_type, stat.mode(data)))

In [4]: base = pd.read_csv("../Data/base_set_joined.csv")

# get state population
nc_cases = base[base["State"] == "NC"]
nc_population = nc_cases["population"].sum()

# get start and end indices
start_index = nc_cases.columns.get_loc("cases 2022-06-01")
end_index = nc_cases.columns.get_loc("cases 2022-12-31")

# get a df with the dates of the weeks
weeks = []
for i in range(start_index, end_index):
    if i % 7 == 0:
        numeric_data = nc_cases.columns[i]
        numeric_data = numeric_data[5:]
        weeks.append(numeric_data)
    i += 1
curr = 0

nc_weekly_cases_df = []
for i in range(start_index, end_index):
    if i % 7 == 0:
        curr_col = nc_cases.columns[i]
        prev_col = nc_cases.columns[i-7]
        curr_week = nc_cases[curr_col].sum()
        prev_week = nc_cases[prev_col].sum()
        nc_weekly_cases_df.append(curr_week-prev_week)
        curr += (curr_week - prev_week)
    i += 1

# get mean median and mode
print_stats(nc_weekly_cases_df, "North Carolina", "cases")
North Carolina weekly cases mean: 2847.00
North Carolina weekly cases median: 2046.00
North Carolina weekly cases mode: 0.00

In [5]: base = pd.read_csv("../Data/base_set_joined.csv")
nc_deaths = base[base["State"] == "NC"]

# get start and end indices
start_index = nc_deaths.columns.get_loc("deaths 2022-06-01")
end_index = nc_deaths.columns.get_loc("deaths 2022-12-31")

curr = 0

nc_weekly_deaths_df = []
for i in range(start_index, end_index):
    if i % 7 == 0:
        curr_col = nc_deaths.columns[i]
        prev_col = nc_deaths.columns[i-7]
        curr_week = nc_deaths[curr_col].sum()
        prev_week = nc_deaths[prev_col].sum()
        nc_weekly_deaths_df.append(curr_week-prev_week)
        curr += (curr_week - prev_week)
    i += 1

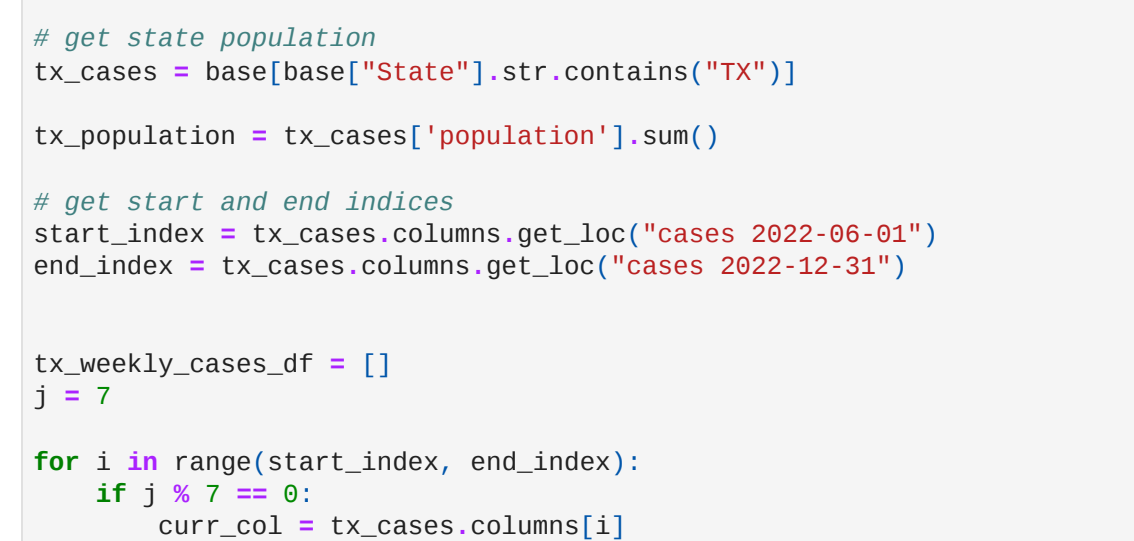
nc_deaths_normalized = []
for data in nc_weekly_deaths_df:
    nc_deaths_normalized.append(data / nc_population * 100000)

# get mean median and mode
print_stats(nc_weekly_deaths_df, "North Carolina", "deaths")
North Carolina weekly deaths mean: 86.00
North Carolina weekly deaths median: 46.00
North Carolina weekly deaths mode: 0.00

In [6]: # NC Normalized Data

# normalize US cases data and log the data
nc_cases_normalized = []
for data in nc_weekly_cases_df:
    nc_cases_normalized.append(data / nc_population * 100000)

plt.plot(weeks, nc_cases_normalized)
plt.xlabel("weeks")
plt.ylabel("New Cases")
plt.title("NC New Cases Per 100000")
plt.xticks(weeks[::2], rotation=45)
plt.show()
```



```
In [7]: base = pd.read_csv("../Data/base_set_joined.csv")

# get state population
tx_cases = base[base["State"] == "TX"]
tx_population = tx_cases["population"].sum()

# get start and end indices
start_index = tx_cases.columns.get_loc("cases 2022-06-01")
end_index = tx_cases.columns.get_loc("cases 2022-12-31")

tx_weekly_cases_df = []
for i in range(start_index, end_index):
    if i % 7 == 0:
        curr_col = tx_cases.columns[i]
        prev_col = tx_cases.columns[i-7]
        curr_week = tx_cases[curr_col].sum()
        prev_week = tx_cases[prev_col].sum()
        tx_weekly_cases_df.append(curr_week-prev_week)
        curr += (curr_week - prev_week)
    i += 1

tx_cases_normalized = []
for data in tx_weekly_cases_df:
    tx_cases_normalized.append(data / tx_population * 100000)

# get mean median and mode
print_stats(tx_weekly_cases_df, "Texas", "cases")
Texas weekly cases mean: 27495.00
Texas weekly cases median: 8.00
Texas weekly cases mode: 0.00

In [8]: base = pd.read_csv("../Data/base_set_joined.csv")
tx_deaths = base[base["State"] == "TX"]

# get state population
tx_deaths = base[base["State"] == "TX"]
tx_population = tx_deaths["population"].sum()

# get start and end indices
start_index = tx_deaths.columns.get_loc("deaths 2022-06-01")
end_index = tx_deaths.columns.get_loc("deaths 2022-12-31")

curr = 0

tx_weekly_deaths_df = []
for i in range(start_index, end_index):
    if i % 7 == 0:
        curr_col = tx_deaths.columns[i]
        prev_col = tx_deaths.columns[i-7]
        curr_week = tx_deaths[curr_col].sum()
        prev_week = tx_deaths[prev_col].sum()
        tx_weekly_deaths_df.append(curr_week-prev_week)
        curr += (curr_week - prev_week)
    i += 1

# normalize data for later
tx_deaths_normalized = []
for data in tx_weekly_deaths_df:
    tx_deaths_normalized.append(data / tx_population * 100000)

# get mean median and mode
print_stats(tx_weekly_deaths_df, "Texas", "deaths")
Texas weekly deaths mean: 88.00
Texas weekly deaths median: 83.00
Texas weekly deaths mode: 0.00

In [9]: base = pd.read_csv("../Data/base_set_joined.csv")

# get state population
fl_cases = base[base["State"] == "FL"]
fl_population = fl_cases["population"].sum()

# get start and end indices
start_index = fl_cases.columns.get_loc("cases 2022-06-01")
end_index = fl_cases.columns.get_loc("cases 2022-12-31")

fl_weekly_cases_df = []
for i in range(start_index, end_index):
    if i % 7 == 0:
        curr_col = fl_cases.columns[i]
        prev_col = fl_cases.columns[i-7]
        curr_week = fl_cases[curr_col].sum()
        prev_week = fl_cases[prev_col].sum()
        fl_weekly_cases_df.append(curr_week-prev_week)
        curr += (curr_week - prev_week)
    i += 1

fl_cases_normalized = []
for data in fl_weekly_cases_df:
    fl_cases_normalized.append(data / fl_population * 100000)

# get mean median and mode
print_stats(fl_weekly_cases_df, "Florida", "cases")
Florida weekly cases mean: 37359.00
Florida weekly cases median: 30505.00
Florida weekly cases mode: 0.00

In [10]: base = pd.read_csv("../Data/base_set_joined.csv")

# get state population
fl_deaths = base[base["State"] == "FL"]

# get start and end indices
start_index = fl_deaths.columns.get_loc("deaths 2022-06-01")
end_index = fl_deaths.columns.get_loc("deaths 2022-12-31")

curr = 0

fl_weekly_deaths_df = []
for i in range(start_index, end_index):
    if i % 7 == 0:
        curr_col = fl_deaths.columns[i]
        prev_col = fl_deaths.columns[i-7]
        curr_week = fl_deaths[curr_col].sum()
        prev_week = fl_deaths[prev_col].sum()
        fl_weekly_deaths_df.append(curr_week-prev_week)
        curr += (curr_week - prev_week)
    i += 1

fl_deaths_normalized = []
for data in fl_weekly_deaths_df:
    fl_deaths_normalized.append(data / fl_population * 100000)

# get mean median and mode
print_stats(fl_weekly_deaths_df, "Florida", "deaths")
Florida weekly deaths mean: 295.00
Florida weekly deaths median: 351.00
Florida weekly deaths mode: 0.00

In [11]: base = pd.read_csv("../Data/base_set_joined.csv")

# get state population
va_cases = base[base["State"] == "VA"]
va_population = va_cases["population"].sum()

# get start and end indices
start_index = va_cases.columns.get_loc("cases 2022-06-01")
end_index = va_cases.columns.get_loc("cases 2022-12-31")

va_weekly_cases_df = []
for i in range(start_index, end_index):
    if i % 7 == 0:
        curr_col = va_cases.columns[i]
        prev_col = va_cases.columns[i-7]
        curr_week = va_cases[curr_col].sum()
        prev_week = va_cases[prev_col].sum()
        va_weekly_cases_df.append(curr_week-prev_week)
        curr += (curr_week - prev_week)
    i += 1

va_cases_normalized = []
for data in va_weekly_cases_df:
    va_cases_normalized.append(data / va_population * 100000)

# get mean median and mode
print_stats(va_weekly_cases_df, "Virginia", "cases")
Virginia weekly cases mean: 4829.00
Virginia weekly cases median: 1885.00
Virginia weekly cases mode: 4596.00

In [12]: base = pd.read_csv("../Data/base_set_joined.csv")

# get state population
va_deaths = base[base["State"] == "VA"]

# get start and end indices
start_index = va_deaths.columns.get_loc("deaths 2022-06-01")
end_index = va_deaths.columns.get_loc("deaths 2022-12-31")

curr = 0

va_weekly_deaths_df = []
for i in range(start_index, end_index):
    if i % 7 == 0:
        curr_col = va_deaths.columns[i]
        prev_col = va_deaths.columns[i-7]
        curr_week = va_deaths[curr_col].sum()
        prev_week = va_deaths[prev_col].sum()
        va_weekly_deaths_df.append(curr_week-prev_week)
        curr += (curr_week - prev_week)
    i += 1

va_deaths_normalized = []
for data in va_weekly_deaths_df:
    va_deaths_normalized.append(data / va_population * 100000)

# get mean median and mode
print_stats(va_weekly_deaths_df, "Virginia", "deaths")
Virginia weekly deaths mean: 54.00
Virginia weekly deaths median: 57.00
Virginia weekly deaths mode: 55.00

In [13]: # Cases
plt.plot(weeks, nc_cases_normalized, label="NC")
plt.plot(weeks, va_cases_normalized, label="VA")
plt.plot(weeks, fl_cases_normalized, label="FL")
plt.plot(weeks, tx_cases_normalized, label="TX")
plt.xlabel("Weeks")
plt.ylabel("New Cases")
plt.title("States New Cases Per 100000")
plt.xticks(weeks[::2], rotation=45)
plt.legend()
plt.show()

# Deaths
plt.plot(weeks, nc_deaths_normalized, label="NC")
plt.plot(weeks, va_deaths_normalized, label="VA")
plt.plot(weeks, fl_deaths_normalized, label="FL")
plt.plot(weeks, tx_deaths_normalized, label="TX")
plt.xlabel("Weeks")
plt.ylabel("New Deaths")
plt.title("States New Deaths Per 100000")
plt.xticks(weeks[::2], rotation=45)
plt.legend()
plt.show()
```

