

```
"""
```

```
Part A: Dashboard
```

```
Rena Liu
```

```
"""
```

```
# Configure Libraries
```

```
import numpy as np
```

```
import pandas as pd
```

```
from datetime import datetime
```

```
# Only for when running on Google Colab: Mounting dataset from personal drive
```

```
# from google.colab import drive
```

```
# drive.mount('/content/drive', force_remount=True)
```

```
Mounted at /content/drive
```

```
def run_dashboard():
```

```
    """
```

```
    Function to pull dataset, provide an overview of the dataset,  
    and output dashboard data for 6/16/2021
```

```
    """
```

```
# Read in csv as dataframe
```

```
data = pd.read_csv('PA_singlestate_timeseries.csv')
```

```
# Dataset overview
```

```
#####
```

```
print()
```

```
print('-----')
```

```
print('Dataset Overview')
```

```
print('-----')
```

```
# Pull first row's state value and earliest date value, splitting
```

```
# description over multiple lines to fit border lines
```

```
print('\t* This dataset contains Covid metrics for the state of')
```

```
print('\t ', data.loc[0]['state'], 'from the timeframe of',
```

```
      np.min(data['date']), 'to', np.max(data['date']))
```

```
print()
```

```
print('\t* More specifically, Covid metrics include data on')
```

```
print('\t cases, deaths, test results, hospital capacity,')
```

```
print('\t vaccinations, and calculated metrics like')
```

```
print('\t case density and risk, among others')
```

```
print()
```

```
print('\t* Missing values in the dataset are denoted as')
```

```
print('\t empty values, "Not available", 0, "N/A", and null')
```

```
print()
```

```
# Count the number of duplicate entries with pandas duplicated() function
```

```
print('\t* Number of Duplicate Entries:', data.duplicated().sum())
```

```
print()
```

```
print()
```

```

# Covid Information
#####
# Set date to be June 16th, 2021.
date_selected = '2021-06-16'
row_selected = data.loc[data['date']== date_selected]

# Also get first date in dataset
first_day = data.loc[0]

# Dictionary to convert State Abbreviation to Full State Name
state_names = {"AL":"Alabama", "AK":"Alaska", "AZ":"Arizona", "AR":"Arkansas", "CA":"California", "CO":"Colorado", "CT":"Connecticut", "DE":"Delaware", "FL":"Florida", "GA":"Georgia", "HI":"Hawaii", "IL":"Illinois", "IN":"Indiana", "IOWA":"Iowa", "KS":"Kansas", "KY":"Kentucky", "LA":"Louisiana", "MA":"Massachusetts", "MD":"Maryland", "ME":"Maine", "MI":"Michigan", "MN":"Minnesota", "MO":"Missouri", "MS":"Mississippi", "MT":"Montana", "NE":"Nebraska", "NH":"New Hampshire", "NJ":"New Jersey", "NM":"New Mexico", "NY":"New York", "NC":"North Carolina", "ND":"North Dakota", "OH":"Ohio", "OK":"Oklahoma", "OR":"Oregon", "PA":"Pennsylvania", "RI":"Rhode Island", "SC":"South Carolina", "SD":"South Dakota", "TN":"Tennessee", "TX":"Texas", "UT":"Utah", "VA":"Virginia", "VT":"Vermont", "WA":"Washington", "WI":"Wisconsin", "WY":"Wyoming"}

# Print Dashboard
print('-----')
print('Covid Dashboard')
print('-----')
# Get state value
print(state_names[row_selected['state']].values[0], 'on',
      datetime.strptime(date_selected, '%Y-%m-%d').strftime('%B %d, %Y'))
print()

# Risk Level
# We also chose to add a covid risk level metric to the dashboard, as it is a quick,
# one metric summary of multiple covid metrics to inform the user of
# "how well" the state is doing

# Else-if block to convert the numeric risk value into a verbal description
risk_val = row_selected['riskLevels.overall'].values[0]
risk = ''
if risk_val == 1:
    risk = 'Very Low'
elif risk_val == 2:
    risk = 'Low'
elif risk_val == 3:
    risk = 'Medium'
elif risk_val == 4:
    risk = 'High'
elif risk_val == 5:
    risk = 'Very High'
print('Covid Risk Level :', risk)
print()
# Splitting explanation of this metric calculation over multiple lines to fit border
print('(Risk level is graded on a 5-level scale of Very Low,')
print('Low, Medium, High, Very High based off the 1. case density,')
print('2. infection rate, and 3. positive test rate of this day)')
print()

print('Infections (Confirmed or Suspected):')
# Pulling and printing new cases for the selected date, formatted to include comma
# for the thousands place and as an integer (with no decimal place for units of people)
print('\t* New Cases:'.format(row_selected['actuals.newCases'].values[0].last))

```

```

print('\t* New Cases:', '{:,}'.format(row_selected['actuals.newCases'].values[0]).ab
# New deaths
print('\t* New Deaths:', '{:,}'.format(row_selected['actuals.newDeaths'].values[0]).ab
# Case density, aka cases per 100k population calculated using a 7-day rolling average
print('\t* Case Density:', row_selected['metrics.caseDensity'].values[0])
print()
# Splitting explanation of this metric calculation over three lines to fit border length
print('(Case Density is defined as the number of cases per 100k people')
print('over a 7-day rolling average)')
print()

# Pull and print first day in the dataset to give background to the cumulative metrics
print('Since', datetime.strptime(first_day['date'], '%Y-%m-%d').strftime('%B %d, %Y'))
# Cumulative cases
print('\t* Cumulative Cases', '{:,}'.format(row_selected['actuals.cases'].values[0]).ab
# Cumulative deaths
print('\t* Cumulative Deaths', '{:,}'.format(row_selected['actuals.deaths'].values[0]).ab
print()

print('Vaccinations (Ratio of the Population):')
# Pull and print ratio of completed and initiated vaccinations, formatted as a percentage
print('\t* Completed vaccinations:', '{:.1%}'.format(np.round(row_selected['metrics.comple
print('\t* Initiated one vaccination:', '{:.1%}'.format(np.round(row_selected['metrics.one
print('-----')
print()

```

run_dashboard()

Dataset Overview

- * This dataset contains Covid metrics for the state of PA from the timeframe of 2020-03-02 to 2021-06-25
- * More specifically, Covid metrics include data on cases, deaths, test results, hospital capacity, vaccinations, and calculated metrics like case density and risk, among others
- * Missing values in the dataset are denoted as empty values, "Not available", 0, "N/A", and null
- * Number of Duplicate Entries: 20

Covid Dashboard

Pennsylvania on June 16, 2021

Covid Risk Level : Very Low

(Risk level is graded on a 5-level scale of Very Low,

Low, Medium, High, Very High based off the 1. case density,
2. infection rate, and 3. positive test rate of this day)

Infections (Confirmed or Suspected):

- * New Cases: 279
- * New Deaths: 16
- * Case Density: 2.9

(Case Density is defined as the number of cases per 100k people
over a 7-day rolling average)

Since March 02, 2020 :

- * Cumulative Cases 1,214,051
- * Cumulative Deaths 27,582

Vaccinations (Ratio of the Population):

- * Completed vaccinations: 47.2%
 - * Initiated one vaccination: 61.1%
-