

# CS 602 Summer 2021 - Final Project

## Building a Data-Driven Python Application

### Your Challenge

In this last project you will develop an interactive data-driven web-based Python application that tells a story with real-world data. You will show your mastery of many coding concepts as you interact with data real-world data. You will use Pandas and NumPy modules for managing and interacting with data, Matplotlib or Pandas charts for plotting, and the Streamlit.io package for creating interactive web applications using Python.

In addition, you need to use another Python module and it's given functions in your code. This will require you to find and research a Python module of your choosing that we have not covered during the semester. Refer to [PyPI · The Python Package Index](https://pypi.org) online at <http://pypi.org>.

Finally, you will use [Streamlit Sharing](#) to publish your application to a web server so others can access it online.

### Tell a Story with Real-World Data

Choose one of these data sets:

Boston Crime Incident Reports 2021	<a href="#">Analyze Boston</a> (data.boston.gov)	<a href="#">Download CSV</a> (7000 sample records). Also <a href="#">download this file</a> containing names of Boston Police Districts.
California Fire Incidents, 2013 to 2020	<a href="https://www.kaggle.com/ananthu017/california-wildfire-incidents-20132020">https://www.kaggle.com/ananthu017/california-wildfire-incidents-20132020</a>	<a href="#">Download CSV</a> (from data.gov)
New York City Vehicle Collisions, 2015 to present	<a href="#">Description</a>	<a href="#">Download CSV</a> (sample data)
Volcanic Eruptions	<a href="#">Description</a>	<a href="#">Download CSV</a> (cleaned data)
Used cars for sale on Craigslist	<a href="#">Description</a>	<a href="#">Download CSV</a> (sampled data)
Skyscrapers around the World	<a href="#">Description</a>	<a href="#">Download CSV</a> (adapted from skyscrapercenter.com)

To ensure students create a variety of projects, you will sign up to use the data set you wish during class. If you miss class, or if the signups are not approximately equally distributed, I will assign a data set for you to use.

### Demonstrate Your Python Coding Skills

Your Python code should demonstrate your Python coding skills as you implement several concepts that we studied throughout the course that appropriate for your project, such as:

- Coding Fundamentals: data types, if statements, loops, formatting, etc.
- Data Structures: Interact with Lists, Tuples, Dictionaries (keys, values, items)

- Functions: passing positional and optional arguments, returning values
- Files: Reading data from a CSV File into a DataFrame
- Statistics or Pandas module functions for calculating mean, median, etc.
- Matplotlib or Pandas for creating different types of charts
- Streamlit.io for making interactive applications, displaying charts and maps using UI widgets for input
- Num.py functions for interacting with arrays (such as np.arange)
- Pandas DataFrames for interacting and manipulating large data sets using filtering, sorting, pivot tables, etc.

## Assignment Details

### Part 1. Design: Tell a Story with your Data

The purpose of this part is to get you thinking about what you might do before you start coding. Identify at least two different queries or questions you can ask about your data set and ways to interact with and present the data based on your understanding of Pandas DataFrames, Matplotlib, and the Streamlit.io packages. Use a combination of charts, graphs maps, word clouds, or other presentation tools.

Describe how your queries will be interactive by incorporating Streamlit's user interface elements to obtain user input. Describe how you will visually present this data using charts, graphs, Streamlit tables or maps. For example, if analyzing housing data, you might use a dropdown list to specify a list of neighborhoods and a slider to specify a price range. You then might display all rooms for rent in that neighborhood within that price range using a table, chart, or map. (That's an easy one. At least one of your queries needs to be more complex than this!)

Try to make your page is "user friendly" and as professional-appearing as possible. Include labels on all controls requiring user interaction, make sure your charts have titles, legends or explanations that would be helpful to the user. Don't use default display values, but customize your charts so they look nice! Streamlit has several new layout and styling features that you can use to make professional looking web applications. Things to do:

1. [Sign up on this Google Spreadsheet](#) for the data set you would like to work with. (Be sure to select the correct tab at the bottom for our section of CS 602). Please do not erase someone else's name after they sign up! Try to keep each column balanced with approximately the same number of students using each data set.
2. Create a Word document describing your plans. **Submit it on Blackboard AND email it to me at my Bentley email address.** I will respond by email approving your proposed questions or making suggestions if they appear to be too complicated or too easy.

You may change your queries or visualizations after you start coding if you need to change your plans. If you do this, please notify me during the coding week.

Feel free to add to your project as you explore Pandas and Streamlit capabilities and find cool ways to implement new or additional features. **Part of your grade will be a "complexity/originality" score.** If you do something cool that we may not have discussed in class or implement more than the minimum requirements, you will receive a higher score. A zero complexity score means you implemented the minimum requirements for this project.

## Part 2. Code.

Create your Python application with a Streamlit UI, various visualizations and Python module functions you have found, and researched. Create at least two different charts, graphs of different types with custom legends, axis labels, tick marks, colors, other features), or a map showing latitude and longitude. Be sure to include appropriate context or labels in your user interface to cue the reader about which values to specify, and the purpose of each chart or graph. You may wish to add a few sentences explaining each chart. Place all UI controls in the left sidebar, and your visualizations in the main content area. Make your application as professional looking as you can.

Post your code to Blackboard before the start of our last class. If your project contains more than one file (such as code in a .py file, and additional files for images, please upload a ZIP file containing everything.)

## Coding Checklist

As you write your code, be sure to demonstrate your mastery of these capabilities in your project:

- At least one function that has two parameters and returns a value
- At least one function that does not return a value
- Interacting with dictionaries, lists, and tuples
- Using a Python module to calculate a statistical function such as average, median, mode, etc.
- User Interface and dashboard with Streamlit.io

Your code should demonstrate your mastery of at least three Pandas capabilities as appropriate for your queries and data. These include:

- Sorting data in ascending or descending order, multi-column sorting
- Filtering data by one or more conditions
- Analyzing data with pivot tables
- Managing rows or columns
- Add/drop/select/create new/group columns, frequency count, other features as you wish

There are a lot of Python modules that have been developed by other Python programmers that we have not covered during this semester. Your code needs to use some other Python functions found in another Python modules. What you decide to use will be determined on how you plan to use a given dataset and how you plan on displaying that final data via the Streamlit UI. It will require:

- Find a module that has functions you will find useful to manipulating or displaying your dataset visualizations.
- Research the module and determine what functions you can use.
- Use the functions in your final coding project.

Usual rules about writing "good" code apply:

- Make your code as modular and easy to follow as possible
- Include a docstring, comments, and meaningful variable names.
- If you did something "cool" in your code that you are incredibly proud of, please write a comment call attention to what you did.
- If you referred to any online articles or other information beyond class examples, please be sure to list them as references in your code.
- Make sure the program runs and the output is correct.

## Documentation String

Use this documentation string at the top of your Python code file:

```
"""
Name:    Your Name
CS230:   Section XXX
Data:    Which data set you used
URL:     Link to your web application online (see extra credit)

Description:

This program ... (a few sentences about your program and the queries and charts)
```

## Part 3. Present.

Plan to present your project in class during the last day, showing both a demonstration of your project running in the browser and then describing at least one section of the code that you wrote of which you are most proud! Show what you feel is the most interesting part of your project. Then display the code and explain the pandas and Streamlit code well enough to convince me that you understand how your code works and what you did.

If you are unable to present the project in class, you may request permission to create a video presentation in advance. I will ask you to review as many projects of your classmates as possible during this time, and offer some minimal amount of extra credit for doing so (to be added to an earlier quiz score). Hey, it can't hurt!

## Part 4. Publish

Publish your application to the web by following these [Streamlit Sharing](#) instructions. This is a newly released feature. **It may take a few days before your request is filled, so sign up for the invite now!**

When you successfully publish your web page online, please enter your name and your site's web address (URL) as the next available blank row at the bottom of [this Google spreadsheet](#) on the tab for our class.

## Grading

This project counts toward 15% of your final course grade for the course and is based on 60 points, as follows:

Requirement	Points
Project: Proposal, Design and Queries submitted on time	4
Design: User Interface – at least three Streamlit UI controls (page has a professional appearance)	9
Design: Identified, selected and researched a Python package that we have NOT covered during the semester.	2
Coding: Use of function(s) from the Python module you selected and researched	5
Coding: At least one function with at least two parameters that returns a value	2
Coding: At least one function with a default parameter called more than once in your application (one time with the default value, and one time without) <sup>1</sup>	2
Coding: Interacting with a Dictionary <sup>2</sup>	3
Coding: Interacting with Lists or List Comprehensions <sup>3</sup>	3
Coding: Pandas Features – at least three (sort, filter, multiple conditions filter, pivot table, etc.)	12
Coding: At least two different charts with custom legends, labels, tick marks, titles, colors, other features. At most one can be a map <sup>4</sup>	10
Coding: Well documented, efficient, modular	2
Project: Complexity compared to other student projects 0 = Your project meets the minimum requirements 1 = Your project somewhat exceeds the minimum requirements or includes additional queries, charts, or UI features 2 = Your project far exceeds the minimum requirements or includes additional queries, charts, or UI features 3 = You went above and beyond by including features, modules, or packages learned independently or not described in class	3
Publish: Publish with Streamlit Sharing	3
<b>Total:</b>	<b>60</b>

## Project Presentation

Your project presentation counts 5% toward your final grade. You have 5-7 minutes to impress us. Please demo your project, select one of the more interesting pieces of code that you wrote, and explain it to us. I will ask you a question about your code, so anything you write is fair game for me to ask you about.

---

<sup>1</sup> For example, you might write a function to display a chart with a default color (such as black) if no color value is specified. Or you can call the function by passing in a value to specify a different color.

<sup>2</sup> Hardcoding values in a dictionary does not count. Please demonstrate that you can use methods and loops to interact with items in a dictionary.

<sup>3</sup> Hardcoding values in a list does not count. Please demonstrate that you can use methods and loops to interact with items in a list, or write a list comprehension.

<sup>4</sup> You may display a DataFrame on your webpage, but that doesn't count as a chart.

## Getting Help:

- This is a final project, so please do not discuss your program with anyone other than your instructor.
- You can ask CIS Sandbox tutors for assistance on related or general topics, but you cannot ask them to help you write your code for this project. For example, you can ask tutors to help review examples of how to create bar charts in Python (in general), but you cannot ask them to help you debug a bar chart you might create using the data set for this project.
- You can ask CIS Sandbox tutors for help with fixing syntax or runtime errors.