Bentley University Fall 2022
CS 230
Introduction to Programming with Python

FINAL PROJECT – INTERACTIVE DATA-EXPLORER

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, PyDeck (or other mapping packages) for maps, and the Streamlit.io package for creating interactive web applications using Python.

TELL A STORY WITH REAL-WORLD DATA

Choose one of these data sets (All sampled to several thousand rows) .

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

To ensure students create a variety of projects, you will sign up for and select a data set. if the signups are not approximately equally distributed, I will assign a data set for you to use.

ASSIGNMENT DETAILS

PART 1. DESIGN – WHAT DO YOU WANT TO SHOW AND TELL 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 three 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, maps, and the Streamlit.io packages.

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!)

Be sure your page is "user friendly" and as "polished" as possible. Be sure to label controls requiring user interaction, make sure your charts have titles, legends or explanations that would be helpful to the user.

Create a Word document describing your plans. **Submit it on Blackboard.** I will grade the proposals in Blackboard within 48 hours 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 development period.

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 use a module or 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 complexity score of 1 means you implemented the minimum requirements for this project. A complexity score of zero means you didn’t meet the requirements.

PART 2. CODE.

Create your Python application with a Streamlit UI and the various visualizations. Create at least three 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.

DUE DATE:

Post your code to Blackboard one days before the final exam class period, and plan to present your project during our final exam class period.

CODING CHECKLIST

As you write your program, be sure to include code that demonstrates each of these items. Each contributes to your project grade (see the rubric below).

Python Coding: At least any four of these:

1. Functions:
    a. At least one function that has two parameters that returns a value
    b. At least one function with a default parameter that returns a value
    c. At least one function that does not return a value
2. Lists and Loops:
    a. A list comprehension
    b. A loop that iterates through items in a list, dictionary, or tuple
3. Data Structures:
    a. Code that uses at least two different methods of lists, dictionaries, or tuples.

Streamlit:

1. At least three different UI Controls (sliders, drop downs, multi-selects, text box, etc.)
2. Page design features (sidebar, fonts, colors, images, navigation)
3. Well-designed, professional-appearing, interactive website

Charts and Maps:

1. At least three different charts (well labeled)
2. Use legends, colors, labels, titles, as appropriate
3. At least one map (**st.map** is only partial credit) – for full credit, include dots, icons, or other map features

Pandas: At least and 4 of these:

1. Sorting data in ascending or descending order, by one or more columns
2. Filtering data by one condition
3. Filtering data by two or more conditions with AND or OR
4. Analyzing data with pivot tables
5. Add/drop/select/create new/group columns, frequency count, other features as you wish
6. Try doing text analysis – find common words in titles, etc, or searching for locations with a particular name.

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 / comments in your code.
- Make sure the program runs and the output is correct.

OTHER MODULES OR PACKAGES. (EXTRA CREDIT)

Python has several modules developed by other Python programmers that we have not covered during this semester. You are welcome to explore other package or modules (especially those available for visualizing data) which are available in Python's Package Index. Use one in a meaningful way in your project for extra credit. Examples include Seaborn charts, Folium maps, WordCloud, etc. See https://pypi.org/ for the complete list.

PART 3. PRESENT.

Please present your project during our class final exam period 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! Demonstrate what you feel is the most interesting part of your project. Then talk through the Pandas and Streamlit code well enough to convince me that you understand how your code works and what you did.

During the presentations you will critique and evaluate the projects of your classmates. Their evaluation of your project may contribute to the complexity and presentation scores on your project.

PART 4. PUBLISH YOUR APPLICATION ONLINE (EASY EXTRA CREDIT)

Post your application to the web. Sign up for a free Streamlit Cloud community account at https://streamlit.io/cloud and follow the instructions to publish your app. Share your published link on the signup sheet.

GRADING

The Project is worth **16 percent** of your final grade. The project is based on **50 points**, as follows:

REQUIREMENTS	POINTS
Project: Proposal, Design and Queries submitted on time	2
Python Coding Features: at least four @ 2.5 points	10
Streamlit Features: three controls and other page design features	8
Charts and Maps: three charts and one map	12
Pandas Features: at least four – sort, filter, etc.	12

Coding: Well documented, efficient, modular	3
Complexity: 0 = Your project implements less than the minimum requirements 1 = Your project implements the minimum requirements 2 = Your project includes some complex queries, charts, or UI features 3 = You went above and beyond in requirements, ether doing more than what is required, or by including features, modules, or packages learned independently or not described in class	3
Total	**50**
Extra Credit: Publish to Streamlit Cloud	2
Extra Credit: Use a Python package or module we did not use in class	2
Extra Credit: Creating a 10-minute video of your entire final project which can be used for interviews and your LinkedIn page.	2

GETTING HELP:

This is a final project, so please do not discuss your program with anyone.

You can ask CIS Sandbox tutors or the instructor 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.