15-110 F26 - Data Analysis and Visualization Mini-Project

Name:

AndrewID:

In this assignment you will work with a real dataset from **NASA's Exoplanet Archive** containing measurements for confirmed exoplanets outside our solar system. You will use `pandas` to load, index, and aggregate the data, and `matplotlib` to visualize it.



Every question in this assignment allows you to use GenAI. The goal is to make sure you understand what the code does and can explain it.

For this assignment there are two submission links on Gradescope.

Project1 - Reflection - upload your responses to questions in this file as a .pdf

Project1 - Code and Process - upload your code in the given data-analysis.pdf starter file and a .pdf log of any communication you had with a GenAI. If you wrote all the code yourself with no GenAI support, upload a short .pdf saying that.

The dataset: exoplanets.csv
Part 1: Let's explore the data with pandas
Part 2: Let's explore the data visually with Matplotlib
Part 3: Bringing it together
Part 4: Cleaning the Dataset and Deriving New Columns

The dataset: `exoplanets.csv`

Download `exoplanets.csv` from the course page. Each row is one confirmed exoplanet. (Note that the data came from https://exoplanetarchive.ipac.caltech.edu/ .)

Note: like most real-world data, some numeric values are missing (NaN); not every planet has a measured mass or radius.

Column	Description
`pl_name`	The planet's official name
`hostname`	Name of the host star the planet orbits
`discoverymethod`	The technique used to detect the planet (e.g., Transit, Radial Velocity, Microlensing, Imaging)
`disc_year`	The year the planet was discovered
`pl_orbper`	The planet's orbital period, in days
`pl_rade`	The planet's radius, in Earth radii
`pl_bmasse`	The planet's "best estimate" mass, in Earth masses
`pl_eqt`	The planet's equilibrium temperature, in Kelvin (the theoretical temperature assuming no atmosphere)
`st_teff`	The host star's effective surface temperature, in Kelvin
`sy_dist`	The distance from Earth to the planetary system, in parsecs

Also download the `data-analysis.py` file. Work through the instructions below and complete each part in the code file as you reach it.

Note: Make sure your .py and .csv files are in the same directory!

Part 1: Let's explore the data with `pandas`

For the following questions:

- You can use GenAI to help you write any code, or you can write it yourself
- Copy and paste the code in the appropriate section of `data-analyis.py`.
- Then run and test the code and answer the questions below.

1. In Thonny, read `exoplanets.csv` into a DataFrame using `pandas`.

2. In Thonny, print the shape (# rows x # cols) of the DataFrame and the column names and data types. Based on the output in Thonny, answer the questions below:

How many rows does the dataset have?	
How many columns does it have?	
Which column(s) are of type integer?	

3. In Thonny, use `.head()` and `.describe()` on the DataFrame to look at the numerical (int, float) columns. Then answer the questions below:

What does the `.head()` function do?	
What does the `.describe()` function do?	
Based on the output in Thonny, what is the `mean` and `standard` deviation of the `pl_orbper` column?	mean:
	standard deviation:

4. In Thonny, use indexing to display the first 10 rows of the `pl_name` and `discoverymethod` columns only.

Based on the output in Thonny, what does the row with index 4 contain?	

Part 2: Let’s explore the data visually with Matplotlib

We will now explore the data a bit more via visualization with matplotlib.

1. Make a histogram with 100 bins of the host star's surface temperature in Kelvin for all the planets in the dataset. Copy the code into Thonny and run it. Add the histogram to the box on the next page.

2. Change the x-axis so it has the range 0 Kelvin to 20000 Kelvin. Then answer the questions below:

How did you update the code to set this range?	
Paste the code you used here.	

3. How would you change the y-axis label to “Binned planets” and the x-axis to “Temp in Kelvin”? Write the code below, no need to change it in Thonny.

4. How would you change the color of the histogram to grey? Write the code below and also modify the code in Thonny.

Upload your histogram image from #1 by clicking on the box below. If that does not work, use a PDF editing tool (like Preview or smallpdf.com/edit-pdf) to insert the image manually. Make sure to add the image in the correct location and/or delete the blank page if you do this so that Gradescope will find your image correctly.

If you have trouble getting your image into the PDF, contact the course staff for help.

Part 3: Bringing it together

Now, we will combine your pandas aggregation skills with your plotting skills to do some more exploration of this dataset. The dataset shows us different host stars with the planets that orbit them. We want to find the top two host stars with the largest number of planets orbiting them. Then we want to create a chart showing the planets grouped by their host star and their weights.

1. First, we want to use pandas to calculate how many planets orbit each host star. Print out this information in Thonny. Based on the code output in Thonny, answer the questions below:

How many unique host stars are in the database?	
Which pandas function did you call to group planets by host star?	
What does the parameter of this function represent?	

2. Now, we want to find the top two host stars with the most planets orbiting them. Copy and paste your code in Thonny and answer the following questions:

What are the top two host stars and how many planets do they each have?	

To find the top two planets, you likely had to call multiple functions and/or methods in Thonny. Below, write each function/method you called as well as a sentence describing what it does. Note: you may not need all the rows.

3. Finally, we want to create a chart showing the top two host star's planet's masses, grouped by their host star. Before you generate code, answer the following questions on your own:

How many dimensions of data are you visualizing?	
What type of data (categorical, ordinal, numerical) is each dimension above?	
What type of chart is best used to visualize this data?	

Now generate the chart and run the code in Thonny. Copy the chart in the space below.

Reflect on the chart you added to the prior page as a way verify the output of the GenAI code:

Does the visualization have the top host stars you found in Question 2 above? If so, write yes, if not, write what is missing.	
Check each host star in the chart above and count how many planets are being represented for each host star. Are these numbers consistent with your answer to Question 1 above? If so, write yes, if not, write down what inconsistencies you notice.	

Part 4: Cleaning the Dataset and Deriving New Columns

When we described the dataset at the beginning of this assignment we mentioned:

Note: like most real-world data, some numeric values are missing (NaN); not every planet has a measured mass or radius.

Real datasets rarely arrive ready to use; as we discussed in class, cleaning is a normal, often necessary first step. In particular, we frequently need to decide, based on the goals of our analysis, whether to include or remove NaN values. This choice matters most when calculating statistics like the mean or standard deviation, since NaN values can silently skew or break these calculations depending on how they're handled.

1. First, create a copy of your DataFrame that drops rows with planets that are missing a radius, orbital period and equilibrium temperature. Then answer the questions below:

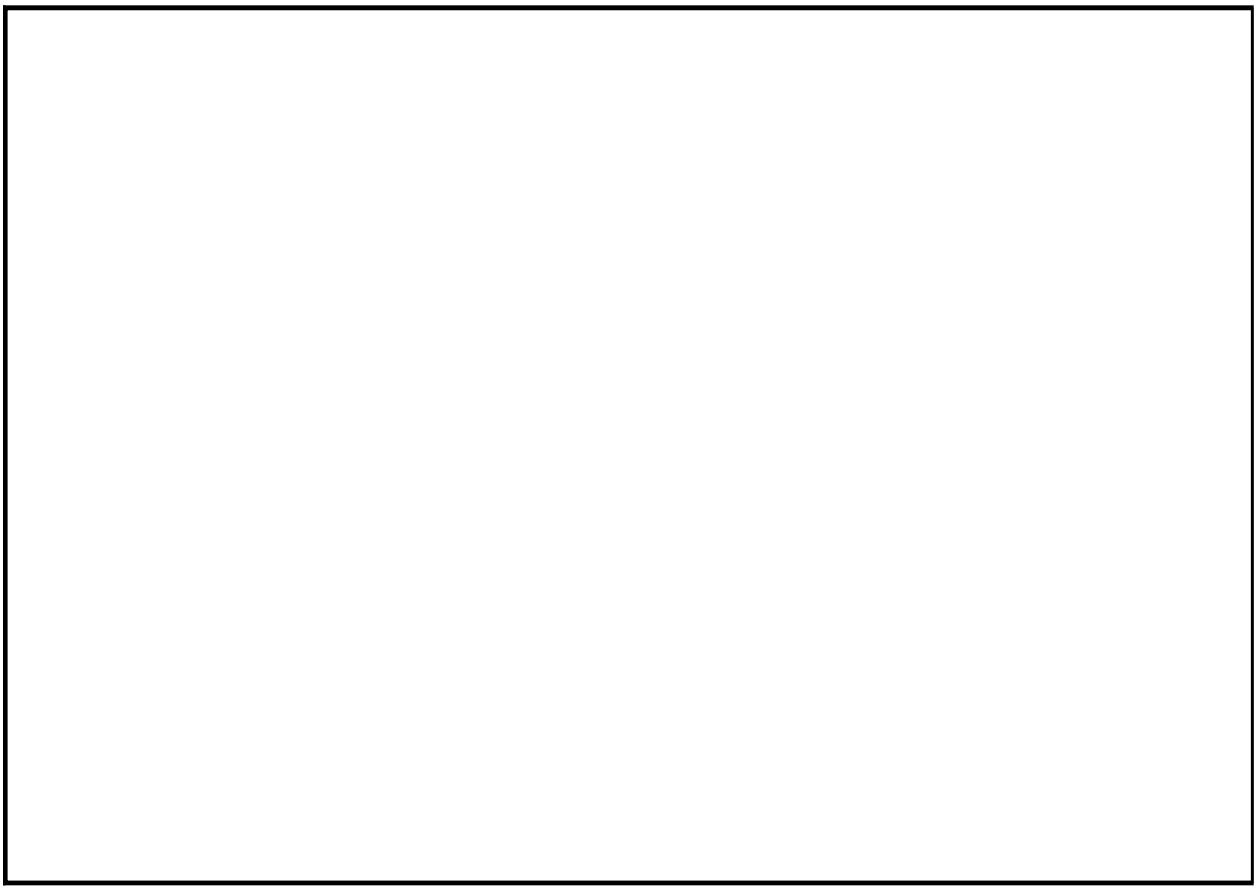
Why might we want to create a copy of the DataFrame and not change the existing one?	
How many rows and columns does this new DataFrame have?	
Using Part 1 of this assignment, how many rows had a NaN radius or weight?	

2. Now, using the cleaned DataFrame from above, we will create a new column called `size_class` that labels each planet as **small** (radius < 0.5), **medium** (radius 0.5 - 2.0 inclusive), or **large** (radius > 2.0). Use pandas to achieve this.

   In the space below, explain what function(s) you used to achieve the above, what parameters the function(s) took, what each parameter represents, etc.

3. Here is a hypothesis: planets that orbit closer to their star should generally be hotter. We want to explore this hypothesis by plotting a scatterplot of the orbital period of each planet against the planet's equilibrium temperature as well as the size of the planet we classified above represented by different colors. You do not need to include a legend.

Upload the chart that was created in Thonny below, then answer a few questions on the next page.



If you used GenAI, did it make certain choices about how to display the data that you did not ask explicitly for? If yes, explain why and what; if no, explain how the choices matched your prompt.

If you did not use GenAI, what choices did you make about how to display the data? Why did you make those choices?

To create the scatter plot, you had to use a scatter function. Explain what the parameters and keyword options of this function are in your code and what they are doing.