General Points

- Use the course material located at:
  - [Python Data Science Handbook](#)
- Assignment 12 can be completed using previously covered material and content from the following chapters:
  - 00.00-Preface *through* 03.13-Further Resources

  After completing requirements, test to ensure all cells run correctly in the .ipynb file.
- Include appropriate markdown cells to identify the requirements below by number. See this [example](#).
- Produce an .html file that shows the .ipynb after a *successful test run*.
  - by File | Download as | HTML (.html) .
- Test the .html file by opening it in a browser and ensure the content is produced correctly from the run in Jupyter Notebook.
- Submit BOTH the .ipynb and .html files to the appropriate link in Blackboard | Assignments. Submit the .html file as a .zip file to pass security settings. Submit other files individually.
- Submit any additional files required to complete the assignment.

Requirements

(*Ensure that all Requirements are complete*)

1. Using Jupyter Notebook (or similar tool), create a file named:
   - assignment-12.ipynb
2. Add an H1 markdown: "This is Assignment 12 - <yournamehere>"
3. Using the Planets dataset from Seaborn, determine the number of rows and columns in the dataset.
4. Using the head() method, show the top fifty rows in the Planets dataset.

5. Use the following seeds for RandomState() to demonstrate that the function returns the same random numbers for constant seeds:

- RandomState(42)
- RandomState(99)
- RandomState(42)
- RandomState(99)

Hint: use print(ser, '\n') four times.

6. Use rand(5) to produce the following dataframe. Naturally, values will vary.

	Texas	Georgia	Maine
0	0.565617	0.769793	0.395454
1	0.297622	0.746767	0.973956
2	0.046696	0.377439	0.524415
3	0.990627	0.494147	0.093613
4	0.006826	0.928948	0.813308

7. Output the common aggregates for the Pandas dataset dropping rows with missing values.
8. Output a list of the datasets that install with Seaborn.
9. Output the top five rows of five Pandas datasets.
10. Apply the describe() method to a planets.groupby() object.
11. Apply the describe() and unstack() methods to a planets.groupby() object.
12. Output the first ten rows from the Seaborn titanic dataset.
13. Use the groupby() method with the titanic dataset to show mean survival rates by sex and class.
14. Use the pivot_table() method with the titanic dataset to show mean survival rates by sex and class.
15. Use a multi-level pivot table to include age with the previous requirement.

16. Use the pivot_table() method to show row data by sex and column data by class when using aggfunc argument to aggregate survived by using sum and fare by mean.

17. Use matplotlib, pivot_table(), and the Seaborn births dataset to produce a plot of total births per year.

18. Create an np array x based on range(11). Output the cubed contents of x.

19. Use list comprehension with a list of string to convert all entries to uppercase. What happens when the list contains the value of None?

20. Convert the list from the previous requirement into a Pandas series. Demonstrate conversion to uppercase with the Pandas series.

21. Demonstrate the use of five Pandas str methods on a Pandas series named *states* with seven created from a list of seven names of U.S. states.

22. Demonstrate the use of three regular expression examples using the *states* Pandas series.

23. Demonstrate the use of three miscellaneous methods using the *states* Pandas series.

24. Use markdown to describe the operations demonstrated in the section *Example: Recipe Database* in chapter 03.10.

25. Create a NumPy date array using the date 01Jan2020. Use the arrange() method to create an array of seven dates starting with 01Jan2020.

26. Construct a Pandas series object and use it to demonstrate indexing data by timestamps.

27. Use a Pandas DataReader object to show the stock prices of a ticker symbol of your choice (e.g. MSFT, GS, TEAM) from 01Jan2020 to 31Mar2020.

28. Use the timeit function to compare processes summing four Pandas dataframe objects using:
   - the typical approach (simple '+' operations)
   - the pd.eval() approach

29. Watch the following three [Pandas videos](#):

30. Use markdown to include a statement at the end of assignment-12.ipynb explaining your experiences with Assignment 12. Make this authentic (minimum of 2-3 sentences).

TEST – TEST – TEST your .ipynb file to ensure all requirements are met. Produce an .html file from a *successful test run* of the .ipynb file. Ensure that the .html is produced correctly by opening it in a browser.

- Use the list above as a confirmation checklist.
- Not meeting all requirements = 0 points for the assignment.