General Points

- Use the course material located at:
    - [Whirlwind Tour of Python](#)
- Assignment 07 can be completed using previously covered material and content from the following chapters:
    - 00-Introduction *through* 15-Preview of Data Science Tools
- After completing the 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 files individually (via a multi-select). However, if your browser posts an error for the .html file, submit it as a .zip.
- 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-07.ipynb
2. Add an H1 markdown: "This is Assignment 07 - <yournamehere>"
3. Include appropriate markdown cells to identify the requirements below by number.
4. From range(30), use list comprehension to output a list with numbers that are not multiples of 3 and not equal to 13 and not equal to 17.

5. Write the operation in the previous Requirement but using loop syntax instead of list comprehension.

6. From range(30), use list comprehension to output a list with positive numbers that are even and negative numbers that are odd.

7. From range(1000), use set comprehension to output a set of numbers based on i % 5.

8. From range(20), use dictionary comprehension to output a dictionary of numbers (i) as keys and their cubes (i**3) as values.

9. From range(50), use a generator function to create a generator named G of even numbers, print the type of G, and then print the contents of G.

10. From range(15), use a generator in two ways to output n/2. For example, as G1 = (…) and G2 = gen_divby2().

11. Use a generator defined as a function to output prime numbers < 100.

12. Import the following modules from Python's Standard Library. Demonstrate a minor example using functionality/capability from each module in the list:

- `os` and `sys`: Tools for interfacing with the operating system, including navigating file directory structures and executing shell commands
- `math` and `cmath`: Mathematical functions and operations on real and complex numbers
- `itertools`: Tools for constructing and interacting with iterators and generators
- `functools`: Tools that assist with functional programming
- `random`: Tools for generating pseudorandom numbers
- `pickle`: Tools for object persistence: saving objects to and loading objects from disk
- `json` and `csv`: Tools for reading JSON-formatted and CSV-formatted files.
- `urllib`: Tools for doing HTTP and other web requests.

13. Using string methods:
- Demonstrate the conversion to upper-case and lower-case
- Convert "this is the title of my latest novel" to a title
- Removing leading and trailing spaces
- Remove leading zeros
- Return the index of a substring

- Determine if a string ends with and begins with a substring
- Replace a substring with another substring
- Split a string of words into a list of individual words
- Join a list into a string using ** as separators
- Print list elements each on its own line using join()
- Define pi = '3.14159265359' Use format() to print pi to 7 decimal places to the right of the decimal point.

14. Use the Regular Expression compile() method to create an email matcher and demonstrate the matcher with 3 valid emails and 3 invalid emails.

15. Use markdown to include a statement at the end of assignment-07.ipynb explaining your experiences with Assignment 07. 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.