Projects to work on until we meet the week of August 5th:

1. Make a dice roller using Python.
   1. Strategy: There are a few different ways to approach this program. Here is one example (from https://www.pythonforbeginners.com/code-snippets-source-code/game-rolling-the-dice):
      1. Import relevant libraries. You will need to use the library “random” here.
      2. initialize two variables: min\_value and max\_value to the lowest and highest number of dice (1 and 6 respectively).
      3. Use the randint( ) function to generate a random integer from 1 to 6.
         1. The randint( ) function takes the minimum value and maximum value of a range as its input arguments and returns a random integer within the range.
         2. We will pass the variables min\_value and max\_value to the randint( )  function to imitate dice rolling by generating a random number from 1 to 6.
      4. To implement the functionality to repeatedly roll the dice, we will use a while loop so that the user can choose to roll the dice again.
         1. For this, we will ask the user if they want to roll the dice again.
         2. We will assign the user input to a variable roll\_again.
         3. If the user inputs "yes" or "y", we will roll the dice by executing the while loop again.
         4. Otherwise, we will come out of the while loop.
   2. Some examples:
      1. https://projects.raspberrypi.org/en/projects/hello-world-editor/3
      2. https://www.geeksforgeeks.org/dice-rolling-simulator-using-python-random/
      3. https://www.pythonforbeginners.com/code-snippets-source-code/game-rolling-the-dice
   3. Video examples:
      1. https://www.youtube.com/watch?v=x-Ag2\_bJ40Y
      2. https://www.youtube.com/watch?v=Mysh7onBVEA
      3. I also have a dice roller on my github: https://gist.github.com/lacey-conrad/f267144dafb188b308bbf1d693511a1d
         1. Your roller doesn’t need to be this elaborate. This example is rolling 4 dice three times and calculating the highest numbers in each set of rolls. There are also great examples of for loops and while loops in this code.
2. Read-in the covid dataset (.csv file) and perform exploratory data analysis on it.
   1. Strategy (you can follow the same strategy outlined in my example on GitHub):
      1. Open the csv file in excel. Rename columns/variables as you see fit (make sure the column variable titles are something easy to “call” in a program).
      2. Create a new Jupyter notebook in your favorite directory location.
      3. Import relevant libraries.
      4. Read-in the csv file.
      5. Print the shape of the dataset.
      6. Print out summary statistics of the dataset.
      7. Check for missing values and print out how many missing values there are in the dataset.
      8. Check for and drop any duplicates.
      9. Data visualization:
         1. Select a target/response variable of interest to you. The target/response variables are usually (but not always) the column titles.
         2. I suggest selecting the “confirmed” category, which gives us a count of how many confirmed cases of Covid occurred at the time of this data collection in each representative country.
            1. This is a good time to refresh your knowledge of what a dependent and independent variable are.
            2. Dependent variables have values that DEPEND on one (or many) independent variables.
            3. For some help: https://www.scribbr.com/methodology/independent-and-dependent-variables/
            4. In this dataset, the independent variable is country, and the dependent variables are the columns.
            5. Generally (but not always) dependent variables are plotted on the y-axis of a graph, and the independent variables are plotted on the x-axis of a graph.
         3. Use seaborn to create a bar chart of the number of covid cases in each country (it may not hurt to create a smaller subset of countries for ease of visualization and space saving reasons here).
         4. Create a pair plot using seaborn to investigate multiple dependent variables. I suggest selection 2-3 variables to run this analysis.
         5. Create histograms using seaborn of several dependent variables. Once again, select 2-3 to focus on.
   2. Examples:
      1. https://www.geeksforgeeks.org/exploratory-data-analysis-in-python/
      2. https://deepnote.com/app/code-along-tutorials/A-Beginners-Guide-to-Exploratory-Data-Analysis-with-Python-f536530d-7195-4f68-ab5b-5dca4a4c3579
      3. https://www.kaggle.com/code/imoore/intro-to-exploratory-data-analysis-eda-in-python
      4. Also see my example in GitHub located at: https://github.com/lacey-conrad/AIML-course/blob/master/MovieLens%20EDA%20Project.ipynb
3. Read-in a microscope slide image and perform simple image conversions using my “IHC image analysis intro”.
   1. Use the general framework in my “IHC image analysis intro” notebook.
   2. The general process is:
      1. Read-in the image.
      2. Display the image within your notebook using your favorite library (we used PIL previously).
      3. Determine the shape of the image.
      4. Convert the image to hematoxylin, eosin, and DAB “slices”.
      5. Display the “slices”.
      6. Save the “slices” to your local machine.
      7. Invert the DAB “slice”.
      8. Save the inverted DAB slice to your local machine.
      9. Perform Voronoi-Otsu labeling (this is the process that produces the “skittles” appearance of the images).
      10. Save this image to your local machine.