* Advanced Data Modeling
  + **basic meta prophet model**
    - Libraries used: tkinter, pandas, prophet, matplotlib
    - Project wherein a user selects a file, enters the holidays impacting the sales of their products (as well as upcoming sales), and receives an output directory containing a graph as well as a spreadsheet of the predictions. Allows for variable historical data.
  + **meta prophet model with two permutations**
    - Libraries used: tkinter, pandas, prophet, matplotlib
    - Same as “**basic meta prophet model,”** with the difference that it accommodates datasets with “division” and “product,” allowing more data to be ingested and predicted. Allows for variable historical data.
  + **Holt winters 1month historical**
    - Libraries used: tkinter, statsmodels, pandas
    - Basic code that uses Holt-Winters forecast to provide demand forecasting with only one month historical data.
  + **Holt winters two variable permutation**
    - Libraries used: Tkinter, pandas, statsmodels, matplotlib
    - Variation on **“meta prophet model with two permutations”** that uses Holt-Winters instead of Prophet.
* Database Interaction
  + **MongoDB import**
    - Libraries used: pandas, pymongo
    - Simple code block to input csv directory into MongoDB tables.
  + **MongoDB import 2**
    - Libraries used: pandas, pymongo
    - Simple code block to input csv directory into MongoDB tables. Slightly varied from above.
* Misc
  + **create a word cloud with inverse image mapping**
    - Libraries used: tkinter, wordcloud, matplotlib, numpy, PIL, colletions, matplotlib
    - Selects a spreadsheet with words, selects an image file (both using tkinter), and creates a wordcloud within the bounds of the selected image file.
  + **Debt payment calculator**
    - Libraries used: tkinter
    - Prompts user for financial information and calculates a recommended debt payment schedule
  + **Image to text**
    - Attempts to scrape an image for text. Doesn’t work.
* NLP
  + **basic sentiment polarity analysis**
    - Libraries used: textblob
    - Simple code block that analyzes multiple text files for sentiment and outputs a CSV with results.
  + **NLP text file analysis**
    - Libraries used: pandas, nltk, textblob, tkinter, wordcloud, matplotlib
    - Larger scale code block that selects a directory with .txt files, lists keywords to search for, runs NLP (keyword count/analysis/polarity) on the directory, and saves results to a CSV.
* Spreadsheet Manipulation
  + **Clean CSV**
    - Removes unnecessary characters from selected CSV.
  + **Compare xlsx spreadsheets 1**
    - Just compares two .xlsx files to find common words.
  + **Compare xlsx spreadsheets 2**
    - Just compares two .xlsx files to find common words.
    - The same as above. I don’t know why I have two.
  + **Convert JSON to CSV**
    - Just converts a JSON to a CSV format.
  + **Convert PDFs in directory all to text**
    - Libraries used: tkinter, spacy, PyPDF2
    - Selects directory and parses all PDFs, returning an output of the plaintext in said PDFs.
  + **Convert xlsx to csv**
    - Does what title says.
  + **Pad numbers in xlsx using openpyxl**
    - Uses openpyxl to pad numbers in selected file to 13 digits.
  + **Suggest payment from xlsx input**
    - Libraries used: tkinter, pandas
    - Using an excel template (that I have saved anywhere but my github), takes in financial information, and recommends a debt payment strategy.
  + **Very basic pandas csv manipulation**
    - Splits data in CSV.
  + **Xlsx workbook cleaning via openpyxl**
    - Using openpyxl, simply cleans some data in an xlsx workbook
* Webscraping
  + **browse Ebay NO API**
    - Searches Ebay for a list of words without an API.
  + **download data from API**
    - Downloads data from the USDA website.
  + **download reports from api**
    - Downloads reports from the USDA website.
  + **google maps API find distance w csv**
    - Reads a .csv of cities and calculates the distance between all listed cities using google maps API
  + **retrieve API data convert to JSON**
    - Libraries used: requests, json, pandas
    - Scrapes the listed URL and converts returned data to JSON format.
  + **retrieve data from API**
    - Another example of scraping data from an API.
  + **scrape multiple files for keywords**
    - Scrapes a web link for keywords.
  + **scrape URL for data 1**
    - Scrapes a URL for MTG card data
  + **scrape URL for data 2**
    - Same as above
  + **scrape website for ingredients**
    - I hate websites for recipes. I wrote python to scrape the ingredients. It kinda works.