PDFs and Spreadsheets

- Python has the ability to work with PDF files and spreadsheet files.
- In this section we will explore libraries that allow us to interact with these files.
- Note: We highly recommend you work in the same location as the lecture notebooks, since we will be referencing many files in that location.

Working with CSV Files

Complete Python Bootcamp

- CSV stands for comma separated variables and is a very common output for spreadsheet programs.
- Example:
    - Name, Hours, Rate
    - David,   20,   15
    - Claire,   40,   20

- Note, that while its possible to export excel files and Google Spreadsheets to .csv files, it **only** exports the information.
- Things like formulas, images, and macros can not be within a .csv file.
- Simply put, a .csv file only contains the raw data from the spreadsheet.

- We will work with the built-in csv module for Python, which will allow us to grab columns, rows, and values from a .csv file as well as write to a .csv file.
- Keep in mind, this is a very popular space for outside libraries, which you may want to explore.

- Other libraries to consider:
    - Pandas
        - Full data analysis library, can work with almost any tabular data type.
        - Runs visualizations and analysis.
        - One of my personal favorites, we teach it in various data science courses.

- Other libraries to consider:
  - Openpyxl
    - Designed specifically for Excel files.
    - Retains a lot of Excel specific functionality.
    - Supports Excel formulas.
    - python-excel.org tracks various other Excel based Python libraries.

- Other libraries to consider:
    - Google Sheets Python API
        - Direct Python interface for working with Google Spreadsheets.
        - Allows you to directly make changes to the spreadsheets hosted online.
        - More complex syntax, but available in many programming languages.

- The common factor between all of these spreadsheet programs is that they can always export to .csv.
- Let’s explore Python’s built-in capabilities with the csv module!

Working with PDF Files

- PDF stands for Portable Document Format and was developed by Adobe in the 1990s.
- The most important thing to keep in mind is that while PDFs share the same extension and can be viewed in PDF readers, many PDFs are **not** machine readable through Python.

- Since PDFs mainly encapsulate and display a fixed-layout flat document, there is no machine readable standard format, unlike CSV files.
- This means that a PDF that was simply scanned is highly unlikely to be readable.

- Additions to PDFs such as images, tables, format adjustments can also render a PDF unreadable by Python.
- There are many paid PDF programs that can read and extract from these files, but we will use the open-source and free **PyPDF2** library.

- We’ve made sure that the PDF files included in this course material are readable by PyPDF2.
- Unfortunately we can’t offer assistance for you own personal PDF files if they are not readable by PyPDF2.

- Let’s explore working with PDF files in Python.
- Remember you will first need to install PyPDF2 at your command line:
  - **pip install PyPDF2**

PDF and CSV Puzzle Exercise Solution