Overview

⚠ Before you start ⚠

Duplicate this Jupyter Notebook in your `week-06` folder (right-click -> Duplicate) and then add your last name to the beginning of it (ie. `blevins-hw-05.ipynb` - otherwise you risk having all your work overwritten when you try to sync your GitHub repository with your instructor's repository.

⚠ No, seriously: check the name of this file. Is it the copy you made and not the origina file? If so, you can proceed ⚠

Student Name: Kylie Miller

This will help you better learn how to use lists and loops, dictionaries, and functions in Python in order to work with textual data.

Getting to Know the Data

In this homework you're going to work with the diary of Martha Ballard, a midwife from Maine in the 1700s and early 1800s made famous by historian Laurel Ulrich's *A Midwife's Tale*. A project at George Mason University digitized her diary and put it online. I've done some research using the entries, and am supplying you with two years' worth of Ballard's diary entries (1804 and 1805). Each entry is contained in a separate text file that I've already processed and cleaned.

You can find all of her diary entries as several hundred text files in the `data` subfolder. Navigate to the `data` folder in Jupyter Lab and open up a few of the .txt files to get a feel for what sort of historical documents you will be working with and how they are structured.

1. What are some observations you have about these as **historical sources**? What jumps out at you?

These diary entries are very brief but also consistent. Therefore, while they do not necessarily include extensive details, they can provide some more routine information for daily life. For instance, from what I could tell the weather of every day was mentioned, which could give insight into weather patterns. (It is also interesting to think about why, in her extremely short entries, weather was important enough to be noted every day.) In general though, such short and not descriptive entries seem fit for data analysis,

especially because she seems to track delivering babies, people, etc. (However, spelling errors may make some parts harder to analyze.) While this diary could provide insight into the daily life of a midwife in main in this time period, it is also important to ask questions about Martha Ballard herself - to what extent was her level of literacy common for a midwife at this time? (She had many spelling mistakes but at the same time could write, etc.)

2. Look at the filenames of Ballard's diary entries (ex. `18040323.txt`). Try to figure out: what information is stored in the file's name and how is that information structured? What does the file name tell you about the diary entry that is NOT contained in the text file itself?

These file names show the date of the entry, strutured by year, month, day (0000/00/00). This can tell more about the time of year as well as specific dates. It also shows that these were daily entries as it appears that days were rarely skipped.

3. Find and open the diary entry for February 5, 1804. What major event happened to Ballard's family that day?

Ballard's son Ephrain got married, so there was a gathering of many family members.

(Clear. Son*s Pollard and Lambard, their wives and par#t$ of their children Came here. Rhoda, Hannah, Samuel & Dolly tarried here, their parents went to meeting. mr Black and Oldes#t$ Daughter Came with them after meeting and partook with me of a Turkey my husband Sent to me Since he went from home. Son Ephraim and Mary Farewel were Joind in wedlock this evening. at home. my children here,mr Black allso. Son Ephraim was Married to Mary Farewel, Oldes#t$Daug#t$ to y#e$ Widdow.)

Wrangling the Data

The goal of this section is to take your hundreds of text files worth of diary entries and add them into a dictionary. Each entry in the dictionary is going to consist of a **key** that corresponds to the name of the file (diary entry) and a **value** that contains the contents of the file (the written text of the entry).

We will be implementing the following steps across several questions:

- Look inside data folder and have Python generate a list of filenames of all the files inside that folder
- Loop through our list of filenames, open each diary entry, and read its contents
- Decide whether each diary entry was written in 1804 or 1805 and put the entry into a corresponding list

First we're going to import the `pathlib` library, which helps us more easily work with folder and files. Run this code:

In [18]:
```
from pathlib import Path
```

I've provided some code below that will allow you to create two new lists: `file_names` and `file_paths`. The list `file_names` contains a list of all the names of the files ending in `.txt` in our `data` folder (ie. `18040101.txt`). The list `file_paths` is a string with the "path" to that file within the `data` folder (ie. `data/18040101.txt`). Run the following code cell:

In [20]:
```
txt_files = list(Path('data').glob('*.txt'))

file_paths = []

# Display the files
for file in txt_files:
    file_paths.append(str(file))
```

4. Add code to loop through the first **10 items** in your list of **file paths** and print out each of those ten file paths in order to make sure you've done this correctly.

In [22]:
```
for item in file_paths[:10]:
    print(item)
```

```
data/18040608.txt
data/18050427.txt
data/18040620.txt
data/18040813.txt
data/18040807.txt
data/18050619.txt
data/18050625.txt
data/18040422.txt
data/18050802.txt
data/18050816.txt
```

We're eventually going to open all of the files in your directory, but with the principal "start small" let's start by just opening and reading just **one** of the diary entry files from January 1, 1804. Run the code cell below:

In [24]:
```
diary_text=open('data/18040101.txt', encoding='utf-8').read()
print(diary_text)
```

```
Cloudy, Snowd at night. mr Ballard and Ephraim to meeting. I have been unwel
l. Son Jonathan, his wife and 6 children Sup#t$ here. we had a puding and ro
ast Spare rib. I was very unwell all nigh#t$ but, as is usual, did with out
much Care taken of me. Rachel to bed at 8 oClock. at home, very unwell.
```

5. Open, read, and print out the contents of the **February 5, 1804** diary entry.

In [26]:
```
diary_text=open('data/18040205.txt', encoding='utf-8').read()
print(diary_text)
```

```
Clear. Son*s Pollard and Lambard, their wives and par#t$ of their children C
ame here. Rhoda, Hannah, Samuel & Dolly tarried here, their parents went to
meeting. mr Black and Oldes#t$ Daughter Came with them after meeting and par
took with me of a Turkey my husband Sent to me Since he went from home. Son
Ephraim and Mary Farewel were Joind in wedlock this evening. at home. my chi
ldren here,mr Black allso. Son Ephraim was Married to Mary Farewel, Oldes#t
$Daug#t$ to y#e$ Widdow.
```

6. Let's try to isolate JUST the filename rather than the full path - ie. we want to go from `data/18040101.txt` to `18040101.txt`. Write a new function called `isolate_filename` that does the following:

- Use the `split()` function to separate the string of the full path into a list with two strings: `data` and `18040101.txt`. Hint: you can specify a specific letter or character to "split" it on.
- Returns the second item in that two-item list (ie. `18040101.txt`)

In [28]:
```
def isolate_filename(filename):
    filename_divided = filename.split('/')
    return(filename_divided[1])

filenames = []

for name in file_paths:
    filenames.append(isolate_filename(name))

print(filenames[:10])
```

```
['18040608.txt', '18050427.txt', '18040620.txt', '18040813.txt', '18040807.t
xt', '18050619.txt', '18050625.txt', '18040422.txt', '18050802.txt', '180508
16.txt']
```

7. Let's stitch together all of our the above steps and apply them to every diary entry in the folder.

- Create an **empty dictionary** named `diary_dictionary`
- Set up a `for loop` to go through your `file_paths` list of file names (ex. `data/18040101.txt`, `data/18040102.txt`, etc.) that you generated above.
- **Inside** your `for loop` you are going to do the following:
  - Assign a new variable called `filename` that gets filled with the returned value from sending the full file path to your function `isolate_filename`
  - Assign a new variable called `diary_text` and assign it the contents of the file using your new variable.
  - Add a new item to your dictionary, with the `filename` as the **key** (ex. `18040101.txt`) and the contents of the file (`diary_text`) as the **value**.
- Print out **the number of entries** that are now in your dictionary

In [30]:
```
diary_text=open('data/18040608.txt', encoding='utf-8').read()
print(diary_text)
```

```
Clear. I have been at home, helped wash. Joanna Smith worked for me this da
y. mrs Dunfee & Daughter Ballard took Tea here. at home.
```

In [31]:
```
diary_dictionary = {}

for name in file_paths:
    filename = isolate_filename(name)
    diary_text = open(name, encoding='utf-8').read()
    diary_dictionary[filename] = diary_text

len(diary_dictionary)
```

Out[31]: 731

8. Complete the following with `diary_dictionary` of entries:

- Use the **key** to access and print the contents for Ballard's entry for **February 5, 1804**.
- Create a new `list` of **words** in the above entry (hint: String Methods)
- Print the number of **words** in the above entry.

In [33]:
```
print(diary_dictionary["18050205.txt"])
```

```
Clear. I have began a Stockin for my Daughter Lambard and done other matter
s. we are informd that the wife of George Hodgkins Departed this life this m
orn abou#t$ 5 O Clock after an illness of 4 years. at home. Death mrs Hodgsk
ens, Varsalboro.
```

In [34]:
```
words_list = diary_dictionary["18050205.txt"].split()
print(words_list)
```

```
['Clear.', 'I', 'have', 'began', 'a', 'Stockin', 'for', 'my', 'Daughter', 'L
ambard', 'and', 'done', 'other', 'matters.', 'we', 'are', 'informd', 'that',
'the', 'wife', 'of', 'George', 'Hodgkins', 'Departed', 'this', 'life', 'thi
s', 'morn', 'abou#t$', '5', 'O', 'Clock', 'after', 'an', 'illness', 'of',
'4', 'years.', 'at', 'home.', 'Death', 'mrs', 'Hodgskens,', 'Varsalboro.']
```

In [35]:
```
print(f"Ballard wrote {len(words_list)} words on February 5, 1805.")
```

```
Ballard wrote 44 words on February 5, 1805.
```

Bonus Question 1:

Let's say we want to do the same thing as Question 8 (finding the length of an entry) but we don't want to write the same code over and over. Review Walsh's Functions chapter. Define a new function that calculates and prints the length of any given diary entry measured by **number of words.** After you've defined the function, "call" it for the entry written on September 22, 1805.

In [37]:
```
def entry_numwords(entry):
    words_list = diary_dictionary[entry].split()
    print(len(words_list))

entry_numwords("18050922.txt")
```

```
106
```

Bonus Question 2:

- How long is the longest entry Ballard wrote in these years measured by the number of words?
- Which entry was it?
- Print the contents of that entry

Functions you might use:

- len()
- max()
- dictionary.values()

Plan:

- run a for loop through the dictionary to calculate number of words
- add this number of words to a list
- calculate the max number in that list
- run through num words again - if the number of words is the max number, print the contents of thatday

In [40]:
```
def entry_numwords(entry):
    words_list = diary_dictionary[entry].split()
    return(len(words_list))

entry_lengths = []

for entry in diary_dictionary:
    entry_lengths.append(entry_numwords(entry))

print(max(entry_lengths))
```

```
229
```

In [41]:
```
for entry in diary_dictionary:
    if entry_numwords(entry) == max(entry_lengths):
        print(f"The longest entry is {entry} with {max(entry_lengths)} words
        name = "data/"+entry
        diary_text = open(name, encoding='utf-8').read()
        print(diary_text)
```

```
The longest entry is 18040317.txt with 229 words.
Clear Part of the day. Son & Daughter Pollard and part of thier children her
e. Shee went on to See her Father. I went afternoon Conducted by Lemuel With
am, a lad who is Com to work with Son Ephraim. I returnd at evening very unw
ell. Lemuel went to take Son Lambard*s hors and Sleigh to Jones*s for Jonath
an and him to Come up in but not finding them ready to Come returnd with it.
they Came up on foot and Jonathan Came here with ou#t$ his hat. took him fro
m his Supper, pusht him out adors, Drove him home to his house Damning and p
ushing him down and Struck him. Shaw and Burr went on after to prevent his b
eing deprived of life. I followed on falling as I went till meeting Daughter
Lambard was assisted by her. I reacht his house, find him Cursing and Sweari
ng he would go and giv him a Hard whiping. my Daughter Lambard desired Haman
to go and Conducted him to Ephraim. he went and tarried all night. Son Lamba
rd brot me home in his Sleigh. O that the God of all Mercy would forgiv him
this and all other misconduc#t$. at Shubael Pitts*s, feel very unwel and wha
t a Sceanhad I to go at evening. may a Good God Support me. funeral mr James
Hinkley of Hollowell, his Death was very Suden.
```

I will be curious how using the "dictionary.values()" function would help with this problem, as I did not use it.

Below I am attempting to translate the name of the entry into a readable date

In [44]:
```
entry = "18040515.txt"

def entry_date(entry):
    if entry[4:6] == "01":
        month = "January"
    elif entry[4:6] == "02":
        month = "February"
    elif entry[4:6] == "03":
        month = "March"
    elif entry[4:6] == "04":
        month = "April"
    elif entry[4:6] == "05":
        month = "May"
    elif entry[4:6] == "06":
        month = "June"
    elif entry[4:6] == "07":
        month = "July"
    elif entry[4:6] == "08":
        month = "August"
    elif entry[4:6] == "09":
        month = "September"
    elif entry[4:6] == "10":
        month = "October"
    elif entry[4:6] == "11":
        month = "November"
    elif entry[4:6] == "12":
        month = "December"
    year = entry[:4]
    day = entry[6:8]
    return month + " " + day + " " + year

print(entry_date(entry))
```

```
May 15 1804
```

Now I am adjusting my previous answer to show the date of the entry rather than name of entry

In [46]:
```
for entry in diary_dictionary:
    if entry_numwords(entry) == max(entry_lengths):
        print(f"The longest entry is {entry_date(entry)} with {max(entry_ler
        name = "data/"+entry
        diary_text = open(name, encoding='utf-8').read()
        print(diary_text)
```

```
The longest entry is March 17 1804 with 229 words.
Clear Part of the day. Son & Daughter Pollard and part of thier children her
e. Shee went on to See her Father. I went afternoon Conducted by Lemuel With
am, a lad who is Com to work with Son Ephraim. I returnd at evening very unw
ell. Lemuel went to take Son Lambard*s hors and Sleigh to Jones*s for Jonath
an and him to Come up in but not finding them ready to Come returnd with it.
they Came up on foot and Jonathan Came here with ou#t$ his hat. took him fro
m his Supper, pusht him out adors, Drove him home to his house Damning and p
ushing him down and Struck him. Shaw and Burr went on after to prevent his b
eing deprived of life. I followed on falling as I went till meeting Daughter
Lambard was assisted by her. I reacht his house, find him Cursing and Sweari
ng he would go and giv him a Hard whiping. my Daughter Lambard desired Haman
to go and Conducted him to Ephraim. he went and tarried all night. Son Lamba
rd brot me home in his Sleigh. O that the God of all Mercy would forgiv him
this and all other misconduc#t$. at Shubael Pitts*s, feel very unwel and wha
t a Sceanhad I to go at evening. may a Good God Support me. funeral mr James
Hinkley of Hollowell, his Death was very Suden.
```

I was making fun of my long code and showed it to my husband and he gave me the idea of doing that with dictionaries. So I feel like maybe it's a good way to get more dictionary practice because I'm still not super comfortable with them so that is what is below :P

In [48]:
```
month_translation = {"01": "January",
                     "02": "February",
                     "03": "March",
                     "04": "April",
                     "05": "May",
                     "06": "June",
                     "07": "July",
                     "08": "August",
                     "09": "September",
                     "10": "October",
                     "11": "November",
                     "12": "December"}

entry = "18040515.txt"

def entry_date_2(entry):
    month = month_translation[entry[4:6]]
    year = entry[:4]
    day = entry[6:8]
    return month + " " + day + " " + year
```

```
print(entry_date_2(entry))
```

```
May 15 1804
```

Submission

Follow the instructions to submit the assignment on Canvas in two files (one `.ipynb` and one `.pdf`).

1. Save your notebook
2. Go to Kernel -> Restart Kernel and Run All Cells
3. Export as PDF or HTML