Tkinter

- design limitations
    - positioning of widgets (pack and grid)
    - positioning of charts

- widgets limitations
    - buttons
    - entry button (have to manually click the Enter button)

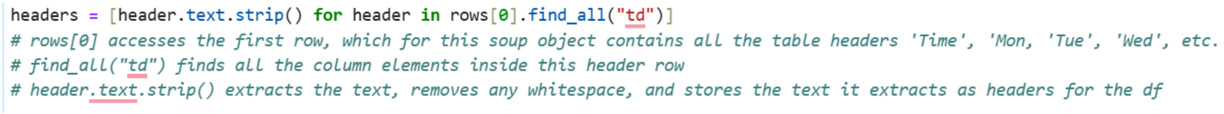
- readability of code
    - multiple screens require nested defs
    - doable without it, but features and aesthetics are very limited

- can sometimes crash because of the constant refreshing of the page

Syllabus Processing

- Difficulties
    - Creating the Sentiment Analysis for finding keywords
    - Optimizing the Keyword Finder to cut down processing time (i was malding and balding - dux)
    - Trying to make a super accurate code that finds all dates and gets each context.
        - Good thing Raiz suggested to do it the other way around
- Limitations
    - Assumes a few formats for it to work
        - Assumes that Dates come after a specific keyword and can be found a few words away from it
            - If syllabus is formatted where date comes before keyword, then it fails to accurately position/find events
            - If date is placed more than 15 words away from keyword, it will not be found
    - Gets 1 word before keyword, which usually results in the name of the event looking weird, sometimes even gets "=" which messes with the gsheet
    - Despite the optimizations made, it still takes roughly 8 to 20+ seconds to process a syllabus depending on the length, unable to find a way to make processing instantaneous because of the combined use of sentiment (wordnet) and vector similarity
    - Does not work with syllabuses in Tagalog, only for syllabuses written in English (Tagalog unsupported in spaCy - sentiment analysis based on WordNet)

Soup (Aisis) to Panda DataFrame



```
headers = [header.text.strip() for header in rows[0].find_all("td")]
# rows[0] accesses the first row, which for this soup object contains all the table headers 'Time', 'Mon, 'Tue', 'Wed', etc.
# find_all("td") finds all the column elements inside this header row
# header.text.strip() extracts the text, removes any whitespace, and stores the text it extracts as headers for the df
```

- Relies on AISIS being formatted in this exact way where the first row of the schedule page contains the headers of ‘Time’ and the names of the days
- Needs to be updated if AISIS’s formatting is updated

Event Creation (on GCal)

- Difficulties
    - Initial research on finding ways to connect to Google Apps
    - Took around 30 minutes to find the video that helped set up the connection between GCal and GSheets and python
- Limitations
    - Assumes correct formatting on GSheets for dates (start date and end date for classes, date for events)
    - Can sometimes fail due to failed token authentication, which requires the user to go into the folder and delete the token.json file.
    -

“ClassUp”

- assumes that gcal is already signed in (gcal blocks selenium log in)
- pyautogui takes a screenshot of the entire screen and crops it
- image quality is unreliable
- background picture must be downloaded and named ‘background_image.jpg’

OVERALL:

- Might be difficult to reproduce on other laptops because it requires Google Cloud setup for the APIs to be able to connect to your account’s GSheets and GCal (for safety reasons)
    - However, after being set-up for the first time, it doesn’t need to be done again.