SI 206 Discussion 9

Midterm Review

Midterm Review

- Reading in files (txt, csv)
- Regex
- BeautifulSoup

Reading Files

- TXT files
  - file_obj = open(<filepath>, ‘r’) as f
  - file_obj.read() #reads entire file as string
  - file_obj.readlines() #reads entire file as list of strings
  - file_obj.close()
- CSV files
  - reader = csv.reader(f)
  - Iterate through reader to read lines of csv as lists
  - writer = csv.writer(f, delimiter=’,’, quotechar=’”’, quoting=csv.QUOTE_MINIMAL)
  - writer.writerow(<list>) #write list to row of csv file
- with statement
  - Closes file automatically

Regex

- re.findall(‘<string>’) #returns a list of strings that match the regex
- When using the \b character, make sure your string is a raw string
- Special characters:
    - \w - Alphanumeric characters and underscore
    - [] - set of characters
    - \s - Any whitespace character
    - . - Any character
    - * - Repeat 0 or more times
    - + - Repeat 1 or more times
    - \b - Boundary between alphanumeric characters and whitespace
    - ^ - start of a string
    - $ - End of a string
    - Consult regex cheat sheet for more special characters

BeautifulSoup

- 3 steps
  - Create variables that stores url of website
  - Get the data from the url
    - r = requests.get(url)
  - Create a BeautifulSoup object using the data
    - soup = BeautifulSoup(r.content, ‘html.parser’)
- Soup object methods
  - soup.find(‘<tag>’, <attribute>=’<value>’) #returns the first tag that matches
  - soup.find_all(‘<tag>’, <attribute>=’<value>’) #returns a list of all tags that match
  - tag.attrs #returns a dictionary of the attributes in the tag object
  - tag.get(‘<attribute>’) #returns the value of a specified attribute

Tasks

- Task 1
  - Implement the get_profs() function. This function should read in `umsi_faculty.csv` and parse it to return a list of lists. Each list should contain the name, title(s), and email address of each professor in the csv file.
- Task 2
  - Implement the get_valid_emails() function. This function should accept the list from Task 1 and return a dictionary. The keys should be the names of professors and the values should be their email addresses. Some of the email addresses were entered erroneously. Use a regular expression to filter out invalid email addresses.
  - A valid email address should:
    - Only have lowercase letters
    - End with @umich.edu

Tasks

- Task 3
  - Implement the function get_lsa_majors(). This function should use BeautifulSoup to scrape the UMich admissions website (the link can be found in the docstring) and return a list of all of the majors offered through the College of Literature, Science, and the Arts.