In [ ]:

```
from bs4 import BeautifulSoup
from requests import get
import re
import pandas as pd
import csv


with open('UCI files.csv', mode = 'w', newline = '') as csvfile:
    writer = csv.DictWriter(csvfile, fieldnames = ['num_people_enrolled', 'total_class_size', 'class_number', 'term','professor'])
    writer.writeheader()

    url = 'http://www.reg.uci.edu/soc/archives_quarterly.html'
    response = BeautifulSoup(get(url).text, 'lxml')
    table = response.find_all('tr')
    seen_term = set()
    seen_term.add('UCI Home')
    for each_term in table:
        term = each_term.find('a')
        if term is not None and term.text not in seen_term:
            seen_term.add(term.text)
            term_year = term.text
            term_link = term.get('href')
            term_page = 'http://www.reg.uci.edu/soc/' + term_link
            #print(term_page)
            term_page_parser = BeautifulSoup(get(term_page).text, 'lxml')
            term_table = term_page_parser.find_all('tr')
            for each_subject in term_table:
                each_subject_link = each_subject.find('a')
                subject_split = None
                if each_subject_link is not None:
                    subject_split = each_subject_link.text.split()
                #print(subject_split)
                if each_subject_link is not None and subject_split != list() and (subject_split[0] == 'Computer' or subject_split[0] == 'CSE'):
                    cs_year = term_page+'/' + each_subject_link.get('href')
                    cs_parser = BeautifulSoup(get(cs_year).text, 'lxml')
                    class_info = cs_parser.find_all('tr', {'valign' : 'top'})
                    for each_class in class_info:
                        enrollment_info = each_class.find_all('td', {'align': 'right'})
                        class_type = each_class.find_all('td')
                        if enrollment_info != list() and len(class_type) > 1 and class_type[1].text == 'Lec':
                            professor_block = each_class.find_all('td', {'bgcolor':'#D5E5FF'})
                            professor = None
                            class_num = None
                            if professor_block != list():
                                professor = professor_block[2].text
                                class_num = professor_block[0].text
                                #print(class_num)
                            #print(professor_block)
                            capacity_enrolled = enrollment_info[1].text
                            split = capacity_enrolled.split('/')
                            if len(split) == 2:
```

```
                                class_info = {
                                    'num_people_enrolled':split[0],
                                    'total_class_size':split[1],
                                    'term': term_year,
                                    'class_number': class_num,
                                    'professor': professor
                                }
                                writer.writerow(class_info)
                            else:
                                class_info = {
                                    'num_people_enrolled':enrollment_info[1].t
ext,
                                    'total_class_size':enrollment_info[0].text
,
                                    'term':term_year,
                                    'class_number': class_num,
                                    'professor': professor
                                }
                                writer.writerow(class_info)
```