get_fb_events

April 19, 2018

```
In [ ]: import os
        import sys
        import time
        import json
        import dateparser
        import pandas as pd
        import numpy as np
        from bs4 import BeautifulSoup
        from lxml import html

        from selenium import webdriver
        from selenium.webdriver.common.by import By
        from selenium.webdriver.support.ui import WebDriverWait
        import selenium.webdriver.support.expected_conditions as EC
        from selenium.common.exceptions import TimeoutException
        from selenium.webdriver.common.keys import Keys
        from selenium.webdriver.support.ui import Select

        import warnings
        warnings.simplefilter('ignore', FutureWarning)
```

```
In [ ]: #csv to dataframe
        df = pd.read_csv('/Users/celarno/Downloads/cat.csv')
```

```
In [ ]: def _remove_attrs(soup):
            for tag in soup.findAll(True):
                tag.attrs = None
            return soup
```

```
In [ ]: final = {
            'name' : [],
            'events' : []
            }
```

```
In [ ]: i = 0
        for row in df.itertuples(index=True, name='Pandas'):
            i = i+1
            venue = getattr(row, "name")
```

```
url = getattr(row, "facebook") + "events/"
print("{} --- {}".format(i, venue))

driver = webdriver.Chrome()
driver.get('https://www.facebook.com/')
print("Opened facebook...")
a = driver.find_element_by_id('email')
a.send_keys('')
b = driver.find_element_by_id('pass')
b.send_keys('')
c = driver.find_element_by_id('loginbutton')
c.click()
print("logged in...")

driver.implicitly_wait(10)
driver.get(url)
time.sleep(5)
try:
    some_object = WebDriverWait(driver, 30).until(
        EC.presence_of_element_located((By.ID,'pagelet_events')))
except:
    print("couldnt find events")
    continue
finally:
    try:
        r = driver.page_source
        driver.quit()
    except:
        print("couldnt find events")
        r = None

if r is None:
    continue

soup = BeautifulSoup(r, "lxml")
fb_events = soup.find("div", {"id": "pagelet_events"})

clean_soup = _remove_attrs(fb_events)
for match in clean_soup.findAll('span'):
    match.unwrap()
for match in clean_soup.findAll('div'):
    match.unwrap()
for match in clean_soup.findAll('a'):
    match.unwrap()
for match in clean_soup.findAll('table'):
    match.unwrap()

fb_events = clean_soup
```

```
            rows = fb_events.find_all('tr')
            data = {
                'date' : [],
                'title' : [],
                'location' : []
                }

            for row in rows:
                cols = row.find_all('td')
                data['date'].append(cols[0].get_text())
                data['title'].append(cols[1].get_text())
                data['location'].append(cols[2].get_text())

            events = pd.DataFrame(data)
            events['location'] = events.location.apply(lambda x: x[:-8])
            events.date = events.date.str.extract('(\d+)') + " " + events.date.str[0:3] + " 20
            events.date.str.strip()
            events.location.str.strip()
            events.title.str.strip()

            final['name'].append(venue)
            final['events'].append(events)

In [ ]: export = pd.DataFrame(final)
        out = export.to_json(orient='records')
        with open('fb_events_pre.json', 'w') as f:
            f.write(out)

In [ ]: test = json.load(open('fb_events_pre.json'))

In [ ]: print(json.dumps(test, indent=4, sort_keys=True))

In [ ]: for t in test:
            for event in t["events"]:
                #event['date'] = "14 MAR 2018"
                new_date = dateparser.parse(str(event['date']))
                new_date = new_date.strftime("%Y-%m-%d")
                event["date"] = str(new_date)
                new_title = event["title"].split("\u00b7")[0].strip(" ")
                event["title"] = new_title
                event["location"] = str(event["location"])

In [ ]: json.dumps(test[0]["events"])

In [ ]: with open('fb_events.json', 'w') as f:
            f.write(json.dumps(test))
```