webscraping-realtor-oregon

April 28, 2024

[1]:
```
from selenium import webdriver
from bs4 import BeautifulSoup
import pandas as pd
import undetected_chromedriver as uc
import time
```

[14]:
```
def scrape_data(url):
    prices = []
    bedrooms = []
    bathrooms = []
    sqfts = []
    addresses = []
    zip_codes = []
    options = uc.ChromeOptions()
    options.add_argument("--enable-javascript")
    options.add_argument("start-maximized")
    driver = uc.Chrome(options=options, browser_executable_path="/usr/bin/
 ↪chromium", driver_executable_path="/tmp/chromedriver")
    driver.get(url)
    time.sleep(1)

    driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
    content = driver.page_source
    soup = BeautifulSoup(content, features='html.parser')
    property_cards = soup.select('div[class^="CardContent__StyledCardContent"]')

    for card in property_cards:
        price = None
        bedroom_raw = None
        bathroom_raw = None
        sqft_raw = None

        price_elem = card.find('div', class_='card-price')
        if price_elem:
            price_text = price_elem.text.strip()
            price = ''.join(filter(str.isdigit, price_text))
            price = price.lstrip('$')
```

```
        bedroom_elem = card.find('li',␣
↪class_='PropertyBedMetastyles__StyledPropertyBedMeta-rui__a4nnof-0')
        if bedroom_elem:
            bedroom_raw = bedroom_elem.text.strip()

        bathroom_elem = card.find('li',␣
↪class_='PropertyBathMetastyles__StyledPropertyBathMeta-rui__sc-67m6bo-0')
        if bathroom_elem:
            bathroom_raw = bathroom_elem.text.strip()

        sqft_elem = card.find('li',␣
↪class_='PropertySqftMetastyles__StyledPropertySqftMeta-rui__sc-1gdau7i-0')
        if sqft_elem:
            sqft_raw = sqft_elem.text.strip()

        bedroom = ''.join(filter(str.isdigit, bedroom_raw)) if bedroom_raw else␣
↪None
        bathroom = ''.join(filter(str.isdigit, bathroom_raw)) if bathroom_raw␣
↪else None
        sqft = ''.join(filter(str.isdigit, sqft_raw.split()[0]))[:4] if␣
↪sqft_raw else None

        address_elem = card.find('div', class_='card-address')
        address = address_elem.text.strip() if address_elem else None
        zip_code = address.split()[-1] if address else None

        prices.append(price)
        bedrooms.append(bedroom)
        bathrooms.append(bathroom)
        sqfts.append(sqft)
        addresses.append(address)
        zip_codes.append(zip_code)

    return prices, bedrooms, bathrooms, sqfts, addresses, zip_codes
```

[18]:
```
all_data = []
for page_num in range(1, 21):
    url = f'https://www.realtor.com/realestateandhomes-search/Portland_OR/
↪show-newest-listings/pg-{page_num}'
    prices, bedrooms, bathrooms, sqfts, addresses, zip_codes = scrape_data(url)
    page_data = {
        'Price': prices,
        'Bedrooms': bedrooms,
        'Bathrooms': bathrooms,
        'Sqft': sqfts,
```

```
        'Address': addresses,
        'ZIP Code': zip_codes
    }
    all_data.append(page_data)

df = pd.concat([pd.DataFrame(page_data) for page_data in all_data],␣
 ↪ignore_index=True)

df.to_csv('scraped_data_oregon.csv', index=False)
```