Práctica 1

Web Scraping

Enlace Github: https://github.com/miguel-rodrigo/Tipologia_ciclo_vida_datos_pract1

1. Contexto

En el panorama actual, donde alquileres se disparan cada vez más hasta límites absurdos, donde una cuota de hipoteca sale más barata, sería interesante disponer de la información que nos permitiera detectar ofertas a precios anormalmente bajos, qué zonas elegir, estimar cuál será la dirección del mercado en un futuro cercano o decidir entre alquilar o comprar.

2. Título

Inmuebles en venta y alquiler en la provincia de Barcelona.

3. Descripción

El dataset ha sido recopilado de la web idealista.com. Se ha observado que los conjuntos de ofertas en diferentes webs no son equivalentes, y existen ofertas que sólo están presentes en algunas de ellas. Por esto, un trabajo futuro debería extraer las ofertas de otras webs y comparar para eliminar repetidos o contrastar información.

Para evitar los CATCHAPs, ha sido necesario copiar la cabecera creada por Google Chrome en la navegación normal. Por posibles cuestiones de seguridad y privacidad, se han eliminado de la cabecera las cookies. Si esto afectara al funcionamiento del script, se recomienda copiar la cabecera creada por el navegador empleado habitualmente. Típicamente, en “opciones de desarrollo” o bajo algún título similar, pueden explorarse las peticiones HTTP que se envían y se reciben.

Además, también ha sido necesario añadir un temporizador que ralentice la exploración, para no ser disruptivos con los sistemas de idealista.com, además de no ser bloqueados por ello por sus sistemas anti-robots.

4. Representación gráfica

5. Contenido

En el dataset se recogen ofertas de la web idealista.com recopiladas de toda la provincia de Barcelona. El nombre del anuncio recoge la ubicación además del tipo de inmueble. Además de esto, también se dispone de su área, el número de habitaciones y, en ocasiones, cierta información adicional como en qué planta se encuentra, si dispone o no de ascensor y si da o no al exterior.

6. Agradecimiento

Agradecimientos a idealista.com por recopilar y mantener la información, además de ofrecerla en un formato tan accesible.

7. Inspiración

Este trabajo se inspira

8. Licencia

La licencia de este contenido es desconocida. Sin embargo, al pertenecer a una entidad privada, es probable que su uso comercial esté prohibido. Se recomienda usar con precaución y contactar con idealista.com en caso de duda.

9. Código

Se incluye a continuación el código por completitud. Las cabeceras, incluyendo las cookies, han sido copiados de la petición generada por Chrome al navegar la web. Esto permite parcialmente sobrepasar el bloqueo de bots.

Es difícil obtener código legible en MS Word, por lo que se recomienda consultar el script en GitHub.

```
import requests
from bs4 import BeautifulSoup
import csv
import time
import random


header = {
    'accept':
'text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,imag
e/apng,*/*;q=0.8,application/signed-exchange;v=b3',
    'accept-encoding': 'gzip, deflate, br',
    'accept-language': 'es-ES,es;q=0.9,en;q=0.8',
    'dnt': '1',
    'upgrade-insecure-requests': '1',
    'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.103
Safari/537.36'
}

root = 'https://www.idealista.com'
url = '/alquiler-viviendas/barcelona-provincia/'

refresh_cycle = 0
while url is not None:
    page = requests.get(root + url, headers=header)
    print(page.status_code)

    if page.status_code != 200:
        with open('result.html', 'w') as result_page:
            result_page.write(page.text)

    soup = BeautifulSoup(page.content, features='html.parser')
    ads = [el for el in soup.find_all('article') if not
el.has_attr('class')]

    with open('data.csv', 'a+', newline='', encoding='utf-8') as
data_file:
        csv_writer = csv.writer(data_file, delimiter=',')
        for ad in ads:
            name = ad.find_all(attrs={'class': 'item-link'})[0].string
            price = ad.find_all(attrs={'class': 'item-
price'})[0].contents[0]
            currency = ad.find_all(attrs={'class': 'item-
price'})[0].contents[1].string
```

```
            #  Property details
            details = ad.find_all(attrs={'class': 'item-detail'})
            if details is not None:
                n_rooms = details[0].contents[0]
            else:
                n_rooms = None

            if len(details) > 1:
                m_sq = details[1].contents[0]
            else:
                m_sq = None

            if len(details) > 2:
                house_type = details[2].text
            else:
                house_type = None

            csv_writer.writerow([name, price, currency, n_rooms, m_sq,
house_type])

    next_button = soup.find('a', attrs={'class': 'icon-arrow-right-
after'})
    if next_button is not None:
        url = next_button.get('href')
    else:
        url = None

    if refresh_cycle < 15:
        t = random.random()
        time.sleep(6 + 8*t)
        refresh_cycle += 1
    else:
        t = random.random()
        time.sleep(60 + 30 * t)
        refresh_cycle = 0
```

10. Dataset

Debido a la extensión del archivo, no se incluirá aquí. Pero se incluye una pequeña muestra. El dataset completo puede consultarse en GitHub.

"Piso en calle de Berlín, 76, Sants, Barcelona",1.300,€/mes,3 ,73 ,3ª planta interior con ascensor

"Piso en josep gales, Llevant, Igualada",670,€/mes,2 ,92 ,3ª planta exterior sin ascensor

"Piso en del mar, 62, La Barceloneta, Barcelona",825,€/mes,1 ,50 ,5ª planta exterior sin ascensor

"Piso en calle de Roman Macaya, 23, Sant Gervasi - La Bonanova, Barcelona",1.150,€/mes,3 ,80 , exterior con ascensor

"Dúplex en paseo Garcia Faria, Diagonal Mar i el Front Marítim del Poblenou, Barcelona",3.000,€/mes,4 ,201 ,12ª planta exterior con ascensor

"Piso en MUNTANER, 7, Centre, Sitges",1.200,€/mes,3 ,82 ,2ª planta exterior sin ascensor

"Ático en EIX MACIA, La Creu Alta, Sabadell",1.200,€/mes,1 ,100 ,10ª planta exterior con ascensor

Los campos son:

- El nombre del inmueble, donde se muestra si es un piso, un chalé…Además de su ubicación exacta o aproximada.
- El precio del alquiler.
- El tipo de moneda y periodicidad.
- El número de habitaciones.
- Los metros cuadrados.
- Algunos detalles adicionales que no siempre aparecen, como la planta en la que se encuentra, si da al exterior o si dispone o no de ascensor.

En la web no son accesibles los datos geográficos de los inmuebles, por tanto, un primer paso para un análisis sería extraer las direcciones y pasarlas a una API como Google Maps para obtener su ubicación

Seguidamente, se extrae la información adicional que aparece codificada como cadenas de texto, como si es un piso, un chalé, un ático…Además de otros datos como si dispone o no de ascensor.

No es el caso de nuestros datos, pero es posible que el coste de alquiler apareciera por semana, o en diferentes monedas. En ese caso, también habría que realizar los cambios pertinentes.

Contribuciones	Firmal
Investigación Previa	Miguel A. Rodrigo Lisbona
Redacción de las Respuestas	Miguel A. Rodrigo Lisbona
Desarrollo del código	Miguel A. Rodrigo Lisbona