# Práctica 1 : Web Scraping

Eréndira Teresa Navarro García

**Contexto.**

*segundamano.mx* es una página web que tiene su antecedente como un periódico de anuncios clasificados y que ha logrado adaptarse al entorno, actualmente es una de las principales páginas de compra-venta de todo tipo de artículos y servicios de segunda mano en internet en México. El interés de obtener información de la página está en el entorno de hacer uso de web scraping, el tipo de artículo (renta de departamento) lo elegí por interés personal, al igual que las ubicaciones, ya que es mi objetivo poder establecer un análisis de las mejores opciones de vivienda ofertadas en la página.

El repositorio se encuentra en : https://github.com/ernavaga/apartmentprices-webscraping (https://github.com/ernavaga/apartmentprices-webscraping)

**Dataset: Web scraping en *segundamano.mx* de renta de departamentos en CDMX.**

**Descripción:**

Contiene las características de departamentos en renta dentro de 3 alcaldías de la CDMX con datos extraídos por medio de web scraping de la página *segundamano.mx*

**Datos:**

In [2]:
```
import pandas as pd
data = pd.read_csv("rentcdmx_segundamano.csv")
data.head()
```

Out[2]:

| | name | desc | price | currency | date | locality | suburb | |
|---|---|---|---|---|---|---|---|---|
| 0 | Se renta oficina ubicadisima | ¡¡¡EN RENTA OFICINA EN LA COLONIA ROMA NORTE!!... | 11000 | MXN | 2019-11-10 20:46:43 | Cuauhtémoc | Roma Norte | https:/ |
| 1 | Se renta departamento amueblado en edificio nu... | Se renta departamento amueblado en edificio nu... | 12000 | MXN | 2019-11-10 19:09:32 | Cuauhtémoc | Doctores | https:/ |
| 2 | Roma norte, amueblado listo para ocuparse | Amueblado, Interior, Muy Bien Ubicado, Listo P... | 11000 | MXN | 2019-11-10 18:22:16 | Cuauhtémoc | Roma Norte | https:/ |
| 3 | Dep amueblado junto Ciudad judicial | Hermoso departamento totalmente amueblado y eq... | 12000 | MXN | 2019-11-10 17:39:23 | Cuauhtémoc | Doctores | https:/ |
| 4 | Depto para estudiante | Lindo depto para una sola persona cuenta con u... | 5000 | MXN | 2019-11-10 16:11:06 | Cuauhtémoc | Cuauhtémoc | https:/ |

**Campos que incluye el dataset:**

+ name : nombre del departamento en renta (ingresada por el ususario)
+ desc : descripción del departamento (ingresada por el ususario)
+ price : precio del inmueble en renta
+ currency : moneda asociada al precio
+ date : fecha de alta del inmueble
+ locality : alcaldía o municipio
+ suburb : colonia
+ url : url del inmueble en renta

Los datos se recolectaron el 10 de noviembre de 2019, contiene las carctarísticas de los primero 300 departamentos en renta de una de las alcaldías Miguel Hidalgo, Cuauhtémoc y Benito Juárez de la página *segundamano.mx*. El entorno en el que se extrajeron los datos es educativo, la página web tiene habilitada la entrada de agentes por lo cuál se pudo cumplir cualquier marco legal.

No tengo antecedente de algún estudio previo de este tipo, debido ala plusvalía de los inmueble en la ciudad y al cambio que pudede estar ejerciendo en ella los nuevos formato de renta como *Airbnb* es interesante conocer cuáles son los precios y característica de inmuebles en renta dentro de las colocinas del primer cuadro de la Ciudad de México y esto permitiría hacer una análisis del estatus actual.

Es necesario trabajar en la limpieza de los datos, lo que también me parece un reto importante para el entorno educativo donde se está elaborando la actividad. Los datos más relevante aparecen dentro del texto de las carcaterística. Es necesario filtrar los inmuebles que no son de vivienda (locales y oficinas), es un dataset que planeo trabajar para poder obtener conocimiento y que además del estatus me de una idea del próximo departamento a buscar.



# Código

In [ ]:
```
# Librerías
import urllib.request as urllib2
from bs4 import BeautifulSoup
import csv
import json

# Adaptado del libro Web Scraping with Python - Richard Lawson
def download(url, user_agent='wswp', num_retries=2):
    headers = {'User-agent': user_agent}
    request = urllib2.Request(url, headers=headers)
    try:
        html = urllib2.urlopen(request).read()
    except urllib2.URLError as e:
        print ('Download error:', e.reason)
        html = None
        if num_retries > 0:
            if hasattr(e, 'code') and 500 <= e.code < 600:
                # retry 5XX HTTP errors
                return download(url, user_agent, num_retries-1)
    return html

""" Urls tomadas del análisis del mapa de sitio, se colocan solamen
te las urls de las 3 alcaldías.
    Se puede transformar para cambiar de ciudad además de alcaldía
y municipio.
    Se agrega el filtro de precio para dejar solo los reportados en
precio de 5,000 a 12,000 pesos.
    Dentro del archivo robots.txt se encuentran habilitados todos l
os agentes, se limita el acceso a
    las cuentas personales de los usuarios.
"""
```

```
urls = ['https://www.segundamano.mx/anuncios/ciudad-de-mexico/cuauh
temoc/renta-inmuebles',
       'https://www.segundamano.mx/anuncios/ciudad-de-mexico/miguel
-hidalgo/renta-inmuebles',
       'https://www.segundamano.mx/anuncios/ciudad-de-mexico/benito
-juarez/renta-inmuebles']
filtro = '&precio=5000-12000'

""" La lógica del código es tomar las primeras 10 páginas de cada d
ataset, en cada una aparecen los
    datos principales de 30 departamentos, de cada uno de ellos se
guarda su url para posteriormente
    explorar detalles adicionales de cada uno de ellos.
    En ambos casos la información está dentro de un formato json in
tegrado en el html.
"""
webs = []
for u in urls:
    for i in range(10):
        url = u + '?pagina=' + str(i+1) + filtro
        html = download(url)
        soup = BeautifulSoup(html, 'lxml')
        json_extract = soup.find(attrs={"type":"application/ld+json
"})
        json_output= BeautifulSoup(str(json_extract), 'lxml')
        json_text=json_output.get_text()
        json_data = json.loads(json_text)
        for j in range(30):
            data1 = json_data['itemListElement'][j]["url"]
            webs.append(data1)

# Se guardan el nombre, descripción, precio, moneda, fecha, ubicaci
ón (ciudad, alcaldía)
name = []
desc = []
price = []
currency = []
date =[]
locality = []
suburb = []
k = 0

# En cada una de las urls se van a buscar los datos antes mencionad
os
for w in webs:
    html_ = download(webs[k])
    soup_ = BeautifulSoup(html_, 'lxml')
    json_extract_ = soup_.find(attrs={"type":"application/ld+json"}
)
    json_output_= BeautifulSoup(str(json_extract_), 'lxml')
    json_text_=json_output_.get_text()
    json_data_ = json.loads(json_text_)
    name.append(json_data_[0]['name'])
    desc.append(json_data_[0]['description'])
    price.append(json_data_[0]['offers']['price'])
    currency.append(json_data_[0]['offers']['priceCurrency'])
    date.append(json_data_[0]['offers']['availabilityStarts'])
```

```
    locality.append(json_data_[0]['offers']['areaServed']['address'
]['addressLocality'])
    suburb.append(json_data_[1]['itemListElement'][5]['name'])
    k += 1

# Se unen cada una de las listas más la url en un listado general
segunda_m = zip(name,desc,price,currency,date,locality,suburb,webs)

# Se guarda el listado en una csv
with open('rentcdmx_segundamano.csv', "w") as csvfile:
    writer = csv.writer(csvfile)
    writer.writerow(['name','desc','price','currency','date','local
ity','suburb','url'])
    for row in segunda_m:
        writer.writerow(row)
```