Capturar datos de la web

November 22, 2022

M2.851 - Tipología y ciclo de vida de los datos

2022-1 · Máster universitario en Ciencia de datos (Data science)

Estudios de Informática, Multimedia y Telecomunicación

El presente documento constituye el informe de la práctica 1 de la asignatura Tipología y ciclo de vida de los datos del Máster universitario en Ciencia de Datos de la Universidad Oberta de Catalunya.

`José Ángel Rodríguez Murillo`

Noviembre de 2022

1 Introducción

La tarea de la práctica consiste en la ejecución de técnicas de web scrapping sobre una web de nuestra elección para obtener un set de datos.

2 Resolución

Se plantea la necesidad de resolver una serie de cuestiones:

2.1 Contexto

Se ha realizado web scrapping sobre una web española de venta de libros: https://www.penguinlibros.com/es/

Esta web es una catálogo de libros de la división española de las editoriales de los grupos Pearson y Bertelsmann donde se pueden consultar precios, links a lugares de compra e información variada sobre audiolibros, libros físicos, digitales, etc.

El objetivo de hacer scrapping sobre esta web es la de obtener un set de datos con información de todos los libros diponibles y sus datos.

La web fue creada en *2020*, es registrada por *NOMINALIA INTERNET S.L.* y pertenece a la organización *Penguin Random House Grupo Editorial, S.A.U.*:

[1]:
```
import requests
from bs4 import BeautifulSoup
```

```
from builtwith import builtwith
website = 'https://www.penguinlibros.com/es/'
```

[2]:
```
builtwith(website)
```

[2]:
```
{'web-servers': ['Nginx'],
 'font-scripts': ['Google Font API'],
 'tag-managers': ['Google Tag Manager'],
 'web-frameworks': ['Twitter Bootstrap'],
 'javascript-frameworks': ['jQuery']}
```

Si se analiza el fichero robots.txt se encuentran de unos ficheros `.xml` públicos donde están los catálogos de libros en venta español, catalán y también para el dominio argentino:

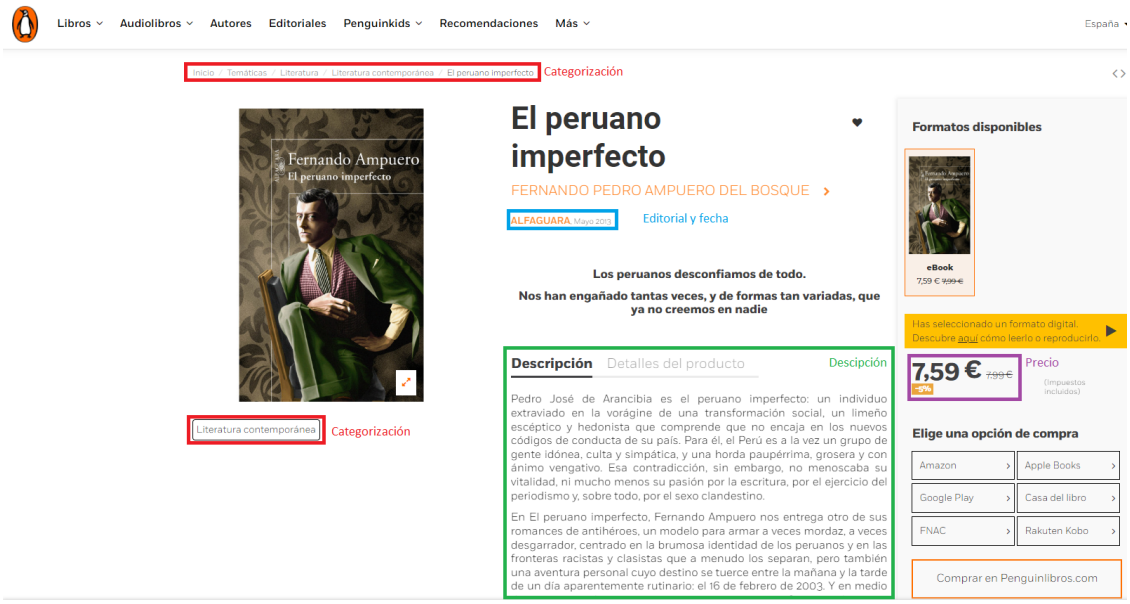
[3]:
```
import requests
robots = "https://www.penguinllibres.com/robots.txt"
headers = {
        "User-Agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_3)␣
 ↪AppleWebKit/537.36 (KHTML, like Gecko) Chrome/56.0.2924.87 Safari/537.36",
        "features":"xml"
    }
page = requests.get(robots, headers = headers)
site = BeautifulSoup(page.content)
site.p.text.split("\n")[32:36]
```

[3]:
```
['Sitemap: https://penguinlibros.com/sitemap-products-1-es.xml',
 'Sitemap: https://penguinlibros.com/sitemap-products-4-ag.xml',
 'Sitemap: https://penguinlibros.com/sitemap-products-23-es.xml',
 'Sitemap: https://penguinlibros.com/sitemap-products-22-ca.xml']
```

A través de esos ficheros se puede acceder a las url de cada uno de los libros del catálogo. La página web de cada libro tiene la siguiente estructura:

[1]:
```
from IPython import display
display.Image("./img1.png")
```

[1]:



Además de más información en el apartado de detalles del producto.

Iterando sobre todo el set de libros y haciendo scrapping sobre la página de cada libro esta web permite hacer un catálogo que luego se pueda exportar a csv y analizar con herramientas de ciencia de datos.

2.2 Título

El título será **penguinlibros_esp_catalog**

2.3 Descripción del dataset

El dataset está construído sobre un fichero `.csv` que contiene filas correspondientes a cada libro.

De cada libro se dispone de la información:

[4]:
```
import pandas as pd
catalog = pd.read_csv("dataset/penguinlibros_esp_catalog.csv")
catalog = catalog.iloc[: , 1:]
for key in catalog.keys():
    print(key)
```

```
Editorial
Fecha
generos
temáticas
titulo
Colección
Páginas
Target de edad
```

```
Tipo de encuadernación
Idioma
Fecha de publicación
Autor
description
price
Serie-Saga
Traductor
Dimensiones
Personaje
nombre_etemocion
```

Se muestra un ejemplo:

[5]:
```
catalog.head()
```

[5]:

	Editorial	Fecha	generos	temáticas	titulo	Colección	Páginas	Target de edad	Tipo de encuadernación	Idioma
0	PUNTO DE LECTURA	marzo 2013\r\n	['\n Literatura contemporánea\n …	[]	\r\n\r\n …	SIN DET. ALFAGUARA	0.0	Adultos	eBook	ES
1	SUDAMERICANA	mayo 2013\r\n	['\n A partir de 7 años\n ']	['\n Lecturas (+ 7 años)\n ']	\r\n\r\n …	ESPECIALES	32.0	A partir de 7 años	eBook diseño fijo	ES
2	SUDAMERICANA	mayo 2013\r\n	['\n A partir de 7 años\n ']	['\n Actividades, enigmas y chistes…	\r\n\r\n …	ESPECIALES	32.0	A partir de 7 años	eBook Kindle	ES
3	SUDAMERICANA	agosto 2013\r\n	['\n A partir de 7 años\n ']	['\n Actividades, enigmas y chistes…	\r\n\r\n …	ESPECIALES				
4	SUDAMERICANA	mayo 2016\r\n	['\n Thriller juvenil\n ', '…	[]	\r\n\r\n …	SUDAMERICANA JOVEN				

```
3     32.0   A partir de 7 años              eBook PDF     ES
4    184.0  A partir de 12 años                  eBook     ES

  Fecha de publicación             Autor  \
0           14-03-2013  Fernando Ampuero
1           01-05-2013  Pablo Bernasconi
2           02-05-2013  Pablo Bernasconi
3           01-08-2013  Pablo Bernasconi
4           01-05-2016  Márgara Averbach

                                          description   price Serie-Saga  \
0  Los textos que componen Gato encerrado son pru…  5,69 €        NaN
1  Este libro fue destacado 2007 de la Asociación…  2,84 €        NaN
2  Este libro fue destacado 2007 de la Asociación…  2,99 €        NaN
3  Este libro fue destacado 2007 de la Asociación…  2,84 €        NaN
4  Los que volvieron se inspira en un hecho real…  4,74 €        NaN

  Traductor Dimensiones Personaje nombre_etemocion
0       NaN         NaN       NaN               NaN
1       NaN         NaN       NaN               NaN
2       NaN         NaN       NaN               NaN
3       NaN         NaN       NaN               NaN
4       NaN         NaN       NaN               NaN
```

En todo caso hay muchos datos vacíos (no todos los libros tienen la información de todos los campos) y falta limpieza y postprocesado.

Si bien en la web se dispone de hasta:

[6]:
```
import warnings
warnings.filterwarnings('ignore')

def load_prettify_page(url, headers):
    page = requests.get(url, headers= headers)
    return BeautifulSoup(page.content)

def get_books_urls(url):
    page = load_prettify_page(url, headers)
    books = page.find_all("url")
    urls = []
    for book in books:
        urls.append(book.loc.text)
    return urls
CATALOGS_URLS = {
        "esp": "https://www.penguinlibros.com/sitemap-products-1-es.xml",
        "cat": "https://www.penguinlibros.com/sitemap-products-22-ca.xml"
    }
```

```
len(get_books_urls(CATALOGS_URLS["esp"])) +␣
 ↪len(get_books_urls(CATALOGS_URLS["cat"]))
```

[6]: 49635

libros, por limitaciones de memoria en la ejecución del scrapping, el número de libros registrados es:

[7]:
```
len(catalog)
```

[7]: 2938

Si bien en la web se disponen de muchos más libros e idealmente ejecutando `pra1.py` se pueden llegar a extraer todos.

2.4 Representación gráfica

Un ejemplo de libro recopilado es:

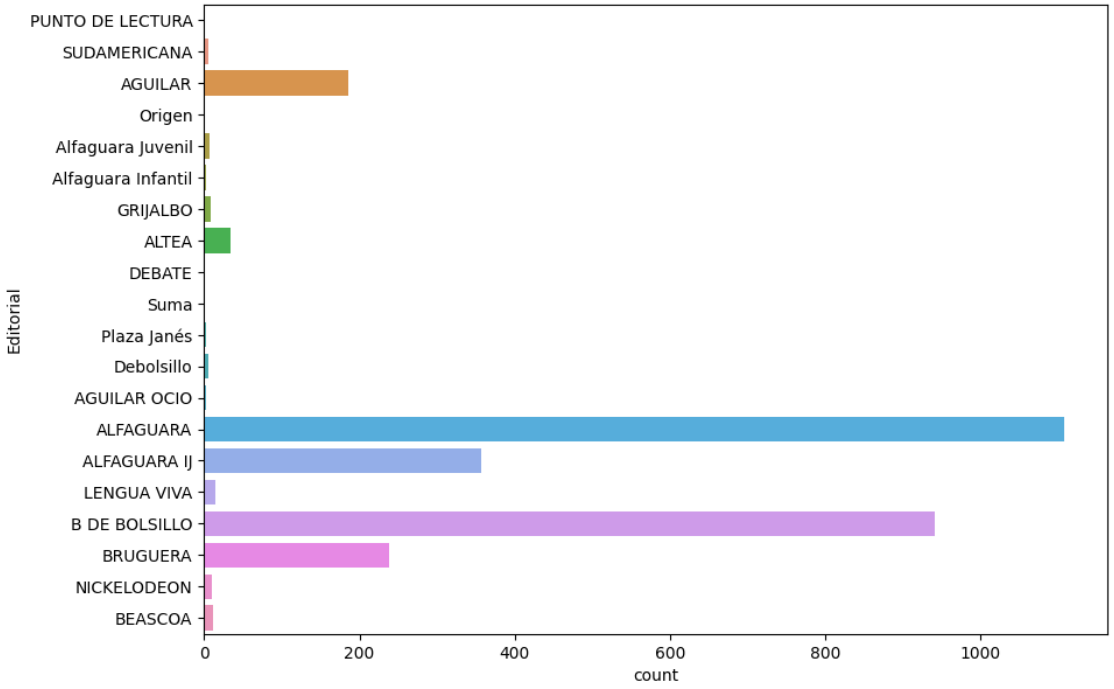
[8]:
```
catalog.iloc[100]
```

[8]:
```
Editorial                                                    AGUILAR
Fecha                        septiembre 2013\r\n                    …
generos                   ['\n            Familia y crianza\n        ', …
temáticas                 ['\n            No Ficción\n        ', '\n    …
titulo                    \r\n\r\n                                     …
Colección                                                 DIVULGACION
Páginas                                                         176.0
Target de edad                                                Adultos
Tipo de encuadernación                          Tapa blanda con solapas
Idioma                                                            ES
Fecha de publicación                                      25-09-2013
Autor                                                 Beatriz Troyano
description               ¿Has descubierto el poder de las enzimas pero …
price                                                         16,15 €
Serie-Saga                                                        NaN
Traductor                                                         NaN
Dimensiones                                             152mm x 230mm
Personaje                                                         NaN
nombre_etemocion                                                  NaN
Name: 100, dtype: object
```

[9]:
```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[10]:
```
def plot_barplot(column):
    plt.figure(figsize=(10, 7))
    sns.countplot(data=catalog, y=column)
```

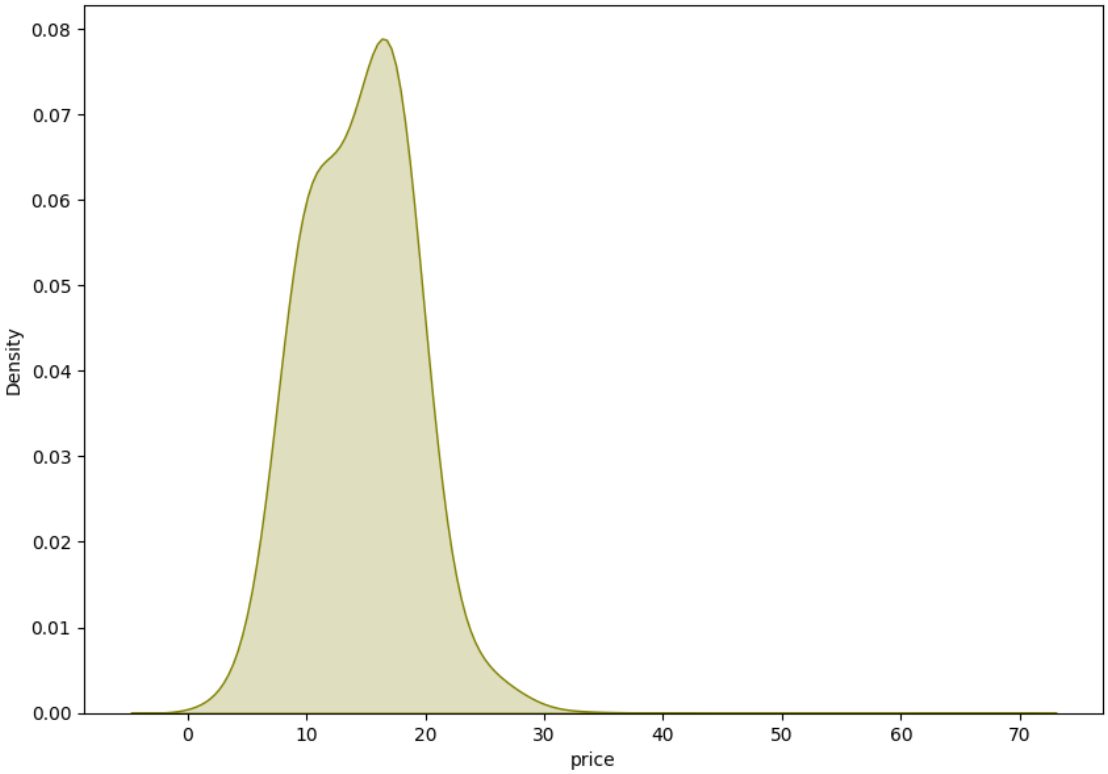
```
def plot_distplot(column):
    plt.figure(figsize=(10, 7))
    sns.kdeplot(catalog[column], shade=True, bw=0.5, color="olive")

plot_barplot("Editorial")
```



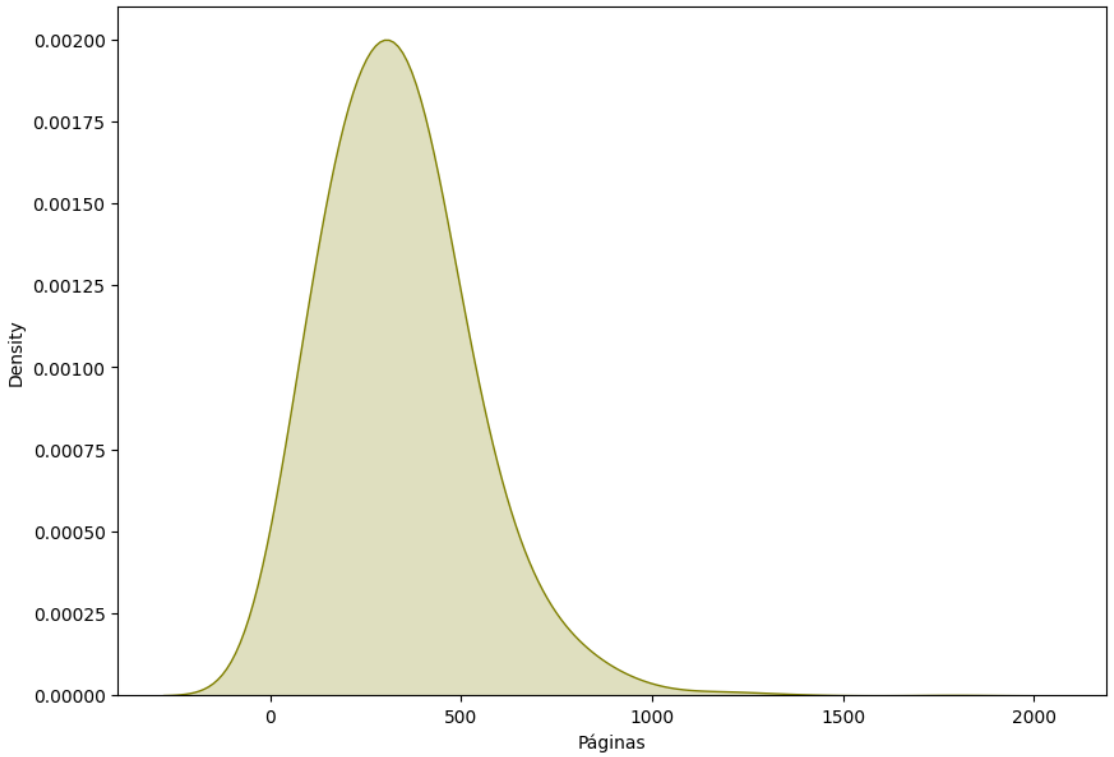
[11]:
```
catalog['price'] = [float(val[:-2].replace(',','.')) for val in␣
 ↪list(catalog['price'])]
```

[12]:
```
catalog['Páginas'] = [float(pag) for pag in list(catalog['Páginas'])]
```

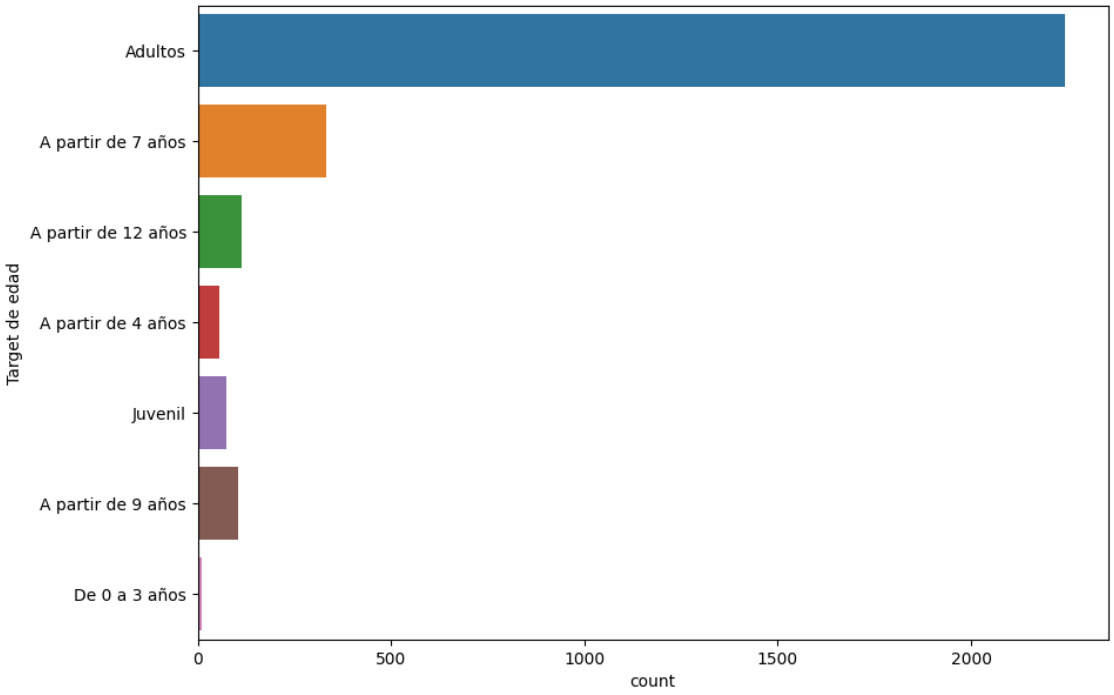
[13]:
```
plot_distplot('price')
```



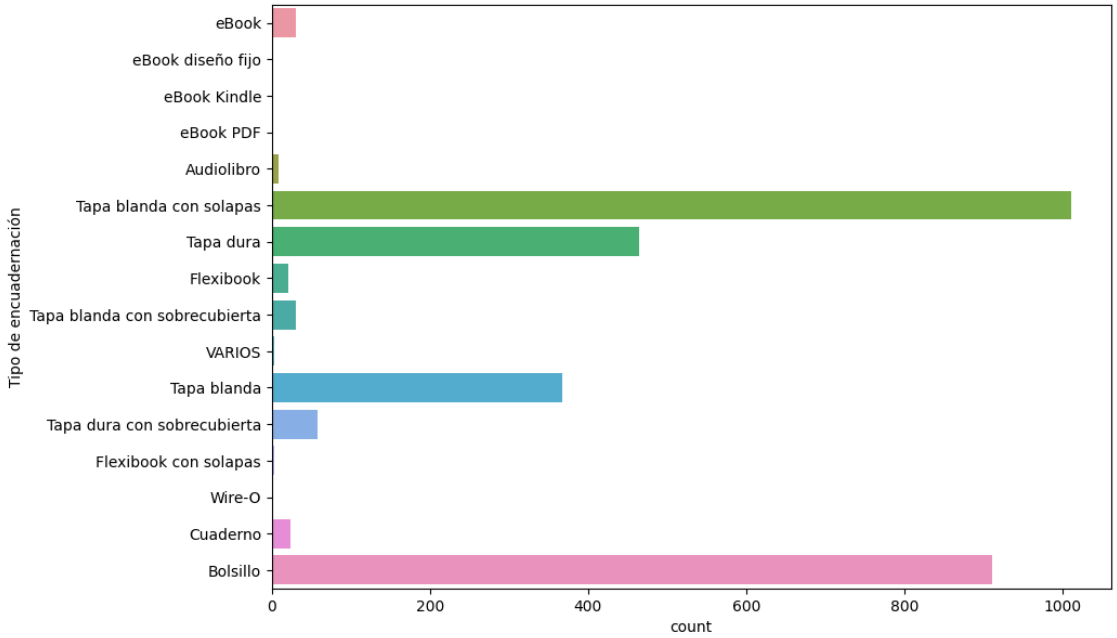
[14]:
```
plot_distplot('Páginas')
```



[15]:
```
plot_barplot("Target de edad")
```



[16]:
```
plot_barplot("Tipo de encuadernación")
```



2.5 Contenido

Los datos se han obtenido sin intervalos de tiempo, esto es en lo que tarda el procesador en ejecutar cada consulta.

Los campos son: - `Editorial`: Editorial del libro - `Fecha`: Fecha de edición del libro - `generos`: Géneros literarios a los que corresponde el libro - `temáticas`: Temáticas a las que corresponde el libro - `titulo`: Título del libro - `Colección`: Colección a la que pertenece el libro - `Páginas`: Número de páginas del libro - `Target de edad`: Target de edad para el que el libro va dirigido - `Tipo de encuadernación`: Tipo de encuadernación del libro (ebook si es electrónico) - `Idioma`: Idioma del libro - `Fecha de publicación`: Fecha de publicación del libro - `Autor`: Autor del libro - `description`: Descripción del libro - `Serie-Saga`: Serie o saga a la que pertenece el libro - `Traductor`: Traductor del libro al idioma en el que se vende - `Dimensiones`: Dimensiones físicas del libro - `Personaje`: Personajes principales del libro

2.6 Propietario

2.6.1 Propietario

El propietario de la web y los datos es la organización:

[17]:
```
import whois
print(whois.whois(website)["org"])
```

`Penguin Random House Grupo Editorial, S.A.U.`

2.6.2 Análsis anteriories

- Amazon books dashboard: Dashboard sobre un dataset de libros obtenido con web scrapping de amazon
- EDA books: EDA sobre dataset de libros resultado de scrapping sobre books.toscrape
- Reccomendation system: anime: Sistema de recomendación desarrollado sobre dataset obtenido con web scrapping sobre myanimelist

2.6.3 Legalidad y ética

Con respecto a avisos legales: - En el aviso legal no se hace referencia a cuestiones de scrapping o automatización de consultas, si bien tampoco se han aceptado los términos. - Se ha consultado y respetado el fichero '`robots.txt` - No hay infracción de leyes de fraude, allanamientos de morada, derechos de autor.

Con respecto a cuestiones éticas. - La información es pública y no es sensible en ningún caso.

2.7 Inspiración

El interés de este dataset es tener una visión más general del catálogo que ofrece la web, pudiendo hacer filtros más específicos por géneros/subgéneros/autores/precio, etc.

Además se pueden hacer análisis estadísticos, obtener información sobre los targets de clientes de la empresa o su política de ventas. Incluso poder eventualmente comprarar la oferta y las políticas de las editoriales con la competencia.

También se pueden utilizar técnicas de minería de datos para extraer información adicional, crear sistemas de recomendación, etc.

2.8 Licencia

Se distribuye con la licencia - *Creative Commons Attribution 1.0 Generic*

Al ser la menos restrictiva que permite Zenodo.

2.9 Código

Se encuentra en el fichero `pra1.py`

2.10 Dataset

Se encuentra en el fichero `penguinlibros_esp_catalog.csv`

2.11 Vídeo

link: Pueden verlo los usuarios `.uoc.edu`

3 Contribuciones

Contribuciones	Firma
Investigación previa	José Ángel Rodríguez Murillo
Redacción de las respuestas	José Ángel Rodríguez Murillo
Desarrollo del código	José Ángel Rodríguez Murillo
Participación en el vídeo	José Ángel Rodríguez Murillo