Gestión de la Información en la Web
Curso 2024-25
Práctica 5 – Web Scraping

Fecha de entrega: domingo 3 de noviembre de 2024

Entrega de la práctica
La entrega de la práctica se realizará a través del Campus Virtual de la asignatura mediante un fichero `pr5.py`. El esqueleto de este fichero se puede descargar del Campus Virtual.

Lenguaje de programación
Python 3.11 o superior.

Calificación
Se medirá la corrección mediante tests de unidad. Además de la corrección, se valorará la **calidad, concisión y claridad del código**, la incorporación de **comentarios** explicativos, su **eficiencia** tanto en tiempo como en memoria y la puntuación obtenida en Pylint.

Declaración de autoría e integridad
Todos los ficheros entregados contendrán una cabecera en la que se indique la asignatura, la práctica, el grupo y los autores. Esta cabecera también contendrá la siguiente declaración de integridad:

> *Declaramos que esta solución es fruto exclusivamente de nuestro trabajo personal. No hemos sido ayudados por ninguna otra persona o sistema automático ni hemos obtenido la solución de fuentes externas, y tampoco hemos compartido nuestra solución con otras personas de manera directa o indirecta. Declaramos además que no hemos realizado de manera deshonesta ninguna otra actividad que pueda mejorar nuestros resultados ni perjudicar los resultados de los demás.*

No se corregirá ningún fichero que no venga acompañado de dicha cabecera.

Se va a considerar una página web de una tienda online ficticia: `https://books.toscrape.com/`. En la tienda se muestran diferentes libros a comprar agrupados en 50 categorías. En cada categoría se muestran los libros incluidos utilizando paginación de 20 elementos, y por cada libro se muestran datos como la portada, el título, la valoración, el precio, etc. Cada libro además tiene su página específica con la descripción del libro y otros detalles.

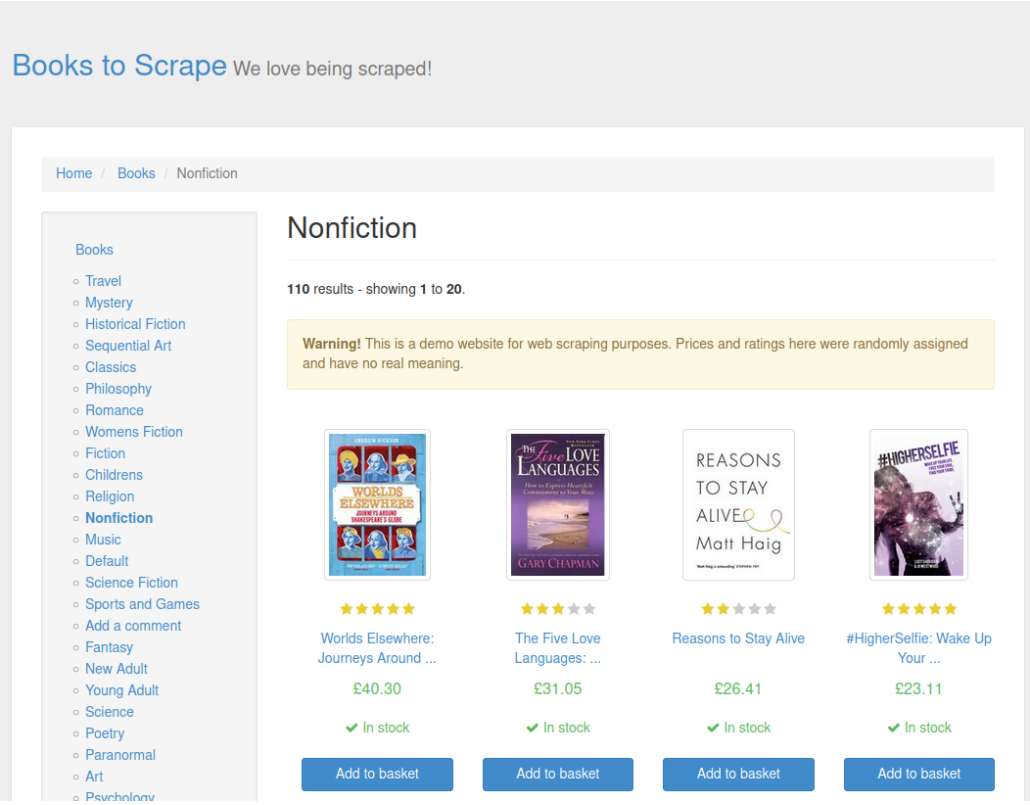

Figura 1: Página principal de la categoría «Nonfiction»

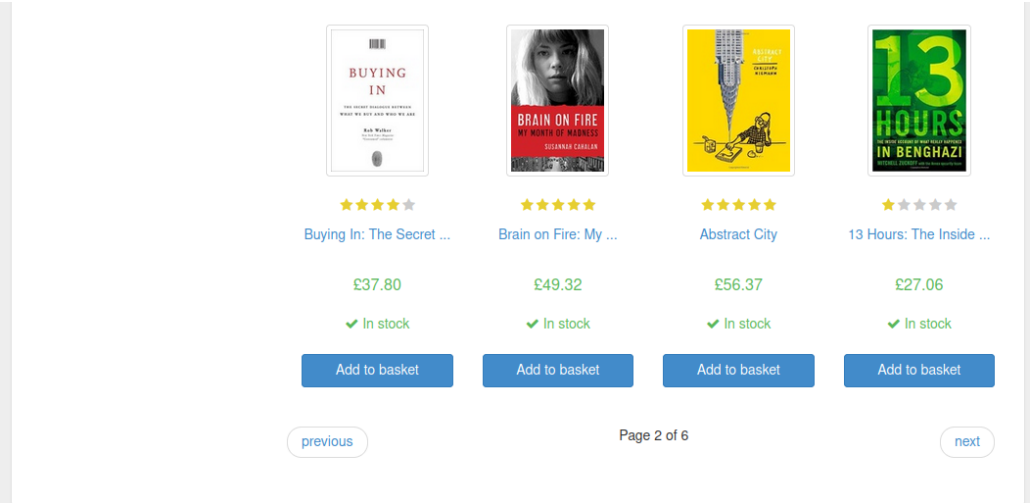

Figura 2: Detalle de paginación

En esta práctica se implementarán distintas funciones Python para extraer información de esta página y devolver listas con los datos buscados.

1. Listado de categorías [4pt]

Implementar una función `categorias()` que devuelve un **conjunto de parejas**

`(nombre, número de libros)`

de todas las categorías que existen en la tienda. Para simplificar este proceso, se *recomienda* implementar también una función auxiliar `explora_categoria(url)` que a partir de la URL de la página principal de una categoría, devuelve el nombre de la categoría y el número de libros.

Ejemplo:

```
>>> categorias()
{('Academic', 1),
 ('Add a comment', 67),
 ('Adult Fiction', 1),
 ('Art', 8),
 ...
}
```

2. Listado de los libros de una categoría [6pt]

Implementar una función `libros_categoria(nombre)` que recibe el nombre de una categoría y devuelve un **conjunto de tuplas**

`(titulo, precio, valoración)`

donde cada tupla representa a un libro, el precio es un número real y la valoración es un número natural. En caso de que no exista ninguna categoría con dicho nombre, devolverá el conjunto vacío.

Ejemplos:

```
>>> libros_categoria('Historical Fiction')
{('A Flight of Arrows (The Pathfinders #2)', 55.53, 5),
 ('A Paris Apartment', 39.01, 4),
 ("A Spy's Devotion (The Regency Spies of London #1)", 16.97, 5),
 ...
}
>>> libros_categoria('Invent')
set()
```

Para simplificar este proceso, se *recomienda* implementar las siguientes funciones auxiliares:

- `url_categoria(nombre)`: devuelve la URL de la página principal de una categoría a partir de su nombre. Para esta búsqueda debéis ignorar espacios al principio y final y también diferencias en mayúsculas/minúsculas.
- `todas_las_paginas(url)`: a partir de la URL de una página, sigue la **paginación hacia delante** devolviendo una lista con todas las **URL absolutas** atravesadas (incluyendo la URL inicial).

Importante: los enlaces de los botones «next» son relativos a la página actual, así que para generar la lista de URL absolutas deberéis utilizar el método `urljoin`[1] de la biblioteca `urllib.parse`. También deberéis ser capaces de transformar la información de la valoración mediante estrellas a un valor numérico entero.

[1] `https://docs.python.org/3/library/urllib.parse.html#urllib.parse.urljoin`