YAZ104 TEMEL PROGRAMLAMA II
Güz 2021-2022
Laboratuvar Ödevi

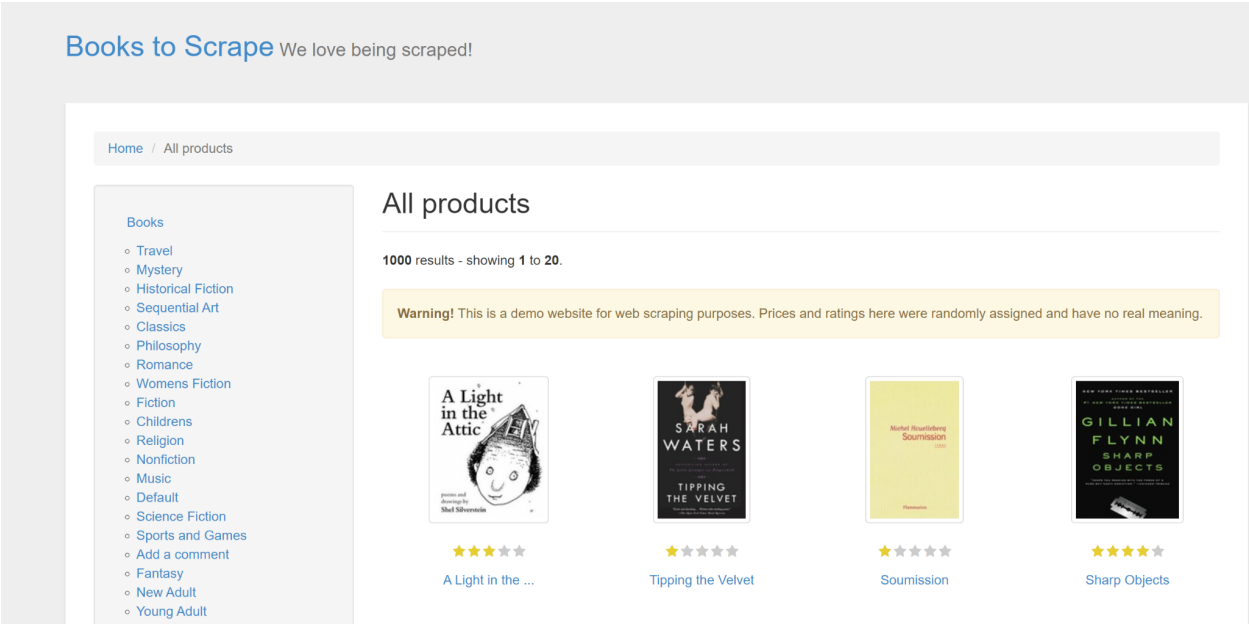
Hafta 08

Veri Kazıma

Laboratuvar raporu (pdf olarak) ve python dosyasını laboratuvar ödevi son yükleme saatinden önce blackboard sistemine yüklemeniz gerekmektedir. Dosyanın ismini ogrenciadi_ogrencisoyadi_numara_labnumarasi ilk isminde olmalıdır (örneğin ahmetcan_erdogan_1111111_lab8.pdf, ahmetcan_erdogan_1111111_lab8.py).

Bu laboratuvar ödevinde sizden verilen bir internet sayfasındaki bilgileri alabilen bir web scrapper (veri kazıyıcı) yazmanız istenmektedir. Sonrasında bu veriler içerisinde birbirine en çok benzeyen kitapları hiyerarşik kümeleme yaparak bulup ekrana bastıracağız. Burada emekleme işlemi **yapmayacaksınız**, onun yerine verilen sayfadaki yapıyı manuel olarak inceleyerek alınmasını istediğiniz bilgileri beatifulsoup ile nasıl çekebileceğinize karar vermeniz gerekecektir.

Tarayacağımız sayfa http://books.toscrape.com/index.html adresidir. Bu sayfada sol tarafta kitap kategorileri ortada ise bu kategorilerde kitaplar, fiyatları ve verilen yıldızlar vardır. Bunu ayıklamak için öncelikle bir WebScrapper sınıfı yaratmalısınız. Bu sınıfın init fonksiyonunda dışarıdan yazılacak database nesnesinin ismi girilebilmelidir. Sınıfın içinde olması gerekli değişkenleri siz belirleyeceksiniz. İçerisinde create_db(self, db_ismi), close_db(self), get_soup(self, webpage), get_categories(self), get_prices_stars(self, soup, link) ve parse(self) fonksiyonlarını içermelidir.



Şekil 1: Veri Kazıması yapılacak Web Sayfası

get_soup(self, webpage):
Dışarıdan verilen **webpage** adresine urllib ile istek gönderip, bilgilerin okunacağı ve dışarıya "soup" nesnesini gönderen fonksiyonudur.

get_categories(self)
Ana sayfada sol taraftaki bulunan kategori listelerini çekebilecek get_categories() fonksiyonunu yazmalısınız. Burada fonksiyonunuz çıktı olarak tüm kategorilerin isimlerini ve kategori linklerini bir sözlükte tutarak vermelidir. Bunun için "soup" nesnesinden hangi elemanı çekmeniz gerektiği ve nasıl bir filtreleme yapmanız gerektiğine karar vermelisiniz. Örnek bir çıktı:
{'travel_2': 'http://books.toscrape.com/catalogue/category/books/travel_2/index.html', 'mystery_3': 'http://books.toscrape.com/catalogue/category/books/mystery_3/index.html', 'historical-fiction_4':....

get_prices_stars(self, soup, link)
Verilen bir soup nesnesi ve bunun yaratıldığı url adresi ile o sayfadaki kitapların isimlerini, puanlarını, fiyatlarını ve url yi aşağıda verilen formatta bir sözlük listesi olarak hazırlar. Rating değişkeni integer, Price değişkeni de float, Name ve URL değişkenleri str formatında olmalıdır:
[{'Name': "It's Only the Himalayas", 'Rating': 2, 'Price': 45.17, 'URL': 'http://books.toscrape.com/catalogue/its-only-the-himalayas_981/index.html'}, {'Name': 'Full Moon over Noah's ...', 'Rating': 4, 'Price': 49.43, 'URL': 'http://books.toscrape.com/catalogue/full-moon-over-noahs-ark-an-odyssey-to-mount-ararat-and-beyond_811/index.html'},

create_db(self, db_ismi):
Bu dışarıdan verilen db_ismi değişkeni ile bir shelve database dosyası açmalı ve bunu bir sınıf değişkenine atamalıdır. Aşağıdaki parse fonksiyonu çağırıldığında bulunan değerler bu değişkende saklanacaktır.

close_db(self)
Yukarıda açılan database nesnesinin kapandığından emin olmak için. Parse işlemi bitince çağrılmalıdır.

parse(self)
Bu fonksiyon ise ayıklanan her kategori sitesi için **sadece ilk sayfalardaki** kitapların fiyat ve puanlamalarını bir database nesnesinde tutmaktadır. Bunun için yukarıda yazdığınız get_prices_starts fonksiyonunu kullanabilirsiniz. Örnek yapı:

{'travel_2': [{'Name': "It's Only the Himalayas", 'Rating': 2, 'Price': 45.17}, {'Name': 'Full Moon over Noah’s ...', 'Rating': 4, 'Price': 49.43},...

Daha önce derste kullanmadığımız shelve modülü, dbm gibi verileri sürekli saklayabileceğimiz bir database nesnesi oluşturabiliyor. Mesela aşağıdaki kod öbeği ile kitaplar.db isimli database, yazma ve okuma işlemleri için oluşturuluyor ve prices değişkenine bağlanıyor (aynı dbm de olduğu gibi). Bu modülün "dbm" veritabanlarından farkı, bir shelve nesnesinin değerleri (anahtarlar değil!) temelde keyfi Python nesneleri - turşu modülünün işleyebileceği herhangi bir şey - olabilmesidir. Böylece aşağıdaki price değerlerine istediğiniz yapıyı herhangi bir pickle işlemi yapmadan atabilirsiniz.

```
import shelve
db_name = 'kitaplar.db'

prices = shelve.open(db_name, writeback=True, flag='c')
prices[‘anahtar1’] = {‘sozlukanahtari’: 1000} // Burada anahtar1 için bir sözlük değeri tanımlanmış
prices[‘anahtar2’] = OrnerSinif() // Burada anahtar2 için bir OrnekSınıf nesnesi tanımlanmış
```

Her zamanki gibi oluşturduğunuz bu sınıfı bir test dosyasından çağırmalı ve raporunuzda gerekli ekran görüntülerini paylaşmalısınız. Yukarıdaki format takip edildiğinde test dosyasını aşağıdaki gibi oluşturabilirsiniz:

```
from lab8 import WebScrapper
import shelve

db_name = 'kitaplar.db'
ws = WebScrapper(db_name)
ws.parse()

with shelve.open(db_name, 'r') as db:
    for keys, values in db.items():
        print(keys, ':', values)
```

Örnek çıktı:

travel_2 : [{'Name': "It's Only the Himalayas", 'Rating': 2, 'Price': 45.17, 'URL': 'http://books.toscrape.com/catalogue/its-only-the-himalayas_981/index.html'}, {'Name': 'Full Moon over Noah’s ...', 'Rating': 4, 'Price': 49.43, 'URL': 'http://books.toscrape.com/catalogue/full-moon-over-noahs-ark-an-odyssey-to-mount-ararat-and-beyond_811/index.html'}, {'Name': 'See America: A Celebration ...', 'Rating': 3, 'Price': 48.87, 'URL': 'http://books.toscrape.com/catalogue/see-america-a-celebration-of-our-national-parks-treasured-sites_732/index.html'}, …],mystery_3 : [{'Name': 'Sharp Objects', 'Rating': 4, 'Price': 47.82, 'URL': 'http://books.toscrape.com/catalogue/sharp-objects_997/index.html'}, {'Name': 'In a Dark, Dark ...', 'Rating': 1, 'Price': 19.63, 'URL': 'http://books.toscrape.com/catalogue/in-a-dark-dark-wood_963/index.html'}, {'Name': 'The Past Never Ends', 'Rating': 4, 'Price': 56.5, 'URL': 'http://books.toscrape.com/catalogue/the-past-never-ends_942/index.html'},...],...}