Instituto Tecnológico de Costa Rica Unidad de computación

Primera investigación.

Estudiante:

Kimberly Morales Arias 2014096395

Sede San Carlos.

Fecha de entrega: 15 de mayo de 17

Introducción	3
Desarrollo:	3
BeautifulSoup:	3
Implementación:	3
Planteamiento:	9
Conclusiones:	9
Bibliografía	10

Introducción

Actualmente en internet la transferencia de datos se ve reflejada en niveles impresionantes, en un minuto millones de correos electrónicos son enviados, en 24 horas la cifra es impresionantemente grande. Existen muchas páginas en internet, algunas estáticas que únicamente brindan información al usuario y otras dinámicas que permiten la interacción de los usuarios, por ejemplo: páginas de compras en línea. Toda esta información es pública y puede consultarse con diversos fines desde cualquier parte del mundo, las grandes empresas o negocios pagan por esta información, por datos importantes que ayudan a construir estadísticas, a raíz de la necesidad de obtener esta información lo más rápido y exacta posible nace el "web scraping" que consiste en una técnica que logra extraer información de las páginas de forma automatizada.

Desarrollo:

BeautifulSoup:

Librería para Python que permite la extracción sencilla de datos específicos de un sitio web en HTML sin mucha programación, una de las ventajas es que toda la información resultante de la extracción de datos lo hacen en formato UTF-8. Esta librería permite la realización del web scraping de cualquier página, además de ser sencilla de utilizar implementa pocas líneas de código y permite la captura de información en formato html como lxml.

Implementación:

Primero deberán ejecutarse algunos comandos para la instalación de la librería:

Se debe acceder a la carpeta donde se ubique python en C, una vez que estemos ahí ejecutamos el archivo get-pip.py.

En la línea de comandos colocamos: pip install beatifullsoup4

Una vez ya terminada la instalación ingresamos: pip install lxml

Creamos un archivo y empezamos a trabajar.

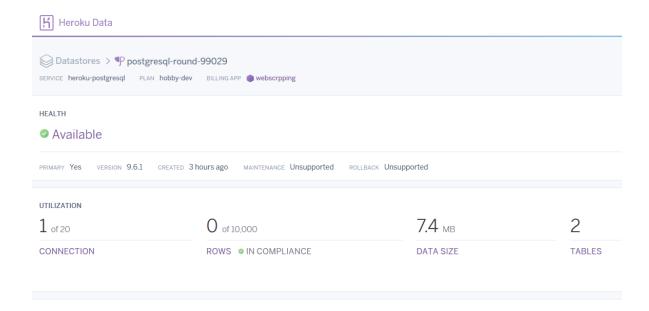
Vamos a crear un archivo python con el siguiente código:

```
import bs4 as bs #se realiza el import de beautiful soup
import urllib.request
from html.parser import HTMLParser
sauce
urllib.request.urlopen('http://crautos.com/rautosnuevos/').re
ad()
soup = bs.BeautifulSoup(sauce,'lxml')
for table in soup.find all(class ='cardetail'): #de esta forma
se puede especificar el campo deseado
    print(table.text.strip())
for table in soup.find all(class ='small'):
   print(table.get('').strip())
    print(table.text.strip())
for url in soup.find all('a'):
```

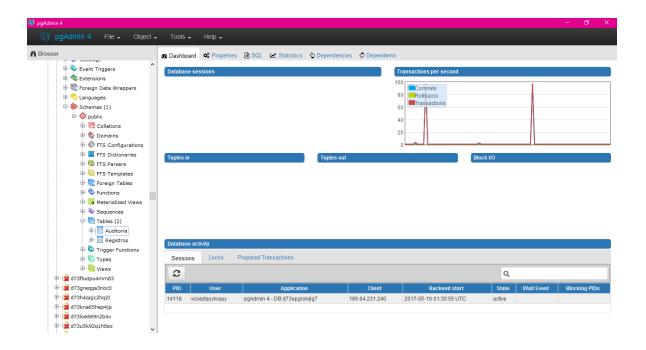
```
print(url.get('href').strip())
Salida del programa:
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 17:54:52) [MSC v.1900 32 bit (Intel)]
Type "copyright", "credits" or "license()" for more information.
== RESTART: C:\Users\Kimberly M\Downloads\Tarea 2 Kimberly\webScrapping.py ==
$108,000$1343/mes
Jaguar
                               F-PACEGasolina
$ 98,000$1218/mes
Mercedes Benz
                             Clase GLE250d 4M
$ 44,900$ 558/mes
Mitsubishi
                              L200GLS HP MT5
$ 39,900$ 496/mes
Mitsubishi
                              L200GLS
$ 54,900$ 682/mes
Mitsubishi
                             Montero WagonGLS AT PLUS
$ 85,600$1064/mes
Toyota
                               TundraTU34 Limited Crew Max
$ 59,900$ 745/mes
Ford
```

Información como esa será la que se almacenará en la base de datos.

Se crea la cuenta y la base de datos en Heroku:



Por medio del pgAdmin4 de postgress, se enlaza y sincroniza la base de datos.



Después se realiza la conexión con Python para poder emplearla.

Connecting in Python

To use PostgreSQL as your database in Python applications you will need to use the psycopg2 package.

```
$ pip install psycopg2
$ pip freeze > requirements.txt
```

And use this package to connect to DATABASE_URL in your code:

```
import os
import psycopg2
import urlparse

urlparse.uses_netloc.append("postgres")
url = urlparse.urlparse(os.environ["DATABASE_URL"])

conn = psycopg2.connect(
    database=url.path[1:],
    user=url.username,
    password=url.password,
    host=url.hostname,
    port=url.port
)
```

Para Python 3 utilizar: import urllib.parse as urlparse en lugar de urlparse

Con el siguiente código se puede conectar a la base de datos:

```
import psycopg2
import urllib.parse as urlparse
import os
```

url =

urlparse.urlparse(os.environ['postgres://vcoeytqsykvquy:da3ec 31136524582d9a2d15da762ce5fcf28d3af0df2ed750ad3c23355e9f209@e

```
c2-107-20-141-145.compute-
1.amazonaws.com:5432/d73epgrslvjlg7'])
dbname = url.path[1:]
user = url.username
password = url.password
host = url.hostname
port = url.port
con = psycopg2.connect(
            dbname=dbname,
            user=user,
            password=password,
            host=host,
            port=port
)
```

Planteamiento:

Se eligió una página web costarricense dedicada a la venta de autos llamada crautos, estos autos poseen muchísimas características que son de gran relevancia. Mediante el lenguaje de programación Python y el uso de la herramienta BeautifulSoup se logró extraer alguna información, aunque no toda.

Conclusiones:

Existen varias formas de realizar la extracción de información de sitios y páginas web, sin embargo se debe analizar muy bien el manejo que se harán con los datos, tomando en cuenta que aunque la información sea pública esta tiene un autor quien es el propietario legítimo, al hacer web scraping estamos extrayendo una información ajena y empleándola en fines propios, aunque a veces pareciera inofensivo a veces puede ser muy valiosa esta captura de información por lo cual se debe ser muy responsable.

Bibliografía

- Debina Laishram, Merin Sebastian. (2015). Extraction of Web News from Web Pages Using a Ternary Tree Approach. Advances in Computing and Communication Engineering (ICACCE), 2015 Second International Conference. IEEE.
- Deepak Kumar Mahto, Lisha Singh. (2016). A Dive into Web Scraper World. *International Conference on Computing for Sustainable Global Development (INDIACom)*.
- Eloisa Vargiu1, 2, Mirko Urru1. (2012). Exploiting web scraping in a collaborative filteringbased. Artificial Intelligence Research, 2013, Vol. 2, No. 1.
- Malik, Sanjay Kumar; , SAM Rizvi;. (s.f.). Information Extraction Using Web Usage Mining, Web Scrapping and Semantic Annotation. *Computational Intelligence and Communication Networks (CICN), 2011 International Conference.* IEEE.
- Martí, M. (08 de 04 de 2016). *sitelabs*. Obtenido de https://sitelabs.es/web-scraping-introducciony-herramientas/
- Rizqi Putri Nourma Budiarti , Nanang Widyatmoko , Mochamad Hariadi and Mauridhi Hery Purnom. (2016). Web Scraping for Automated Water Quality. *International Seminar on Intelligent Technology and Its Application*.
- Richardson, L. (2015). *Beautiful Soup Documentation*. Obtenido de https://www.crummy.com/software/BeautifulSoup/bs4/doc/