Anamitra Bhattacharyya
PREDICT-452 Sect. 55
Individual Assignment 1,Option 1
Automated Data Acquisition

Executive Summary

The aim of this project was to extract website information from the Chicago-based Divvy Bike Share Program, not just from the main landing page for the site, describing aspects of the rideshare program, but also from the data sources that are made available by Divvy for third parties for data analytics purposes. The latter will be used in my subsequent Capstone/thesis project or my extracurricular data analytics endeavors, to build models and improve the bike share program. Website location information for rider and bike station usage data files were successfully identified in the project. The extracted information was not parsed extensively, except to extract the links to the ridership data sources.

Research design and methods

The strategy used for this exercise was to use a Python program (using the Scrapy package) designed to crawl and scrape data from the Divvy Bikeshare web-site (https://www.divvybikes.com/). Specifically, using the main landing page, a web spider would be used to obtain text and other website information from the main pages and link to other internal referenced pages and output the data in JSON file format (Javascript Object Notation). In addition, the URLs for the data sources, comprising rider usage statistics and ride station location, were identified.

Implementation and programming

The Python package 'scrapy' was used to develop a Python program (AB_run_one_site_crawler.py) that called a web spider helper program (run_one_site_crawler.py) directed to the web location for the Divvy Bikeshare program. The web spider program this spider is designed to crawl just one website.

Main program: **AB_run_one_site_crawler.py** (main program that calls a web spider class helper program, there a part of the program that extracts HTML links to the data sources comprises rider/station information). The results of the web scraping were captured in JSON format in 2 files (*e.g.* divvyitems.json,

divvyitems.jsonlines) The main program also contained code that employed the BeautifulSoup package to extract all the data source links from rider usage and station data from 2013-2017 (residing at the URL: https://www.divvybikes.com/system-data). The output was printed to the command line but captured in a text file (**DataURLOutputfromDivvy.txt**; see 'Output files' below).

Spider program: **one_site_crawler.py** (modified to go to the Divvy main website)

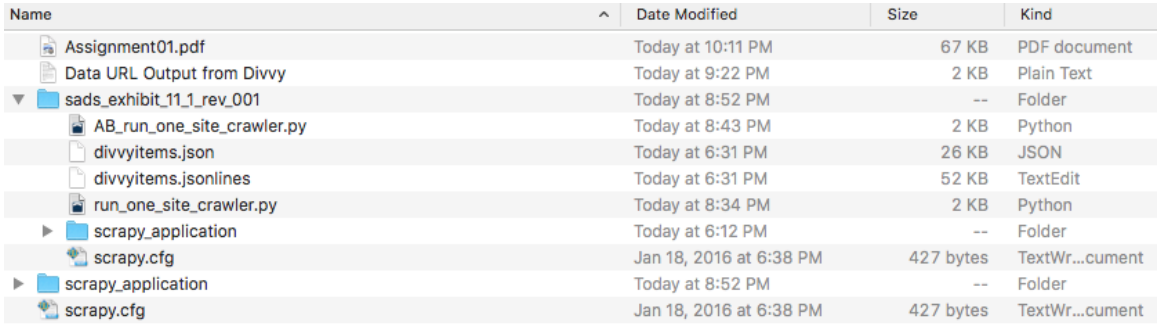
Output files:

DataURLOutputfromDivvy.txt: URL links to data sources from the data sources page on Divvy Bike

divvyitems.json: JSON format of the web scraped data from the project

divvyitems.jsonlines

Directory structure of submission



Name	Date Modified	Size	Kind
Assignment01.pdf	Today at 10:11 PM	67 KB	PDF document
Data URL Output from Divvy	Today at 9:22 PM	2 KB	Plain Text
sads_exhibit_11_1_rev_001	Today at 8:52 PM	--	Folder
AB_run_one_site_crawler.py	Today at 8:43 PM	2 KB	Python
divvyitems.json	Today at 6:31 PM	26 KB	JSON
divvyitems.jsonlines	Today at 6:31 PM	52 KB	TextEdit
run_one_site_crawler.py	Today at 8:34 PM	2 KB	Python
scrapy_application	Today at 6:12 PM	--	Folder
scrapy.cfg	Jan 18, 2016 at 6:38 PM	427 bytes	TextWr...cument
scrapy_application	Today at 8:52 PM	--	Folder
scrapy.cfg	Jan 18, 2016 at 6:38 PM	427 bytes	TextWr...cument

Conclusions

Website information from the Chicago-based Divvy Bike Share Program describing aspects of the rideshare program rider as well URLs of rider usage and bike station usage data files were successfully identified in the project. The extracted information from the Divvy Bike Share site was not parsed, except to extract the links to the ridership data sources, but was stored in JSON format for later parsing and analysis.