webcrawlAll

Table of contents

- Introduction
- Documentation

Introduction

ProfileManager was designed for the aggregation of information related to corporate entities to support building business profiles. It uses the United States Securities and Exchange Commission (SEC) Central Index Key (CIK) to act as universally unique identifiers (UUIDs) and allows the user to compile a variety of information on corporate entities in an easy to use and query format because each profile is a dictionary. Assisting the accessibility of information, **Profile Manager** includes a series of mappings from CIK codes to names and back, names to aliases, and mappings from industry codes (namely The North American Industry Classification System (NAICS) and Standard Industrial Classification (SIC) codes) and descriptions of them. The hope to provide for a flexible data solution for complex business oriented applications.

Documentation

crawlerWrapper(search_query, enging, doSetDriver, headless = False)

Takes in the query to search for on a portal. NOTE: Saves to file, does not return anything.

Currently supported portals:

1. google: searches the google page for entered company
2. sec10k: searches the 10k filing for that company
3. sec10kall: finds the 10Ks of all the companies
4. secsic10k: uses the SIC codes to gather all the companies' 10K in all SICs
5. generalSEC: performs any general query on SEC using `urlmaker_sec`
6. sitespecific: Performs Crawling specifically on any particular website
7. tri: Returns the TRI page for given facility ID
8. everything-all: finds the 8Ks, 10Ks and EX-21s of all the companies on the SEC website, via CIK

Parameters

- search_query (dict) : the format for search query for different engines is as follows:

    i. google
    - name (str): the mandatory portion of the search query
    - aliases (str[]): optional words of the search query
    - filetype (str): filetype to be searched for

    i. sec10k:
    - cik (str): String typed CIK code of the company
    - dateStart (str): Starting date of filings, '/' seperated date, MM/DD/YYYY
    - dateEnd (str): Ending date of filings, '/' seperated date, MM/DD/YYYY

    i. sec10kall:
    - None

    i. secsic10k:
    - None

    i. generalSEC:
    - searchText (str) : Company name to be searched (Default: '*')
    - formType (str): Type of the document to be retrieved (Default: '1')
    - sic (str): SIC code for the companies to be searched (Default: '*')
    - cik (str): CIK code for the company to be searched (Default: '*')
    - startDate (str): Start date of the produced results (YYYYMMDD) (Default: '*')
    - endDate (str): End date of the produced results (YYYYMMDD) (Default: '*')
    - sortOrder (str): Ascending (Value = 'Date') or Descending (Value = 'ReverseDate') retrieval of results, (Default: 'Date')

    i. tri:
    - tri_id (str): TRI ID of the facility

    i. google-subs:
    - name: name of the company whose subsidiaries have to be discovered

    i. everything-all:
    - None

- search_query (str)

    i. sitespecific - Default Settings:
    ii. -p0: only parse URLS, don't download anything
    iii. -%I : make an index of links
    iv. set depth of 5
    v. language preference: en
    vi. -n: get non-HTML files near an HTML
    - search_query['name']: url of the website we need to download
    - -O output directory
    - -r set the depth limit

    - -m, non-HTML,HTML file size limit in bytes
    - %e, number of external links from the targetted website
    - '%P0' don't attempt to parse link in Javascript or in unknown tags
    - -n get non-HTML files near an HTML-files (images on web-pages)
    - t test all URLs
    - -%L , loads all the links to be tracked by the function
    - K0 Keep relative links
    - K keep original links
    - -%l "en, fr, *" language preferences for the documents
    - -Z debug log
    - -v verbose screen mode
    - I make an index
    - %I make a searchable index
    - -pN priority mode (0): just scan (1): just get HTML (2): just get non-HTML (3): save all files (7): get HTML files first, then treat other files

- engine (str) : specify which type of crawler to use, refer module summary for options

Returns

- None

linkFilter_google(url)

Filters out the links of social media websites from the returned google search results using `filterList` defined implicitly.

Parameters

- url (str) : URL to be tested against `filterList`

Returns

- int : returns 0 (is a social media link), 1 (is not a social media link)

search_google(query, driver, number_of_pages)

Searches Google websites for the top page results

Parameters

- query (dict)
  - name (str): the mandatory portion of the search query
  - aliases (str[]): optional words of the search query
  - filetype (str): filetype to be searched for
- driver (selenium.webdriver.Chrome) : An instance of browser driving engine
- number_of_pages (int) : number of pages of Google web results

Returns

- list of strings : list of links returned from the Google search engine

setDriver(headless = True)

Sets a selenium webdriver object for running web-crawlers on various systems. Note: Requires chromedrivers for various platforms in a chromedrivers directory

Parameters

- headless (bool) : if True, sets a headless browser. if False (Default), sets a browser with head

Returns

- selenium.webdriver.Chrome : driver with standard option settings

urlmaker_sec(queryDic)

Produces the URL, which can be entered into the search (Designed for SEC.gov)

Parameters

- queryDic (dict)
  - searchText (str): Company name to be searched (Default: '*')
  - formType (str): Type of the document to be retrieved (Default: '1')
  - sic (str): SIC code for the companies to be searched (Default: '*')
  - cik (str): CIK code for the company to be searched (Default: '*')
  - startDate (str): Start date of the produced results (YYYYMMDD) (Default: '*')
  - endDate (str): End date of the produced results (YYYYMMDD) (Default: *')
  - sortOrder (str): Ascending (Value = 'Date') or Descending (Value = 'ReverseDate') retrieval of results, (Default: 'Date')

Returns

- str : URL to be searched on the SEC website