

# Site\_Crawler\_Parser\_All

---

## 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 is to provide for a flexible data solution for complex business oriented applications.

## Documentation

---

### **company\_to\_product(company, driver)**

Search a company name on EWG and get all products made by the company in EWG database

#### Parameters

- company (str) : A company name to find products for
- driver (selenium.webdriver.Chrome) : Chrome driver after calling 'driver = setDriver()'

#### Returns

- dict
  - COMPANY (str) : a list of products made by the company

### **##### get\_comp\_name(text)**

---

Extract relevant content of company name in a html tag

## Parameters

- text (str) : raw content in a html tag

## Returns

- str : a clean company name after junk texts are filtered

## #####

## get\_parent\_child\_dict(company,parent,children\_list)

---

Build a dictionary that contains parent company, subsidiary company information for a certain company

## Parameters

- company (str) : company name to build dictionary for
- parent (str) : The parent company name for the company
- children\_list (list of strings) : list of subsidiary names of the company

## Returns

- dict :
  - parent (str) : the parent company name
  - child (list of str) : a list of subsidiary names of the company

## ##### get\_recursive\_sub(company, driver)

---

Search "COMPANY\_NAME+subsidiaries" RECURSIVELY on google chromedrivers directory to get all-level subsidiaries of a company and build a master dictionary that contains all-level subsidiary information for a company

## Parameters

- company (str) : company name to find subsidiary for
- driver (selenium.webdriver.Chrome) : Chrome driver after calling 'driver = setDriver()'

## Returns

- dict

- company (str) :
- parent (str) : the parent company of the company, 'NA' if not found
- child(list): a list of subsidiary names

## ##### get\_tri\_dict(tri\_id, driver)

---

Open facility report page and scrape facility information into a dictionary

### Parameters

- tri\_id (str) : TRI facility id used as a unique identifier for a facility on TRI Search
- driver (selenium.webdriver.Chrome) : Chrome driver after calling 'driver = setDriver()'

### Returns

- fac\_dict (dict)
  - fac\_name(str): Facility Name
  - tri\_id(str): TRI facility ID
  - address(str): Facility Address
  - frs\_id(str): FRS ID
  - mailing\_name(str): Facility Mailing Name
  - mailing\_address(str): Facility Mailing Address
  - duns\_num(str): Facility Duns Number
  - parent\_company(str): Facility's Parent Company Name
  - county(str): County
  - pub\_contact(str): Public Contact Name
  - region(str): EPA Region Code
  - phone(str): Contact Number
  - latitude(str): Latitude
  - tribe(str): Tribe
  - longitude(str): Longitude
  - bia\_tribal\_code(str): BIA Tribal Code
  - naics(str): Naics Code
  - sic(str): SIC Code
  - last\_form(str): Last Year of Report

## ##### google\_sub(company, driver)

---

Search "COMPANY\_NAME+subsidiaries" on google chromedrivers directory and scrape the knowledge graph results of subsidiary names returned by Google on the top

## Parameters

- company (str) : A company name to find subsidiary for
- driver (selenium.webdriver.Chrome) : Chrome driver after calling 'driver = setDriver()'

## Returns

- list : a list of subsidiary names(str)

## **hazard\_to\_company(chemical,driver)**

Search NPIRS by entering a chemical name and get a list of companies that use the chemical in their products in NPIRS database

## Parameters

- chemical (str) : a hazard name
- driver (selenium.webdriver.Chrome) : Chrome driver after calling 'driver = setDriver()'

## Returns

- list of strings : a list of companies that use the hazard

## #####

## **product\_to\_ingredient(comp\_prod\_dict,driver)**

---

Search a product name on EWG, get all ingredients in the product in EWG database, and build a master dictionary that contains information for company-products-ingredients

## Parameters

- comp\_prod\_dict (dict) : dictionary that contains company to products information after calling 'comp\_prod\_dict = company\_to\_product(company, driver)'
- driver (selenium.webdriver.Chrome) : Chrome driver after calling 'driver = setDriver()'

## Returns

- dict
  - COMPANY (str) :
  - PRODUCT (str): a list of ingredients in the company product

## **remove\_null(comp\_list)**

Remove null values in a company list

Parameters

- `comp_list` (list of strings) : a list of companies

Returns

- `str` : a clean list of company names with no null values

## ##### **setDriver(headless = False)**

---

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

## **wikiParser(company)**

Search the Wikipedia page for a company and get wikipedia infobox together with all other contents

Parameters

- `company` (str) : the company you would like to query Wikipedia for

Returns

- tuple
  - dict : a dictionary of all other contents on wikipedia
  - dict : a dictionary of wikipedia infobox
  - str : page title
  - str : page url
  - `beautifulsoup.table` : wikipedia infobox HTML