WEB SCRAPING

## WORKSHEET – 1

In Q1 to Q9, only one option is correct, Choose the correct option:

1. Which of the following extracts information from user generated content?

A) Java script tagging B) Web scraping

C) A/B testing D) MROCs

Answer: (B)

1. Which of the following is not a web scraping library in python?

A) selenium B) Beautiful soup

C) Requests C) scrapy

Answer: (B)

1. Selenium tests \_\_\_\_\_\_\_\_\_\_?

A) Browser based applications B) DOS applications

C) GUI applications D) All of the above

Answer: (A)

1. Task of crawling is performed by a complex software which is known as:

A) Scraper B) Crawler

C) Boat D) Spider

Answer: (B),(C),(D)

1. Which of the following commands is used to access name of a tag in Beautiful Soup?

A) tag.attrs B) tag.name

C) tag,id C) tag[‘id’]

Answer : (B)

1. Which of the following is the default parser in Beautiful Soup?

A) html.parser B) html5lib

C) lxml D) lxml-xml

Answer: (C)

1. In selenium the webdriver is used to?

A) design a test using selenese

B) test a web application on firefox only

C) execute tests on HtmlUnit browser

D) to download any content from a webpage

Answer: (B)

1. In selenium, driver.find\_elements\_by\_xpath(‘given xpath’) returns:

A) the first webelement associated with the ‘given xpath’

B) the url of first webelement associated with the ‘given xpath’

C) the list of all webelements associated with the ‘given xpath’

D) all the attributes of the first webelement associated with the ‘given xpath’

Answer: (D)

1. The script ‘window.scrollBy(0,a) scrolls the webpage by?

A) ‘a’ number of horizontal spaces

B) ‘a’ number of lines

C) ‘a’ number of pixels horizontally

D) ‘a’ number of pixels vertically

Answer: (D)

In Q10, more than one options are correct, Choose all the correct options:

1. Which of the following is(are) tags of HTML?

A) <a> B) <b>

C) <image> D) <href>

Answer: (a)(b)(c)

Q10 to Q13 are subjective answer type questions, Answer them briefly.

1. What is the main difference between a web scraper and a web crawler?

Answer: Web crawlers work by browsing to a series of webpages and analyzing their contents for links to other webpages. The links to the other webpages are then followed and searched for more links. The process of following and recording these links is referred to as “crawling.” While crawling through various web pages can reveal useful information about the structure of the web, extracting data from those sites, or “web scraping”, captures the content of those pages which can then be analyzed to reveal more information about the crawled pages. Many web crawlers utilize web scraping to contextualize the pages that they have crawled.

* The term crawling comes from the way a spider would crawl. That’s why a [web crawler](https://en.wikipedia.org/wiki/Web_crawler) is also sometimes called a spider. It’s basically an internet bot that systematically browses (read crawls) the World Wide Web, usually for the purpose of web indexing.
* It is used for indexing the information on the page using bots also known as crawlers.
* It involves looking at a page in its entirety and indexing it, including its last letter and dot on the page, in the quest for information.
* Crawling through every nook and crevice of the World Wide Web, the spider locates and retrieves the information lying in the deeper layers. Web crawlers or bots navigate through heaps of data and information and procure whatever is relevant for your project.

Example of Web crawling

* What Google, Yahoo or Binge does is a straightforward example of web scraping.
* These search engines crawl web pages and use the information for indexing the web pages.

A [web scraper](https://www.skylynapps.com/scrapemate/)'s main purpose is to extract data from webpages. [Web scrapers](https://www.skylynapps.com/scrapemate/) often have the ability to browse to different pages and follow links. Though [web scrapers](https://www.skylynapps.com/scrapemate/) can crawl to different pages their primary purpose is scraping the data on those pages, not indexing the web.

* [Web scraping](https://prowebscraper.com/blog/what-is-web-scraping/) is basically extracting data from websites in an automated manner.
* It is automated because it uses bots to scrape the information or content from websites.
* It’s a programmatic analysis of a web page to download information from it.
* Data scraping involves locating data and then extracting it. It does not copy and paste but directly fetches the data in a precise and accurate manner. It does not limit itself to the web; data can be scraped virtually from anywhere it is stored. It does not have to be from the Internet. It is about data and not where it is stored.

Example of Web Scraping

* Web scraping would involve scraping specific information from a particular web page or pages.
* For example, you want to work on price intelligence. You would extract the price of various/specific products from Amazon or any other e-commerce site.
* This would qualify as web scraping. Likewise, you can extract data and use it for business leads, stock market data, real estate listings.

1. What is ‘robots.txt’ file? What is the use of ‘robots.txt’ file?

Answer: Robots.txt is a text file webmasters create to instruct web robots (typically search engine robots) how to crawl pages on their website. The robots.txt file is part of the the robots exclusion protocol (REP), a group of web standards that regulate how robots crawl the web, access and index content, and serve that content up to users. The REP also includes directives like [meta robots](https://moz.com/learn/seo/robots-meta-directives), as well as page-, subdirectory-, or site-wide instructions for how search engines should treat links (such as “follow” or “nofollow”).

In practice, [robots.txt files indicate](https://moz.com/blog/interactive-guide-to-robots-txt) whether certain user agents (web-crawling software) can or cannot crawl parts of a website. These crawl instructions are specified by “disallowing” or “allowing” the behavior of certain (or all) user agents.

##### **format:**

User-agent: [user-agent name]Disallow: [URL string not to be crawled]

The robots. txt file, also known as the robots exclusion protocol or standard, is a text file that tells web robots (most often search engines) which pages on your site to crawl. It also tells web robots which pages not to crawl.

For web pages (HTML, PDF, or other [non-media formats that Google can read](https://support.google.com/webmasters/answer/35287)), robots.txt can be used to manage crawling traffic if you think your server will be overwhelmed by requests from Google's crawler, or to avoid crawling unimportant or similar pages on your site.

You should not use robots.txt as a means to hide your web pages from Google Search results. This is because, if other pages point to your page with descriptive text, your page could still be indexed without visiting the page. If you want to block your page from search results, use another method such as password protection or a [noindex](https://developers.google.com/webmasters/control-crawl-index/docs/robots_meta_tag) directive.

1. What are static and dynamic web pages?

Answer: For static web pages data do not changes until someone changes it manually and hence data is static in nature. On other hand for Dynamic web page data is first interoperate at server side and due to which it does not remain same on every call and this makes data dynamic in nature.

For example Static Web Pages, Facebook is a website and a web application. However, a business's simple website is not a web application. A static site is the most basic kind of website, and the easiest to create. It requires no server-side (also called back-end) processing, only client-side and Netflix.

Examples of Dynamic Websites are writing blogs, e-commerce sites, calendar, to-do sites and other types of sites which needs updating frequently.

Q14 and Q15 are programming practice questions. Solve it using JUPYTER NOTEBOOK and paste the solution in your answer sheets.

1. Write a python program to check whether a webpage contains a title or not.

Program:

import requests

import csv

from bs4 import BeautifulSoup

page = requests.get('https://web.archive.org/web/20121007172955/http://www.nga.gov/collection/anZ1.htm')

soup = BeautifulSoup(page.text, 'html.parser')

last\_links = soup.find(class\_='AlphaNav')

last\_links.decompose()

# Create a file to write to, add headers row

f = csv.writer(open('z-artist-names.csv', 'w'))

f.writerow(['Name', 'Link'])

artist\_name\_list = soup.find(class\_='BodyText')

artist\_name\_list\_items = artist\_name\_list.find\_all('a')

for artist\_name in artist\_name\_list\_items:

names = artist\_name.contents[0]

links = 'https://web.archive.org' + artist\_name.get('href')

# Add each artist’s name and associated link to a row

f.writerow([names, links])

1. Write a python program to access the search bar and search button on images.google.com.

Program:

pip install beautifulsoup4

pip install google

search(query, tld='com', lang='en', num=10, start=0, stop=None, pause=2.0)

try:

from googlesearch import search

except ImportError:

print("No module named 'google' found")

# to search

query = "Geeksforgeeks"

for j in search(query, tld="co.in", num=10, stop=10, pause=2):

print(j)