Web Scraping using Python

- Web Scraping (also termed Screen Scraping, Web Data Extraction, Web Harvesting etc.) is a technique employed to extract large amounts of data from websites whereby the data is extracted and saved to a local file in our computer or to a database in table (spreadsheet) format.
- Data displayed by most websites can only be viewed using a web browser.
- They do not offer the functionality to save a copy of this data for personal use.
- The only option then is to manually copy and paste the data - a very tedious job which can take many hours or sometimes days to complete.
- Web Scraping is the technique of automating this process, so that instead of manually copying the data from websites, the Web Scraping software will perform the same task within a fraction of the time.
- A web scraping software will automatically load and extract data from multiple pages of websites based on our requirement.
- It is either custom built for a specific website or is one which can be configured to work with any website.
- With the click of a button we can easily save the data available in the website to a file in our computer.
- Web scraping is a term used to describe the use of a program or algorithm to extract and process large amounts of data from the web.
- We can perform Web scraping using Python in better way because of the libraries which are provided by Python to perform Web scrapping.
- Whether we are a data scientist, engineer, or anybody who analyzes large amounts of datasets, the ability to scrape data from the web is a useful skill to have.
- If we find data from the web, and there is no direct way to download it, web scraping using Python is a skill by using which we can use to extract the data into a useful form that can be imported.
- We can perform Web scraping using Python in better way because of the libraries which are provided by Python to perform Web scrapping.

Libraries required for web scraping using Python

As we know, Python is an open source programming language.
We may find many libraries to perform one function.
Hence, it is necessary to find the best to use library.
We prefer BeautifulSoup (Python library), since it is easy and intuitive to work on.

Precisely, we will use two Python modules for scraping data:

Urllib2:

- It is a Python module which can be used for fetching URLs.
- It defines functions and classes to help with URL actions (basic and digest authentication, redirections, cookies, etc).
- urllib2 is the name of the library included in Python 2.
- We can use the urllib.request library included with Python 3, instead.
- The urllib.request library works the same way urllib.request works in Python 2. Because it is already included we don't need to install it.

BeautifulSoup:

- It is an incredible tool for pulling out information from a webpage.
- We can use it to extract tables, lists, paragraph and we can also put filters to extract information from web pages.
- Beautiful Soup is a Python package for parsing HTML and XML documents (including having malformed markup, i.e. non-closed tags, so named after tag soup).
- It creates a parse tree for parsed pages that can be used to extract data from HTML, which is useful for web scraping.
- It is available for Python 2.7 and Python 3.
- BeautifulSoup does not fetch the web page for us. That's why, we use urllib2 in combination with the BeautifulSoup library.

Python has several other options for HTML scraping in addition to BeatifulSoup. as:

- mechanize
- scrapemark
- scrapy

Installing Beautiful Soup library

To install Beautiful Soup library we can use PIP utility as

```
pip install beautifulsoup4
```

After installing the BeautifulSoup on our machine we can import the module as

```
import bs4
```