**LinkedIn Job Listing Analysis**

**Initial Setup:**

* Install packages – selenium, beautiful soup, requests
* Import the libraries –
  + - from selenium import WebDriver
    - from selenium.webdriver.chrome.service import Service
    - Set the chrome driver –

chromedriver\_path = 'D:/chromedriver-win64/chromedriver.exe'

service = Service(chromedriver\_path)

driver = webdriver.Chrome(service=service)

* Setup a config file with user details and others
* Load the config file in the setup up file using the load\_dotenv("config.env")
  + - Get the username, password, etc using var = os.getenv(“variablename”)
* Load the base URL using driver.get () and send login data using var.send\_keys(creds)

**Load:**

* To load the data, navigate to the page (job search); use wait if necessary.

Example of wait: job\_icon = WebDriverWait(driver, 30).until(EC.presence\_of\_element\_located((By.XPATH, "//li-icon[@type='job']")))

* Initialise beautiful soup by soup = BeautifulSoup(driver.page\_source, “html.parser”)

It takes two parameters the page source and a parser.

* Get all elements using all = soup.find\_all(“tag”, class\_=”classname”)
* Run a for loop and extract each job titile, location, company name and url.

Sample –

job\_list = []

for job in all\_jobs:

 title = job.find("a", class\_=re.compile('job-card-list\_\_title-- link')).find("strong").get\_text(strip=True)

           job\_list.append([title,company,location,job\_link])