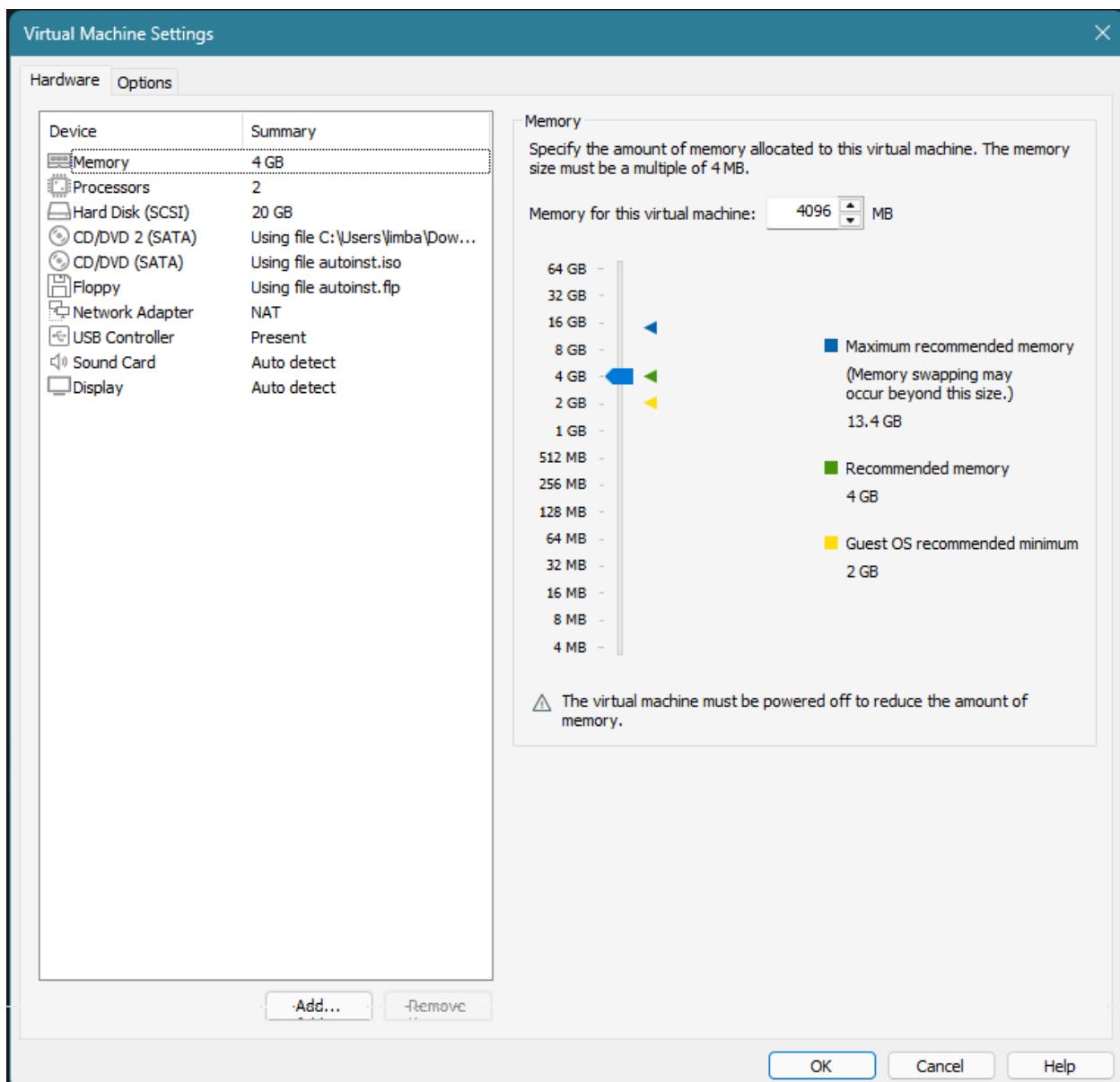


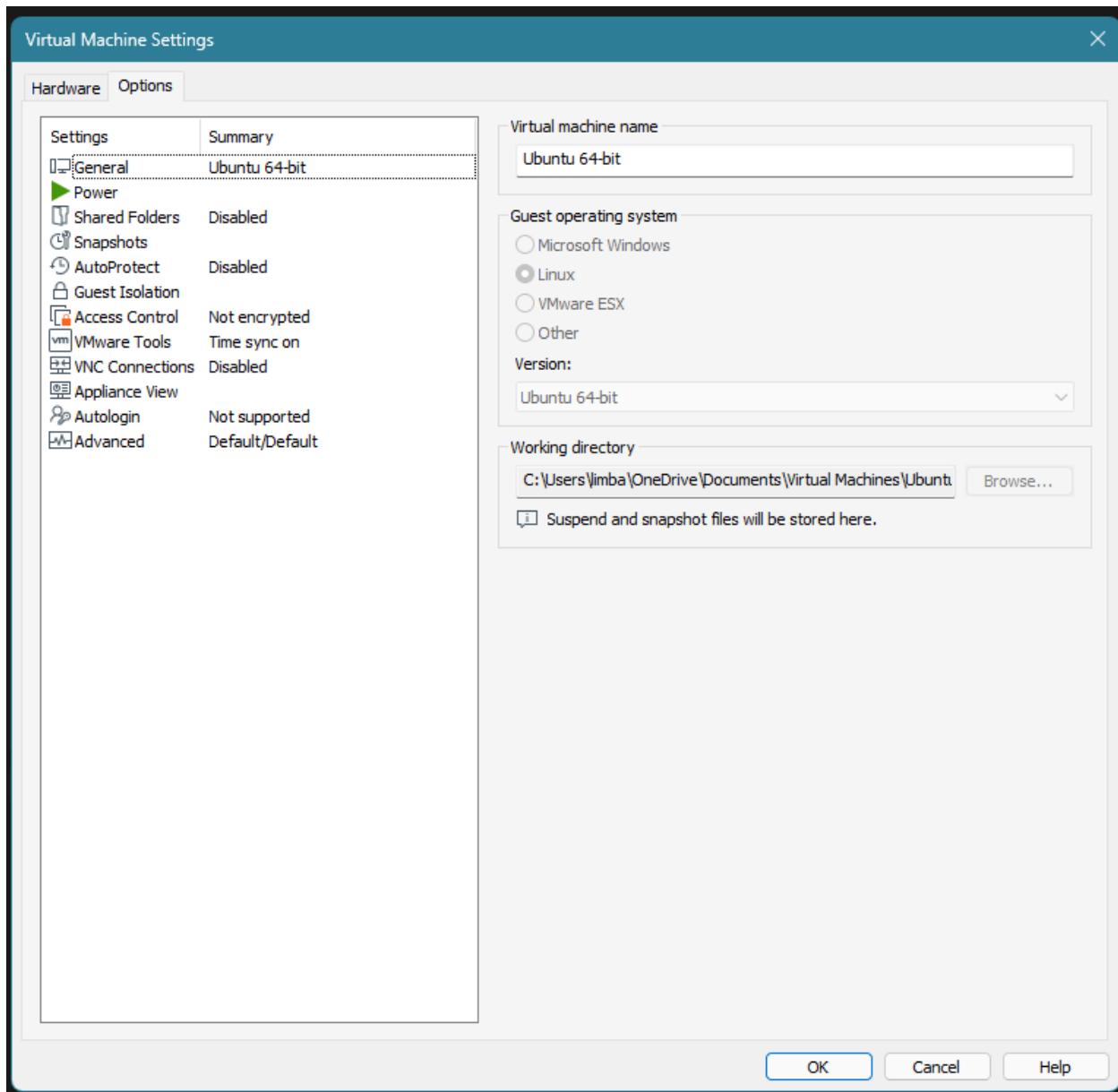
LAB 1 REPORT

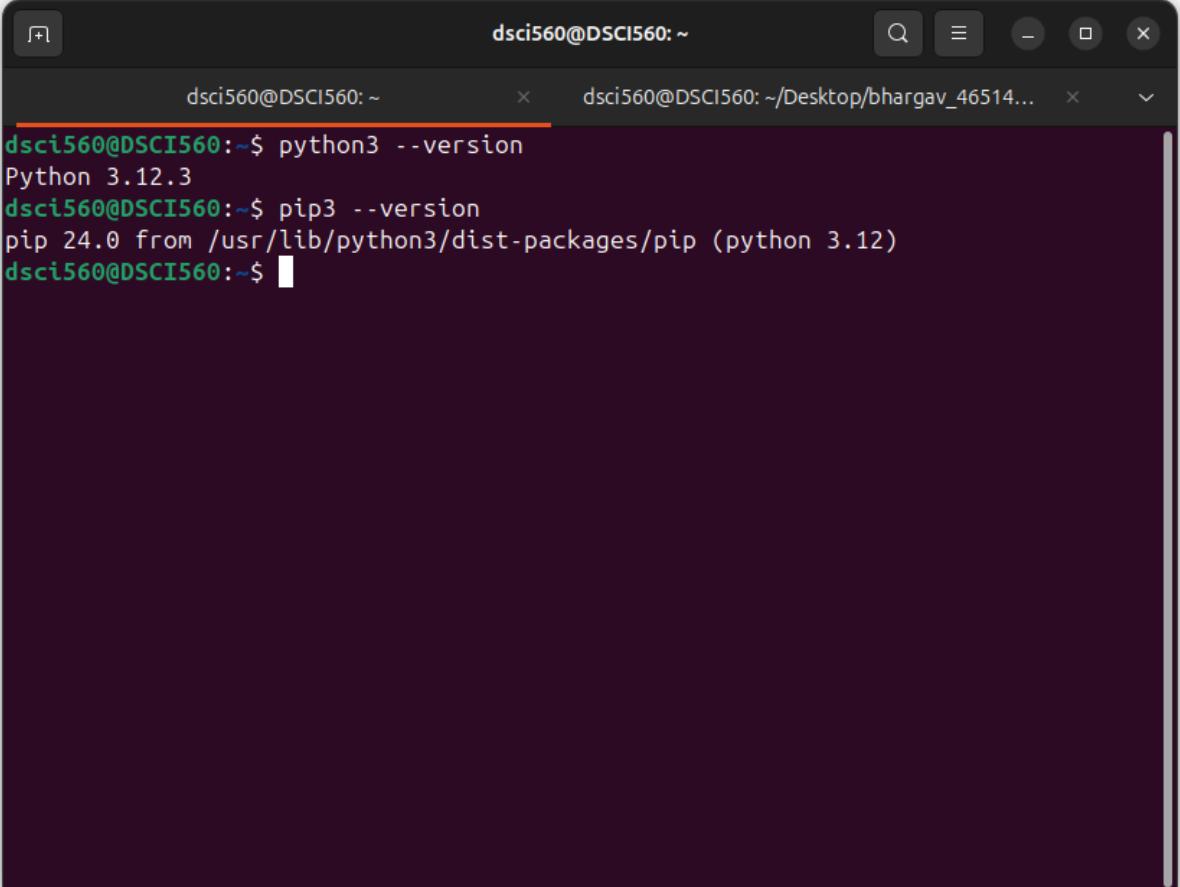
DSCI 560 | Bhargav Limbasia | 4651477356 | [Github Repo](#)

Setting up VM and Install Python on Linux

- Ubuntu Desktop ISO was downloaded from ubuntu.com and installed via VMware installation wizard







The screenshot shows a terminal window with two tabs. The left tab is titled 'dsci560@DSCI560: ~' and the right tab is titled 'dsci560@DSCI560: ~/Desktop/bhargav_46514...'. The terminal window has a dark background and light-colored text. It displays the following command and output:

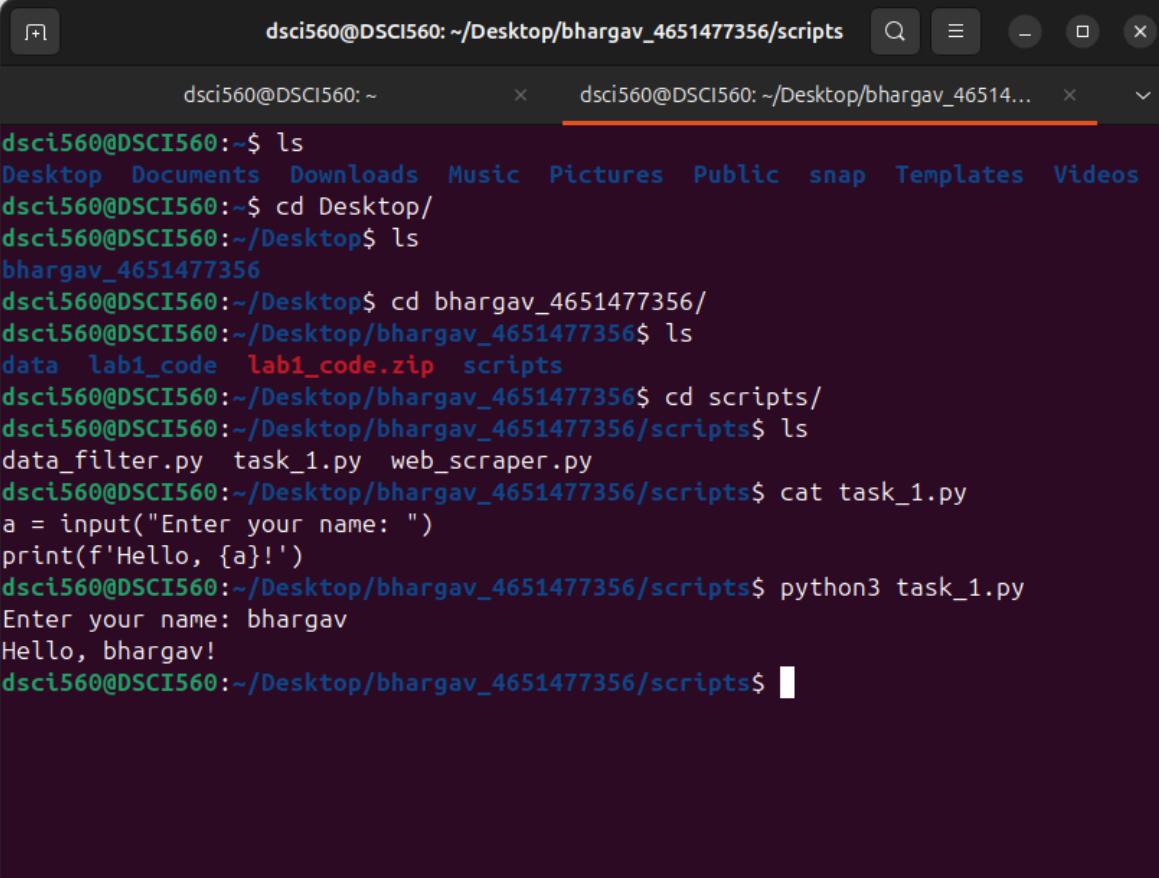
```
dsci560@DSCI560:~$ python3 --version
Python 3.12.3
dsci560@DSCI560:~$ pip3 --version
pip 24.0 from /usr/lib/python3/dist-packages/pip (python 3.12)
dsci560@DSCI560:~$
```

Packages to install before executing the scripts:

- # Update package list
sudo apt update
- # Install Python3 and pip
sudo apt install -y python3 python3-pip
- # Install Chrome/Chromium
sudo apt install -y chromium-browser
- # Install all Python packages via pip
pip install selenium beautifulsoup4 lxml webdriver-manager
--break-system-packages

A basic Python Script

- The screenshot below shows the folders and their locations, as mentioned in the lab.
- Created the task_1.py using the command ‘nano task_1.py’
- Used a simple input and a formatted print statement to complete the task.

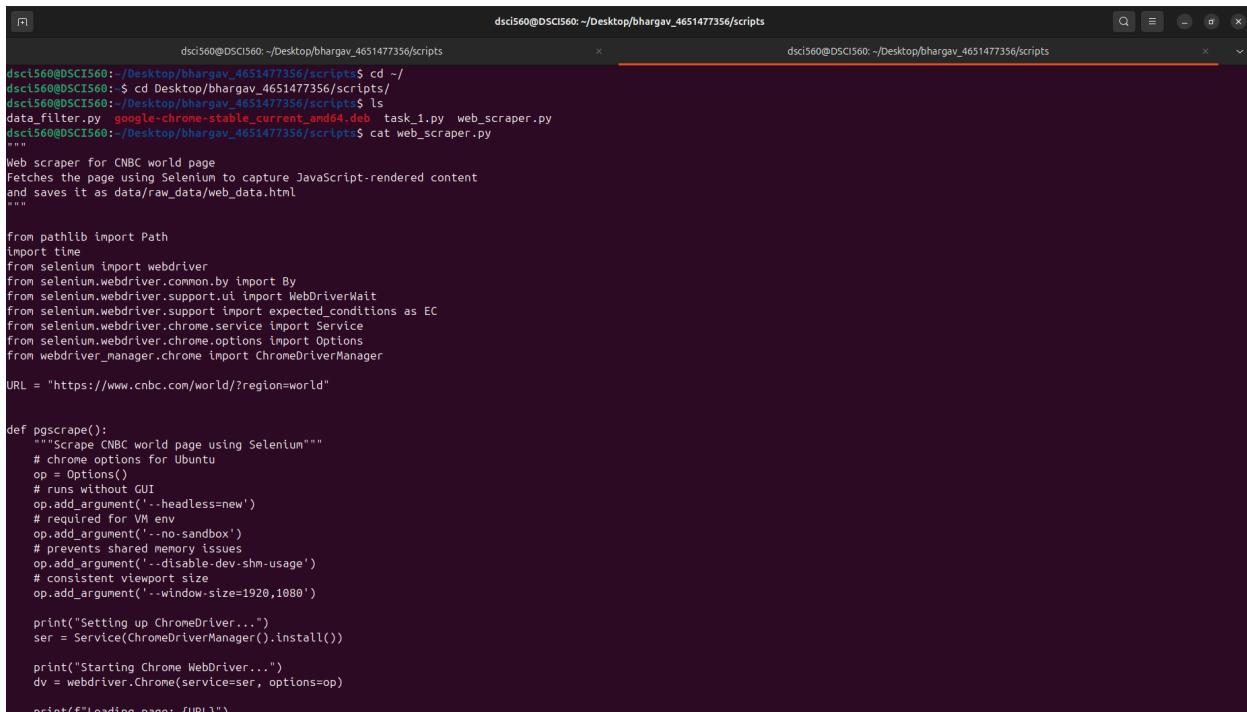


The screenshot shows a terminal window with two tabs. The left tab shows the directory structure of the user's desktop, including Desktop, Documents, Downloads, Music, Pictures, Public, snap, Templates, and Videos. The right tab shows the user navigating to the 'scripts' directory within their desktop folder and executing the 'task_1.py' script. The terminal output shows the script being created with nano, its contents (a simple input and print statement), and its execution, which prints 'Hello, bhargav!' back to the user.

```
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts
dsci560@DSCI560:~$ ls
Desktop Documents Downloads Music Pictures Public snap Templates Videos
dsci560@DSCI560:~$ cd Desktop/
dsci560@DSCI560:~/Desktop$ ls
bhargav_4651477356
dsci560@DSCI560:~/Desktop$ cd bhargav_4651477356/
dsci560@DSCI560:~/Desktop/bhargav_4651477356$ ls
data lab1_code lab1_code.zip scripts
dsci560@DSCI560:~/Desktop/bhargav_4651477356$ cd scripts/
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ ls
data_filter.py task_1.py web_scraper.py
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ cat task_1.py
a = input("Enter your name: ")
print(f'Hello, {a}!')
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ python3 task_1.py
Enter your name: bhargav
Hello, bhargav!
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$
```

Python Web-scraping Task

- Uses Selenium WebDriver with Chrome to fetch JavaScript-rendered content from the CNBC world page
- Configures Chrome to run in headless mode (no GUI) with Ubuntu-optimized arguments
- Automatically downloads and manages ChromeDriver using the webdriver-manager library
- Waits for page elements to load using WebDriverWait to ensure content is fully rendered
- Adds a 3-second sleep after initial load to allow dynamic content to populate
- Extracts the complete HTML source code after JavaScript execution
- Saves raw HTML to data/raw_data/web_data.html for subsequent processing
- Creates output directory structure automatically if it doesn't exist
- Closes browser session after successful scraping to free resources



The screenshot shows a terminal window with two tabs. The left tab displays the command-line history and the right tab shows the output of the script execution. The script itself is a Python file named 'web_scraper.py' which uses Selenium to scrape the CNBC world page.

```
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts$ cd ~/  
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts$ ls  
data_filter.py google-chrome-stable_current_and64.deb task_1.py web_scraper.py  
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts$ cat web_scraper.py  
...  
Web scraper for CNBC world page  
Fetches the page using Selenium to capture JavaScript-rendered content  
and saves it as data/raw_data/web_data.html  
  
from pathlib import Path  
import time  
from selenium import webdriver  
from selenium.webdriver.common.by import By  
from selenium.webdriver.support.ui import WebDriverWait  
from selenium.webdriver.support import expected_conditions as EC  
from selenium.webdriver.chrome.service import Service  
from selenium.webdriver.chrome.options import Options  
from webdriver_manager.chrome import ChromeDriverManager  
  
URL = "https://www.cnbc.com/world/?region=world"  
  
def pgscrape():  
    """Scrape CNBC world page using Selenium"""\n    # chrome options for Ubuntu\n    op = Options()\n    # runs without GUI\n    op.add_argument('--headless=new')\n    # required for VM env\n    op.add_argument('--no-sandbox')\n    # prevents shared memory issues\n    op.add_argument('--disable-dev-shm-usage')\n    # consistent viewport size\n    op.add_argument('--window-size=1920,1080')\n  
    print("Setting up ChromeDriver...")\n    ser = Service(ChromeDriverManager().install())\n  
    print("Starting Chrome WebDriver...")\n    dv = webdriver.Chrome(service=ser, options=op)  
  
    print(f"Loading page: {URL}")
```

```
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts
```

```
def pgscrape():
    """Scrape CNBC world page using Selenium"""
    # chrome options for Ubuntu
    op = Options()
    # runs without GUI
    op.add_argument('--headless=new')
    # required for VM env
    op.add_argument('--no-sandbox')
    # prevents shared memory issues
    op.add_argument('--disable-dev-shm-usage')
    # consistent viewport size
    op.add_argument('--window-size=1920,1080')

    print("Setting up ChromeDriver...")
    ser = Service(ChromeDriverManager().install())

    print("Starting Chrome WebDriver...")
    dv = webdriver.Chrome(service=ser, options=op)

    print(f"Loading page: {URL}")
    dv.get(URL)

    # waiting for page to load
    wait = WebDriverWait(dv, 15)
    wait.until(EC.presence_of_element_located((By.TAG_NAME, "a")))
    time.sleep(3)

    # getting page source
    htmlCont = dv.page_source

    # quit browser
    dv.quit()

    return htmlCont

def savehtml(htmlCont):
    #Save HTML content to file
    outDir = Path(__file__).parent.parent / 'data' / 'raw_data'
    outDir.mkdir(parents=True, exist_ok=True)
    outFile = outDir / 'web_data.html'

    outFile.write_text(htmlCont, encoding='utf-8')
    print(f"Saved to: {outFile}")
```

```
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts
```

```
# waiting for page to load
wait = WebDriverWait(dv, 15)
wait.until(EC.presence_of_element_located((By.TAG_NAME, "a")))
time.sleep(3)

# getting page source
htmlCont = dv.page_source

# quit browser
dv.quit()

return htmlCont

def savehtml(htmlCont):
    #Save HTML content to file
    outDir = Path(__file__).parent.parent / 'data' / 'raw_data'
    outDir.mkdir(parents=True, exist_ok=True)
    outFile = outDir / 'web_data.html'

    outFile.write_text(htmlCont, encoding='utf-8')
    print(f"Saved to: {outFile}")

def main():
    html = pgscrape()
    savehtml(html)
    print("Task 2.3 - Scraping completed successfully!")

if __name__ == '__main__':
    main()
```

```
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ python3 web_scraper.py
Setting up ChromeDriver...
Starting Chrome WebDriver...
Loading page: https://www.cnbc.com/world/?region=world
Saved to: /home/dsci560/Desktop/bhargav_4651477356/data/raw_data/web_data.html
Task 2.3 - Scraping completed successfully!
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$
```

```

dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts$ cd ..data/raw_data/
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts$ ls
web_data.html
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/data/raw_data$ head -n 2 web_data.html
<html lang="en" prefix="og:https://opengraphprotocol.org/ns#">
  <head><script async="" type="text/javascript" id="cxJsonpCBscript" src="https://id.c
  ... (large block of JSON data)

```

Data Filtering Task

- Parses the saved HTML file using BeautifulSoup for structured data extraction
- Targets specific CSS classes to locate market data cards (MarketCard-container)
- Extracts three fields from each market card: symbol, stock position, and change percentage
- Searches for news items using the LatestNews-item class identifiers
- Extracts timestamp, title, and link for each news article
- Converts relative URLs to absolute URLs by prepending the CNBC base domain
- Writes market data to data/processed_data/market_data.csv with three columns
- Writes news data to data/processed_data/news_data.csv with three columns
- Uses CSV DictWriter for clean, structured output with proper headers
- Returns empty rows if no data is found to maintain CSV structure consistency

```

dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ cd Desktop/bhargav_4651477356/scripts/
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ ls
data_filter.py  google-chrome-stable_current_and64.deb  task_1.py  web_scraper.py
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ cat data_filter.py
"""

parse data/raw_data/web_data.html and produce two CSVs in data/processed_data:
market_data.csv with columns: marketCard_symbol, marketCard_stockPosition, marketCard-changePct
news_data.csv with columns: LatestNews-timestamp, title, link
"""

from pathlib import Path
from bs4 import BeautifulSoup
import csv
import re

def safetext(el):
    # extract text from element safely
    if el:
        return el.get_text(separator=' ', strip=True)
    return ''

def extractmarket(s):
    # extract market data from MarketCard elements
    print("Filtering market data fields...")
    mrows = []

    # find all MarketCard containers
    cards = s.find_all('a', class_=lambda x: x and 'MarketCard-container' in x)

    for card in cards:
        # extract symbol
        syelem = card.find('span', class_='MarketCard-symbol')
        sym = safetext(syelem)

        # extract stock position (price)
        poselem = card.find('span', class_='MarketCard-stockPosition')
        pos = safetext(poselem)

        # extract change percentage
        pctelem = card.find('span', class_='MarketCard-changesPct')
        cngpct = safetext(pctelem)

        # only add if we have at least a symbol
        if sym:
            mrows.append({
                'marketCard_symbol': sym,
                'marketCard_stockPosition': pos,
                'marketCard-changePct': cngpct
            })

    # if no data found, return empty row - fallback
    if not mrows:
        mrows = [{"marketCard_symbol": '', 'marketCard_stockPosition': '', 'marketCard-changePct': ''}]

    return mrows

def extractlatestnews(s):
    # find Latest News section and extract timestamp, title, link for each news item
    print("Filtering news data fields...")
    news = []

    # find all LatestNews-item elements
    newsitems = s.find_all('li', class_=lambda x: x and 'LatestNews-item' in x)

```

```

dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ dsci560@DSCI560:~/Desktop/bhargav_4651477356/data/raw_data$ dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$
def extractmarket(s):
    # extract market data from MarketCard elements
    print("Filtering market data fields...")
    mrows = []

    # find all MarketCard containers
    cards = s.find_all('a', class_=lambda x: x and 'MarketCard-container' in x)

    for card in cards:
        # extract symbol
        syelem = card.find('span', class_='MarketCard-symbol')
        sym = safetext(syelem)

        # extract stock position (price)
        poselem = card.find('span', class_='MarketCard-stockPosition')
        pos = safetext(poselem)

        # extract change percentage
        pctelem = card.find('span', class_='MarketCard-changesPct')
        cngpct = safetext(pctelem)

        # only add if we have at least a symbol
        if sym:
            mrows.append({
                'marketCard_symbol': sym,
                'marketCard_stockPosition': pos,
                'marketCard-changePct': cngpct
            })

    # if no data found, return empty row - fallback
    if not mrows:
        mrows = [{"marketCard_symbol": '', 'marketCard_stockPosition': '', 'marketCard-changePct': ''}]

    return mrows

def extractlatestnews(s):
    # find Latest News section and extract timestamp, title, link for each news item
    print("Filtering news data fields...")
    news = []

    # find all LatestNews-item elements
    newsitems = s.find_all('li', class_=lambda x: x and 'LatestNews-item' in x)

```

```

dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/data/raw_data
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts

def extractlatestnews(s):
    # find Latest News section and extract timestamp, title, link for each news item
    print("Filtering news data fields...")
    news = []

    # find all LatestNews-item elements
    newsitems = s.find_all('li', class_=lambda x: x and 'LatestNews-item' in x)

    for item in newsitems:
        # extract timestamp
        t = item.find('time', class_='LatestNews-timestamp')
        timestamp = safetext(t)

        # extract headline link
        headelem = item.find('a', class_='LatestNews-headline')
        if headelem:
            title = safetext(headelem)
            href = headelem.get('href', '')

        # make sure the link is absolute
        if href and not href.startswith('http'):
            href = 'https://www.cnbc.com' + href

        if title and href:
            news.append({
                'LatestNews.timestamp': timestamp,
                'title': title,
                'link': href
            })
    return news

def writecsv(path, fieldnames, rows):
    # write data to CSV file
    path.parent.mkdir(parents=True, exist_ok=True)
    with path.open('w', newline='', encoding='utf-8') as fh:
        writer = csv.DictWriter(fh, fieldnames=fieldnames)
        writer.writeheader()
        for r in rows:
            writer.writerow(r)

```

```

dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/data/raw_data
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts

def main():
    # set up paths
    base = Path(__file__).resolve().parents[1]
    rawfile = base / 'data' / 'raw_data' / 'web_data.html'
    procdir = base / 'data' / 'processed_data'
    procdir.mkdir(parents=True, exist_ok=True)
    marketcsv = procdir / 'market_data.csv'
    newscsv = procdir / 'news_data.csv'

    # read HTML file
    html = rawfile.read_text(encoding='utf-8')
    s = BeautifulSoup(html, 'html.parser')

    # extract data
    print("\nStarting data extraction...")
    marketrows = extractmarket(s)
    print(f"Storing market data ({len(marketrows)} rows...")

    news = extractlatestnews(s)
    print(f"Storing news data ({len(news)} rows...")

    # write CSVs
    writecsv(marketcsv, ['marketCard_symbol', 'marketCard_stockPosition', 'marketCard-changePct'], marketrows)
    print(f"Market CSV created: {marketcsv}")

    writecsv(newscsv, ['LatestNews-timestamp', 'title', 'link'], news)
    print(f"News CSV created: {newscsv}")

    print("\nData filtering complete!")

if __name__ == '__main__':
    main()
dsci560@DSCI560: ~/Desktop/bhargav_4651477356/scripts$ python3 data_filter.py

Starting data extraction...
Filtering market data fields...
Storing market data (5 rows)...
Filtering news data fields...
Storing news data (30 rows)...
Market CSV created: /home/dsci560/Desktop/bhargav_4651477356/data/processed_data/market_data.csv
News CSV created: /home/dsci560/Desktop/bhargav_4651477356/data/processed_data/news_data.csv

```

```

dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ python3 data_filter.py
print("\nData filtering complete!")

if __name__ == '__main__':
    main()
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ python3 data_filter.py

Starting data extraction...
Filtering market data fields...
Storing market data (5 rows)...
Filtering news data fields...
Storing news data (30 rows)...
Market CSV created: /home/dsci560/Desktop/bhargav_4651477356/data/processed_data/market_data.csv
News CSV created: /home/dsci560/Desktop/bhargav_4651477356/data/processed_data/news_data.csv

Data filtering complete!
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ cd ..data/processed_data/
dsci560@DSCI560:~/Desktop/bhargav_4651477356/data/processed_data$ ls
market_data.csv news_data.csv
dsci560@DSCI560:~/Desktop/bhargav_4651477356/data/processed_data$ cat market_data.csv
marketCard_symbol,marketCard_stockPosition,marketCard-changePct
STOX600*,614.38,-0.03%
DAX*,25,297.13,-0.22%
FTSE*,10,235.29,-0.04%
CAC*,8,258.94,-0.65%
FTSE MIB*,45,799.69,-0.11%
dsci560@DSCI560:~/Desktop/bhargav_4651477356/data/processed_data$ cat news_data.csv
LatestNews-timestamp,title,link
11 Hours Ago,Week in review: Stocks battled a flood of news and we booked some profits,https://www.cnbc.com/2026/01/17/week-in-review-stocks-battled-a-flood-of-news-and-we-booked-some-profits.html
11 Hours Ago,Trump threatens to sue JPMorgan Chase for 'debanking' him,https://www.cnbc.com/2026/01/17/trump-jpmorgan-chase-debanking.html
13 Hours Ago,Trump: NATO members to face tariffs up to 25% until a Greenland deal is struck,https://www.cnbc.com/2026/01/17/trump-greenland-tariffs-nato.html
14 Hours Ago,"Led by Texas, states race to prove they can put bitcoin on public balance sheet",https://www.cnbc.com/2026/01/17/texas-us-states-budgets-bitcoin-crypto-strategic-reserve.html
16 Hours Ago,Unshaken: Why Brazilian stocks have looked past the Venezuela attack,https://www.cnbc.com/2026/01/17/unshaken-why-brazilian-stocks-have-looked-past-the-venezuela-attack.html
16 Hours Ago,Bestselling author: How to create better habits without relying on discipline,https://www.cnbc.com/2026/01/17/james-clear-how-to-create-better-habits-without-relying-on-discipline.html
16 Hours Ago,Warren Buffett: To maximize your potential, ask yourself this question",https://www.cnbc.com/2026/01/17/warren-buffett-to-maximize-your-potential-ask-yourself-this-question.html
16 Hours Ago,"Buy these five stocks ahead of earnings, Bank of America says",https://www.cnbc.com/2026/01/17/stocks-to-buy-ahead-of-earnings-bank-of-america-says.html
16 Hours Ago,This week's most overbought names include Darden Restaurants and Target,https://www.cnbc.com/2026/01/17/this-weeks-most-overbought-names-include-darden-restaurants-and-target.html

```

```

dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts$ cat news_data.csv
LatestNews-timestamp,title,link
11 Hours Ago,Week in review: Stocks battled a flood of news and we booked some profits,https://www.cnbc.com/2026/01/17/week-in-review-stocks-battled-a-flood-of-news-and-we-booked-some-profits.html
11 Hours Ago,Trump threatens to sue JPMorgan Chase for 'debanking' him,https://www.cnbc.com/2026/01/17/trump-jpmorgan-chase-debanking.html
13 Hours Ago,Trump: NATO members to face tariffs up to 25% until a Greenland deal is struck,https://www.cnbc.com/2026/01/17/trump-greenland-tariffs-nato.html
14 Hours Ago,"Led by Texas, states race to prove they can put bitcoin on public balance sheet",https://www.cnbc.com/2026/01/17/texas-us-states-budgets-bitcoin-crypto-strategic-reserve.html
16 Hours Ago,Unshaken: Why Brazilian stocks have looked past the Venezuela attack,https://www.cnbc.com/2026/01/17/unshaken-why-brazilian-stocks-have-looked-past-the-venezuela-attack.html
16 Hours Ago,Bestselling author: How to create better habits without relying on discipline,https://www.cnbc.com/2026/01/17/james-clear-how-to-create-better-habits-without-relying-on-discipline.html
16 Hours Ago,Warren Buffett: To maximize your potential, ask yourself this question",https://www.cnbc.com/2026/01/17/warren-buffett-to-maximize-your-potential-ask-yourself-this-question.html
16 Hours Ago,"Buy these five stocks ahead of earnings, Bank of America says",https://www.cnbc.com/2026/01/17/stocks-to-buy-ahead-of-earnings-bank-of-america-says.html
16 Hours Ago,This week's most overbought names include Darden Restaurants and Target,https://www.cnbc.com/2026/01/17/this-weeks-most-overbought-names-include-darden-restaurants-and-target.html
16 Hours Ago,"Buffett on parenting, giving up horse betting and why he stopped talking politics",https://www.cnbc.com/2026/01/17/warren-buffett-on-parenting-horse-betting-and-why-he-stopped-talking-politics.html
16 Hours Ago,"Unexpected expenses take 10% of retirees' income, on average, research shows",https://www.cnbc.com/2026/01/17/retirees-emergency-savings.html
17 Hours Ago,Disney dominated the 2025 box office. Here's how it could keep the crown in 2026,https://www.cnbc.com/2026/01/17/disney-dominated-2025-box-office.html
20 Hours Ago,The founders of billion-dollar AI startups are getting younger – here's why,https://www.cnbc.com/2026/01/17/billion-dollar-ai-startup-founders-are-getting-younger-heres-why.html
"January 16, 2026",Elon Musk's xAI faces tougher road building data centers after EPA rule update,https://www.cnbc.com/2026/01/16/musks-xai-faces-tougher-road-expanding-memphis-area-arter-epa-update.html
"January 16, 2026",Here's why Jim Cramer thinks chip stocks can go higher,https://www.cnbc.com/2026/01/16/heres-why-jim-cramer-thinks-chip-stocks-can-go-higher.html
"January 16, 2026",Cramer's Lightning Round: Sell Super Micro Computer,https://www.cnbc.com/2026/01/16/cramers-lightning-round-sell-super-micro-computer.html
"January 16, 2026",Cramer's week ahead: Earnings from Netflix, Intel, Capital One, McCormick ,https://www.cnbc.com/2026/01/16/cramers-week-ahead-earnings-from-netflix-intel-capital-one-mccormick.html
"January 16, 2026",Google files to appeal search monopoly case,https://www.cnbc.com/2026/01/16/google-files-to-appeal-search-monopoly-case.html
"January 16, 2026","More employers worry about workers' financial well-being, research shows",https://www.cnbc.com/2026/01/16/employers-focusing-more-on-employee-financial-wellbeing-study-shows.html
"January 16, 2026",Republicans want to end the 'marriage penalty' for this childcare tax credit,https://www.cnbc.com/2026/01/16/child-and-dependent-care-tax-credit.html
"January 16, 2026",Labor Department accused of echoing Nazi slogan in social media post,https://www.cnbc.com/2026/01/16/trump-labor-naazi-slogan-social-media.html
"January 16, 2026",Education Department to delay collections on defaulted student loans,https://www.cnbc.com/2026/01/16/student-loan-collections-paused.html
"January 16, 2026","Earnings reports, fears around interest rate outlook may sway markets next week",https://www.cnbc.com/2026/01/16/stock-market-next-week-outlook-for-jan-19-23-2026.html
"January 16, 2026",OpenAI has committed billions to recent chip deals. Some big names have been left out,https://www.cnbc.com/2026/01/16/openai-chip-deal-with-cerebras-adds-to-roster-of-nvidia-and-broadcom.html
"January 16, 2026",One of our top stocks this week just lost its CFO – what it means for investors,https://www.cnbc.com/2026/01/16/one-of-our-top-stocks-this-week-just-lost-its-cfo-what-it-means.html
"January 16, 2026",Hassett pivots to possible 'Trump cards' amid credit card battle with banks,https://www.cnbc.com/2026/01/16/white-house-hassett-trump-cards-credit-card-battle.html
"January 16, 2026",Can Home Depot's AI-powered push to court pros move the needle in its own business?,https://www.cnbc.com/2026/01/16/can-home-depots-ai-powered-push-to-court-pros-move-the-needle-in-its-own-business.html

```

```

dsci560@DSCI560:~/Desktop/bhargav_4651477356/data/processed_data
dsci560@DSCI560:~/Desktop/bhargav_4651477356/scripts          dsci560@DSCI560:~/Desktop/bhargav_4651477356/data/raw_data      dsci560@DSCI560:~/Desktop/bhargav_4651477356/data/processed_data
html
16 Hours Ago,Unshaken: Why Brazilian stocks have looked past the Venezuela attack,https://www.cnbc.com/2026/01/17/unshaken-why-brazilian-stocks-have-looked-past-the-venezuela-attack.html
16 Hours Ago,Bestselling author: How to create better habits without relying on discipline,https://www.cnbc.com/2026/01/17/james-clear-how-to-create-better-habits-without-relying-on-discipline.html
16 Hours Ago,"Warren Buffett: To maximize your potential, ask yourself this question",https://www.cnbc.com/2026/01/17/warren-buffett-to-maximize-your-potential-ask-yourself-this-question.html
16 Hours Ago,"Buy these five stocks ahead of earnings, Bank of America says",https://www.cnbc.com/2026/01/17/stocks-to-buy-ahead-of-earnings-bank-of-america-says.html
16 Hours Ago,This week's most overbought names include Darden Restaurants and Target,https://www.cnbc.com/2026/01/17/this-weeks-most-overbought-names-include-darden-restaurants-and-target.html
16 Hours Ago,"Buffett on parenting, giving up horse betting and why he stopped talking politics",https://www.cnbc.com/2026/01/17/warren-buffett-on-parenting-horse-betting-and-why-he-stopped-talking-politics.html
16 Hours Ago,"Unexpected expenses take 10% of retirees' income, on average, research shows",https://www.cnbc.com/2026/01/17/retirees-emergency-savings.html
17 Hours Ago,Disney dominated the 2025 box office. Here's how it could keep the crown in 2026,https://www.cnbc.com/2026/01/17/disney-dominated-2025-box-office.html
20 Hours Ago,The founders of billion-dollar AI startups are getting younger – here's why,https://www.cnbc.com/2026/01/17/billion-dollar-ai-startup-founders-are-getting-younger-heres-why.html
"January 16, 2026",Elon Musk's xAI faces tougher road building data centers after EPA rule update,https://www.cnbc.com/2026/01/16/musk-s-xai-faces-tougher-road-expanding-memphis-area-after-epa-update.html
"January 16, 2026",Here's why Jim Cramer thinks chip stocks can go higher,https://www.cnbc.com/2026/01/16/heres-why-jim-cramer-thinks-chip-stocks-can-go-higher.html
"January 16, 2026",Cramer's Lightning Round: Sell Super Micro Computer,https://www.cnbc.com/2026/01/16/cramers-lightning-round-sell-super-micro-computer.html
"January 16, 2026",Cramer's week ahead: Earnings from Netflix, Intel, Capital One, McCormick",https://www.cnbc.com/2026/01/16/cramers-week-ahead-earnings-from-netflix-intel-capital-one-mccormick.html
"January 16, 2026",Google files to appeal search monopoly case,https://www.cnbc.com/2026/01/16/google-files-to-appeal-search-monopoly-case.html
"January 16, 2026",More employers worry about workers' financial well-being, research shows",https://www.cnbc.com/2026/01/16/employers-focusing-more-on-employee-financial-wellbeing-study-shows.html
"January 16, 2026",Republicans want to end the 'marriage penalty' for this childcare tax credit,https://www.cnbc.com/2026/01/16/child-and-dependent-care-tax-credit.html
"January 16, 2026",Labor Department accused of echoing Nazi slogan in social media post,https://www.cnbc.com/2026/01/16/trump-labor-nazi-slogan-social-media.html
"January 16, 2026",Education Department to delay collections on defaulted student loans,https://www.cnbc.com/2026/01/16/student-loan-collections-paused.html
"January 16, 2026",Earnings reports, fears around interest rate outlook may sway markets next week",https://www.cnbc.com/2026/01/16/stock-market-next-week-outlook-for-jan-19-23-2026.html
"January 16, 2026",OpenAI has committed billions to recent chip deals. Some big names have been left out,https://www.cnbc.com/2026/01/16/openai-chip-deal-with-cerebras-adds-to-roster-of-nvidias-and-broadcom.html
"January 16, 2026",One of our top stocks this week just lost its CFO – what it means for investors,https://www.cnbc.com/2026/01/16/one-of-our-top-stocks-this-week-just-lost-its-cfo-what-it-means.html
"January 16, 2026",Hassett pivots to possible 'Trump cards' amid credit card battle with banks,https://www.cnbc.com/2026/01/16/white-house-hassett-trump-cards-credit-card-battle.html
"January 16, 2026",Can Home Depot's AI-powered push to court pros move the needle in its own business?,https://www.cnbc.com/2026/01/16/can-home-depots-ai-powered-push-to-court-pros-move-the-needle-in-its-own-business.html
"January 16, 2026",These stocks reporting earnings next week have a history of beating expectations,https://www.cnbc.com/2026/01/16/these-stocks-reporting-earnings-next-week-have-a-history-of-beating-expectations.html
"January 16, 2026",Coastal Virginia Offshore Wind to restart work after judge lifts Trump suspension,https://www.cnbc.com/2026/01/16/biggest-offshore-wind-project-in-us-to-resume-construction-after-judge-lifts-trump-suspension.html
"January 16, 2026",OpenAI to begin testing ads on ChatGPT in the U.S.,https://www.cnbc.com/2026/01/16/open-ai-chatgpt-ads-us.html
"January 16, 2026",OpenAI to begin testing ads on ChatGPT in the U.S.,https://www.cnbc.com/2026/01/16/open-ai-chatgpt-ads-us.html
dsci560@DSCI560:~/Desktop/bhargav_4651477356/data/processed_data$ 

```

REFERENCES

1. <https://realpython.com/modern-web-automation-with-python-and-selenium/>
2. <https://stackoverflow.com/questions/38552949/extract-json-from-script-tag-with-beautifulsoup>
3. <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>
4. <https://www.selenium.dev/documentation/webdriver/browser/chrome/>