

## ASSIGNMENT 1 REPORT

**Name:** Tusharbir Singh Mutty

**Student Id:** 110193040

**Topic:** Home Internet Service Provider Web Scraping (VMedia)

**Tool Used:** Java + Selenium WebDriver

**Date:** 22-05-2025

---

### A. ABOUT THE ISP SITE VMEDIA

<https://www.vmedia.ca/en>

VMEDIA is a Canadian telecommunications company that offers affordable home services like internet, TV, phone, and home security. Founded in 2013 and based in Toronto, VMedia aims to provide Canadians with flexible, no-contract options as an alternative to larger providers .

In 2020, VMedia launched RiverTV, Canada's first standalone live and on-demand streaming TV service . The company was acquired by Quebecor in 2022 and now operates under Freedom Mobile

VMedia is known for its competitive pricing, unlimited usage, and commitment to customer service, making it a popular choice for Canadians seeking cost-effective telecom solutions.

---

### B. TASKS OF ASSIGNMENT 1

#### TASK 1: BASIC PLAN SCRAPING

For Task 1, (<https://www.vmedia.ca/en/internet/ontario>) is the link as my target to scrape internet service provider plans. The goal was to extract basic information from each internet plan card. I used **WebDriverWait** to ensure elements were loaded before interaction.

#### **About the Code:**

The code scrapes internet plan data from VMedia's Ontario page using **Selenium WebDriver** and **ChromeDriver**. Key components used:

- **WebDriver & ChromeDriver:** To launch and control the browser.
- **WebDriverWait + ExpectedConditions:** Ensures elements like plan cards load before interaction.
- **By.cssSelector:** Selects HTML elements containing plan data.
- **List<WebElement>:** Collects multiple plan elements for iteration.

## Data Extracted:

- Plan Name
- Unlimited Label means the total data available at high speed in the plan
- Price (handled for CSS ::before tag)
- Download Speed
- Upload Speed
- Technology (e.g., Cable, DSL, FTTN)

All extracted information was saved in a CSV file named **vmedia\_featured\_plans.csv**.

## Screenshots:

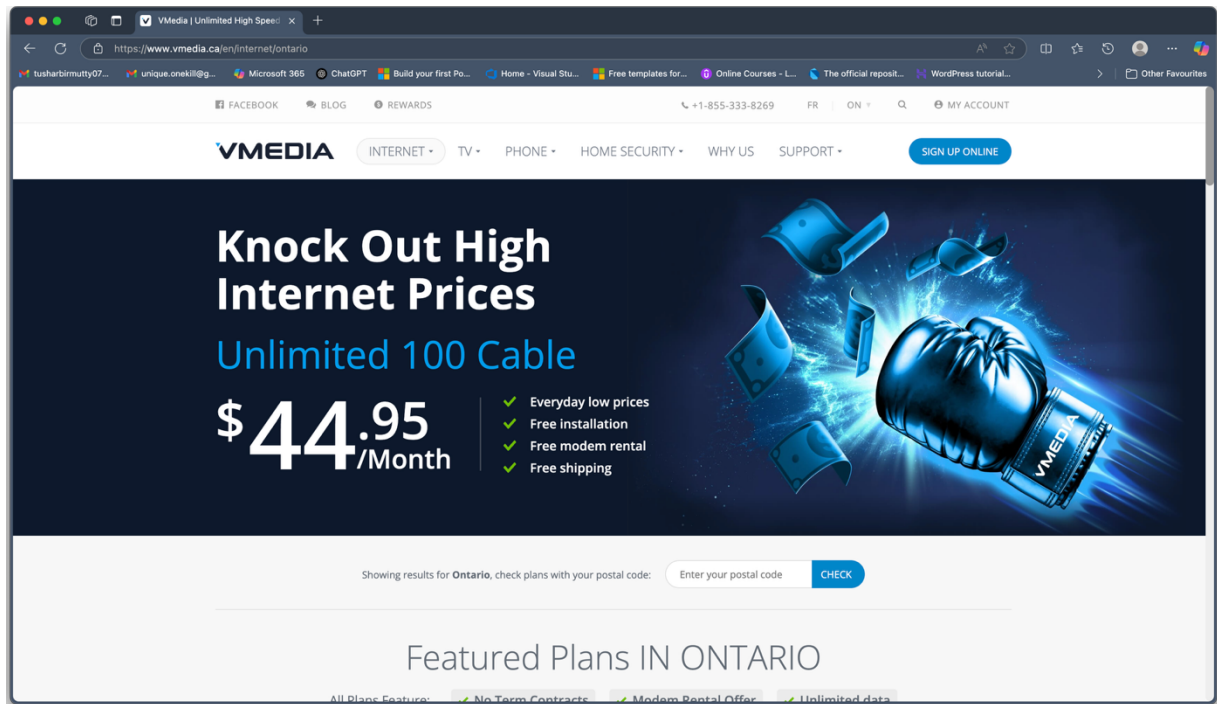


Figure 1: Initial Site: Screenshot of the site page used for the scraping in the task 1

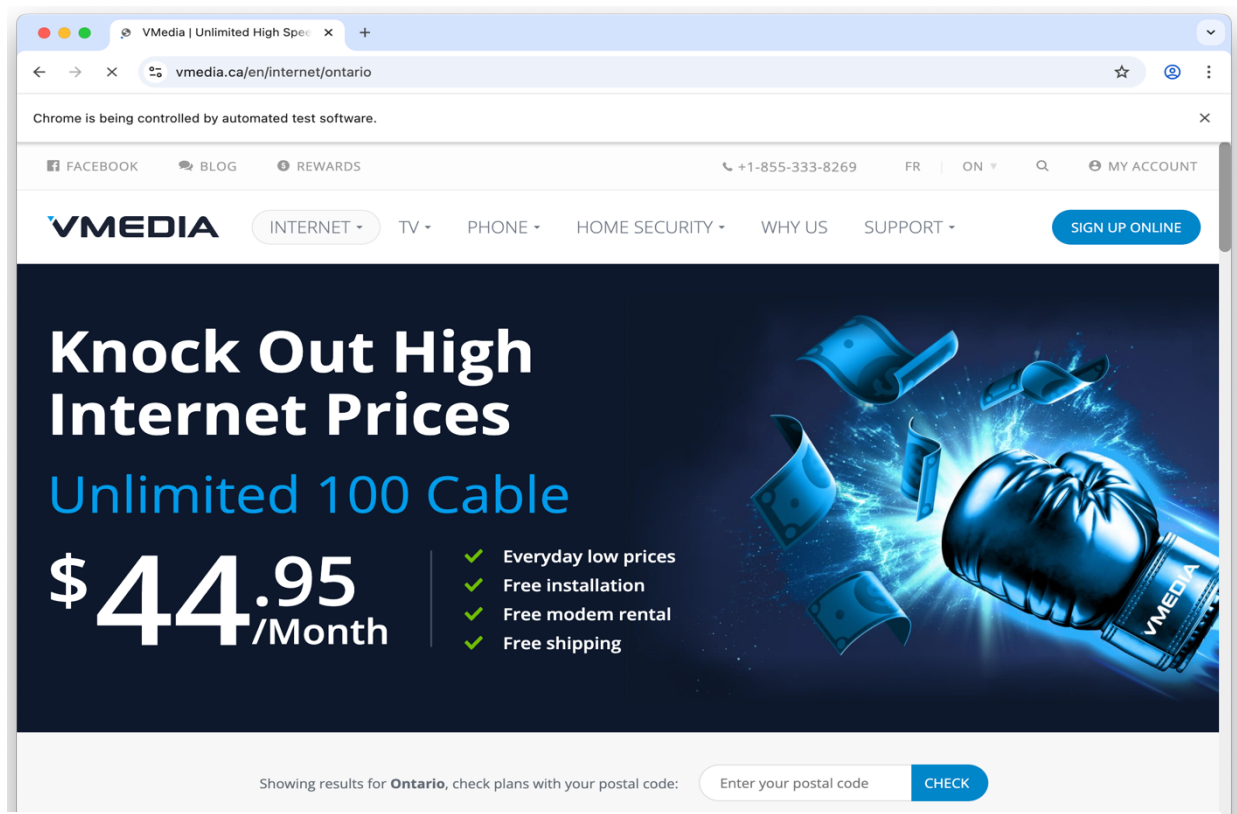


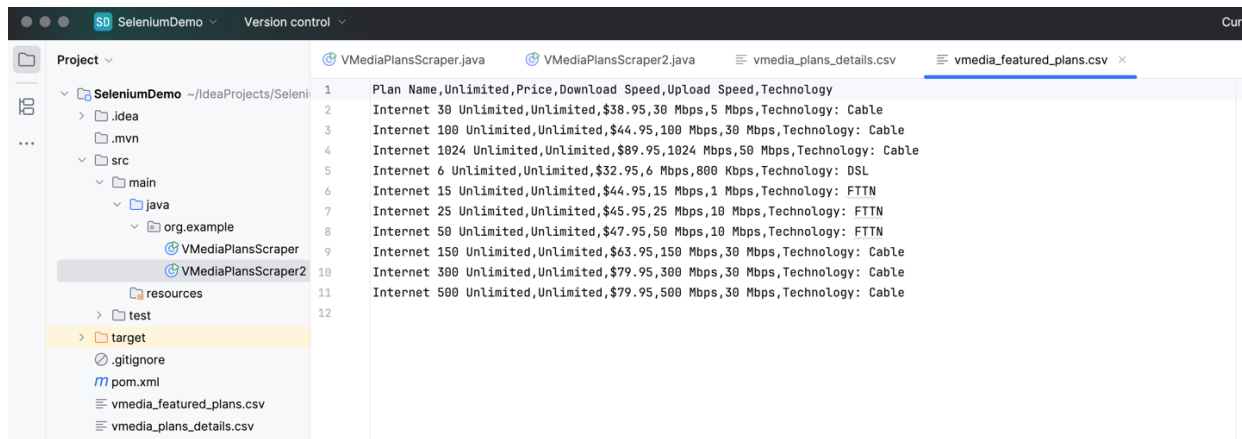
Figure 2: VMedia site opened in chrome by selenium



```
Run VMediaPlansScraper
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java ...
May 22, 2025 10:43:50 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
WARNING: Unable to find CDP implementation matching 136
May 22, 2025 10:43:50 PM org.openqa.selenium.chromium.ChromiumDriver lambda$new$5
WARNING: Unable to find version of CDP to use for 136.0.7103.114. You may need to include a dependency on a specific version of the CDP using something
Scraping complete! Data saved to vmedia_featured_plans.csv

Process finished with exit code 0
```

Figure 3: Console Output of Task 1



Plan Name	Unlimited	Price	Download Speed	Upload Speed	Technology
Internet 30	Unlimited	\$38.95	30 Mbps	5 Mbps	Cable
Internet 100	Unlimited	\$44.95	100 Mbps	30 Mbps	Cable
Internet 1024	Unlimited	\$89.95	1024 Mbps	50 Mbps	Cable
Internet 6	Unlimited	\$32.95	6 Mbps	800 Kbps	DSL
Internet 15	Unlimited	\$44.95	15 Mbps	1 Mbps	FTTN
Internet 25	Unlimited	\$45.95	25 Mbps	10 Mbps	FTTN
Internet 50	Unlimited	\$47.95	50 Mbps	10 Mbps	FTTN
Internet 150	Unlimited	\$63.95	150 Mbps	30 Mbps	Cable
Internet 300	Unlimited	\$79.95	300 Mbps	30 Mbps	Cable
Internet 500	Unlimited	\$79.95	500 Mbps	30 Mbps	Cable

Figure 4: CSV output of Task 1

## TASK 2: DETAIL PAGE SCRAPING

For Task 2, I extended the scraper to follow each plan's detail page link and extract additional plan-specific information.

### **Additional Data Extracted:**

- Full Plan Features
- Important Information (offers and policies)
- Plan Features (bullets)
- Pros (e.g., No Contract, No Hidden Fees)
- Compatible Modems URL

### **ABOUT THE CODE:**

1. **JavascriptExecutor**: Used to **open each plan's detail page in a new browser tab** via JavaScript: `window.open()` simulates a user opening a new tab.

2. **Tab Handling with driver.getWindowHandles():** Collects all open browser tabs/windows. Switches to the **newly opened tab** using driver.switchTo().window(...) to load the plan's detail page.
3. **Detail Page Data Extraction:** After switching to the plan's page, it extracts:
  - > **Description** from the .internet-package\_\_description sections
  - > **Features** from a list of feature elements.
  - > **Pros** (advantages or highlights) from pros section.
  - > **Compatible modem link**, if available.
4. **Tab Cleanup:** After extracting info, the **detail tab is closed**. Focus is switched back to the original tab (the main plans page) for the next iteration.

This data was appended to the original dataset and saved in a new CSV file called vmedia\_plans\_details.csv.

## SCREENSHOTS

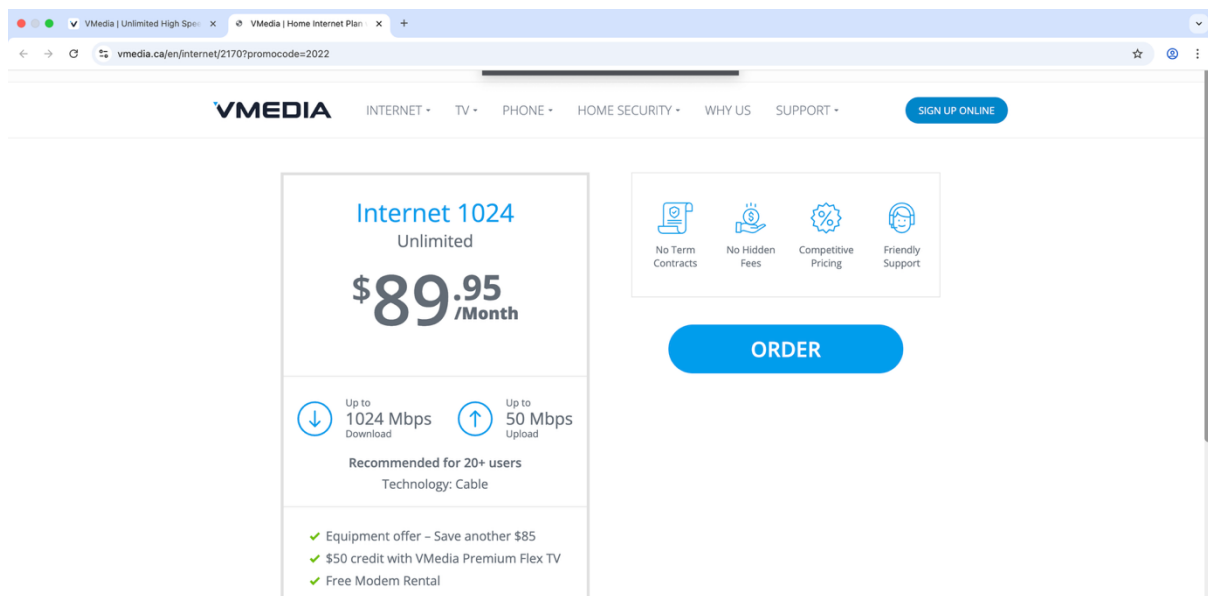


Figure 5: Automated tabs opening by selenium

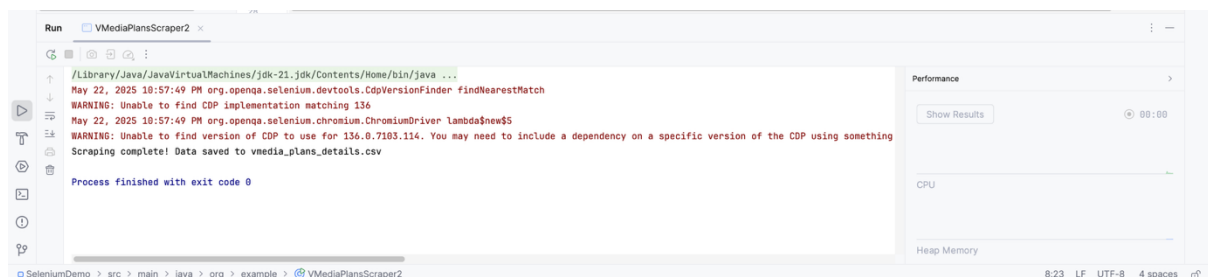


Figure 6: Console output of task 2

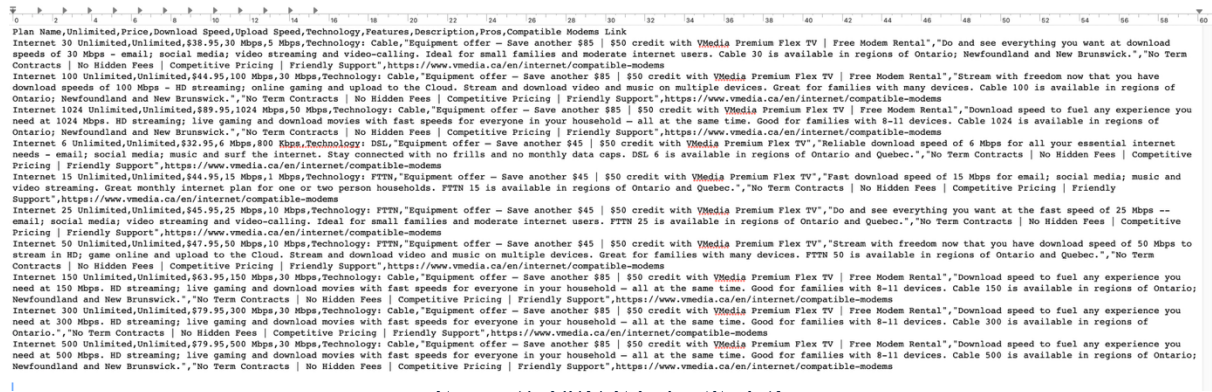


Figure 7: CSV File for Task 2

## TASK 3: ADVANCED SELENIUM COMMANDS

For Task 3, I demonstrated the use of the following advanced Selenium functionalities:

### 1. Explicit Waits:

WebDriverWait and ExpectedConditions were used to **wait for dynamic elements** (e.g., plan cards, descriptions) to **fully load before interaction**, ensuring stability and reliability of the scraper.

### 2. Handling Multiple Tabs (Pop-Up Simulation):

The plan detail pages were **opened in new tabs** using JavascriptExecutor with window.open(). Then, the code **switched between tabs** using driver.getWindowHandles() and driver.switchTo().window(...), mimicking advanced user behavior like handling pop-up windows or new pages.

### 3. Switching Contexts and Closing Tabs:

After extracting additional information (features, pros, modem links), the new tab was **closed** and the driver **returned to the original tab**, maintaining the session efficiently.

These techniques go beyond basic Selenium use and demonstrate proper application of **advanced browser control and synchronization** — fulfilling the requirements of **Task 3**.

## LinkedIn Learning Certificate

As part of this assignment, I completed the assigned LinkedIn Learning course related to Selenium Web Scraping. The **certificate of completion** is attached with this report as required.

## Conclusion

All three tasks were completed successfully. I extracted both summary and detailed data for internet plans using advanced Selenium features and Java. All output was validated and saved in CSV format. Screenshots were captured for verification.

---