Tribhuvan University

Institute of Science and Technology

A Final Year Internship Report
On
"Web Scraping"
At
Prime Vendor Nepal Pvt. Ltd.

Submitted To:

Office of the Dean

Institute of Science and Technology

Tribhuvan University

Kirtipur, Nepal

In the partial fulfillment of the requirement for the Bachelor of Science in Computer Science and Information Technology (BSc. CSIT).

Submitted By:

Rabin Neupane

(TU Exam Roll No. 24109/076)

Under the supervision of

Mr. Santosh Dhungana

June, 20

MENTOR'S RECOMMENDATION

I hereby endorse the submission of the internship report titled "Web Scraping Intern" concerning the "Prime Vendor Nepal Pvt. Ltd. " project, which was completed under my guidance by **Rabin Neupane**. This report is being submitted as part of the requirements for the degree of B.Sc. in Computer Science and Information Technology and should proceed for evaluation.

……

Mr. Nabin Adhikari

Java Developer

Prime Vendor Nepal Pvt Ltd

SUPERVISOR'S RECOMMENDATION

I hereby recommend that the report prepared under my supervision by **Rabin Neupane** in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Technology (BSc. CSIT) be processed for evaluation.

……..…….

Mr. Santosh Dhungana

Internal Supervisor

Asian school of Management and Technology

Department of Computer Science and Technology

LETTER OF APPROVAL

This is to certify that this internship report prepared by **Rabin Neupane** entitled in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Technology (BSc. CSIT) has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

.....	.....
Mr. Supervisor	Mr. Supervisor
Internal Supervisor	**Internship Supervisor**
.....	.....
Internal Examiner	**External Examiner**

ACKNOWLEDGEMENT

This report has been prepared as part of the requirements for my Bachelor's degree in Computer Science and Information Technology, in fulfillment of my internship at **Prime Vendor Nepal Pvt. Ltd**. This internship has been an invaluable opportunity for learning and professional growth, and I have had the privilege of working alongside some exceptional individuals who have guided me throughout this journey.

I would like to extend my heartfelt gratitude to my mentor and senior Software Developer, **Mr. Nabin Adhikari**, whose unwavering support and guidance made my transition into the workplace seamless. His mentorship has been instrumental in enhancing both my theoretical knowledge and practical skills.

I also wish to express my appreciation to my supervisor, **Mr. Santosh Dhungana**, for his guidance during the internship and assistance in the preparation of this report. Additionally, I would like to acknowledge the encouragement and support of our college principal, **Er. Anil Lal Amatya**, in fostering our growth.

I consider this internship experience a significant milestone in my career development, and I am committed to applying the knowledge and skills I have acquired in the best possible way.

With Regards,

Rabin Neupane

(T.U. Roll No. 24109/076)

ABSTRACT

This internship projects on Web Scraping at Prime Vendor Nepal Pvt. Ltd. aimed to extract and analyze the data from the various websites. The web scraping tool automates the collection of diverse data sets, including product prices, user reviews, and other relevant information from targeted web pages. The extracted data is then processed and visualized through dynamic charts and graphs, providing valuable insights for business intelligence and decision-making.

In the current era of big data, the ability to efficiently collect and analyze large volumes of web-based information is crucial. This web scraping system enables organizations to obtain timely and accurate data, facilitating competitive analysis, market research, and strategic planning. The system leverages powerful Java libraries and frameworks such as Jsoup, HTMLUnit and Selenium for robust and scalable performance.

This report outlines the architecture, development process, and technical details of the web scraping system. It also addresses ethical considerations and legal compliance issues associated with web scraping. The practical applications and benefits of this tool in various domains, including e-commerce, finance, government websites and research, are discussed. By automating data collection and analysis, the web scraping system significantly enhances the efficiency and effectiveness of data-driven decision-making processes.

Keywords: Web Scraping, Automation, Java, Jsoup, Selenium, Data Visualization, Analytics, Charts

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ABN	American Business Network
API	Application Programming Interface
BPM	Business Process Management
B2G	Business-to-Government
CAPTCHA	Completely Automated Public Turing test to tell Computers and Humans Apart
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
IT	Information Technology
JDBC	Java Database Connectivity
JS	JavaScript
JVM	Java Virtual Machine
MD	Managing Director
MySQL	My Structured Query Language
OS	Operating System
PVN	Prime Vendor Nepal
SQL	Structured Query Language

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Introduction

Web scraping, or web crawling, refers to the process of fetching and extracting arbitrary data from a website. This involves downloading the site's HTML code, parsing that HTML code, and extracting the desired data from it. Web Scraping is the automation of the data extraction process from websites. One way is to copy-paste the data, which is both tedious and time-consuming manually, So This event is done with the help of web scraping software known as web scrapers. They automatically load and extract data from the websites based on user requirements. These can be custom-built to work for one site or can be configured to work with any website.

The primary motivation for web scraping is the need for timely and accurate data that can inform business decisions, market research, and strategic planning. Traditional methods of data collection, which often involve manual efforts, are not only time-consuming but also prone to errors. Web scraping offers a more efficient and reliable alternative by automating the data retrieval process. Web scraping is widely used across various industries. In e-commerce, it helps businesses monitor competitor prices and product availability. In finance, it enables the aggregation of market trends and news articles. Researchers and academics utilize web scraping to collect data for studies and analyses.

This report explores the development of a web scraping system using Java, a versatile and powerful programming language. Java offers a range of libraries and frameworks, such as Jsoup and Selenium, which facilitate the creation of robust and scalable web scraping solutions. The following sections will detail the architecture and implementation of the system, discuss the technical aspects of web scraping, and highlight its practical applications and benefits. Additionally, the report will address the ethical considerations and legal implications associated with web scraping, providing a comprehensive overview of the practice.

1.2 Problem Statement

Manually copying information from websites is tedious, especially for large datasets or data that updates frequently. The primary challenge lies in the efficient and automated extraction of relevant data from diverse web sources, each with its unique structure and content. Websites frequently update their content and employ various anti-scraping measures,

making it difficult to maintain the accuracy and reliability of the data extraction process. Additionally, handling dynamic content, such as JavaScript-rendered pages, further complicates the scraping process.

1.3 Objectives

- To automate large-scale data extraction and minimize human error, web scraping eliminates the need for tedious manual copying.
- To analyze frequently updated data on websites, web scraping enables the monitoring of trends and real-time information.
- To maintain high accuracy and reliability in data extraction despite frequent website updates and anti-scraping measures.
- To handle dynamic content by implementing solutions for scraping JavaScript-rendered pages and other dynamically generated web content.

1.4 Scopes and Limitations

1.4.1 Scopes

This system is designed to automatically collect data from various websites, including those with both regular and dynamic content. It uses advanced methods to handle web pages that load content with JavaScript, making sure it works well with modern websites. The system is optimized for fast and efficient data collection to meet real-time needs. It includes an easy-to-use interface for setting up and managing web scraping tasks, allowing users to choose target websites, data fields, and schedules for data extraction. Comprehensive guides and training materials will help users effectively use and manage the system. The collected data will mostly come from public websites.

1.4.2 Limitations

The limitations of the project can be listed out as follows:

- Extracted data might not always be accurate or complete due to changes in website structure.
- Privacy concerns as the data is stored without the consent of the users.
- Accuracy of data collected may be affected as the hits may also contain the office IPs.
- Some websites block or limit scraping activities with CAPTCHAs, IP blocking, or rate limiting.

1.5 Report Organization

Altogether the report is divided into four different chapters, each representing different phases of the internship report. The chapters can be described as, in **Chapter 1**, it deals with the introductory part of the report and explains what the report is about, what are problem statements, scope and limitation. **Chapter 2** is all about the organization, introduction to the organization, what hierarchy that particular organization follow, working domains of the organization and description of the intern department. **Chapter 3** deals with the internship activities, what are our roles and responsibilities, what are the things we perform over the period of intern and the description of the project we did and the task or activities we performed. **Chapter 4** is all about the conclusion and the things we learned during our internship period.

CHAPTER 2: ORGANIZATION DETAILS AND LITERATURE REVIEW

2.1 Introduction to Organization

Prime Vendor Nepal (PVN) is a key division of American Business Network, Inc. (ABN), a Business Process Management (BPM) company with offices in New York, USA, and Vancouver, Canada. PVN leverages the extensive resources and expertise developed by ABN, particularly in the field of e-procurement solutions.

ABN provides a wide range of business services, including application innovation, business analytics, business strategy, commerce consulting, and procurement and logistics. Additionally, ABN offers managed outsourcing services such as application management, global process services, IT infrastructure services, IT outsourcing, and staff training. The parent company also owns and operates Prime Vendor Inc., a procurement solution service company based in North Carolina. Prime Vendor Inc.'s flagship product is a highly advanced modular B2G (Business-to-Government) e-procurement solution software, renowned for its efficiency and effectiveness in managing government procurement processes.

Prime Vendor Nepal capitalizes on the success and resources of Prime Vendor Inc.'s e-procurement software. PVN plays a vital role within ABN, offering comprehensive IT expertise and services, including IT design and development, data entry services, and IT and business process outsourcing services management. Through its strategic location and dedicated team, Prime Vendor Nepal provides clients with top-notch IT solutions and services, ensuring they benefit from the latest advancements in technology and business process management. PVN’s commitment to excellence makes it an indispensable part of ABN's global operations, driving innovation and efficiency for businesses worldwide.

Table 2: Contact Details of the Company

Company Name	Prime Vendor Nepal Pvt. Ltd.
Address	Dhumbarahai, Kathmandu
Contact	01-4989747
Mail	hr@nepaldatacenter.com
Website	https://primevendornepal.com.np/

2.2 Organizational Hierarchy

The company's hierarchical organizational structure is structured with the Managing Director (MD) at the pinnacle, overseeing the entire operation. Below the Chairman, there are two key leadership roles: the Project Manager and the Data Manager.

The MD is responsible for the overall direction and strategic vision of the company, while the manager focuses on key operational aspects.

Under the MD, there are two critical departments: IT and Data, each led by a dedicated manager.

The Project Manager supervises a team of Research and Development Engineers and Senior Software Developers. These professionals are responsible for driving innovation and creating cutting-edge products. Additionally, within the Senior Software Developer role, there is an opportunity for interns to gain valuable experience and contribute to the development process.

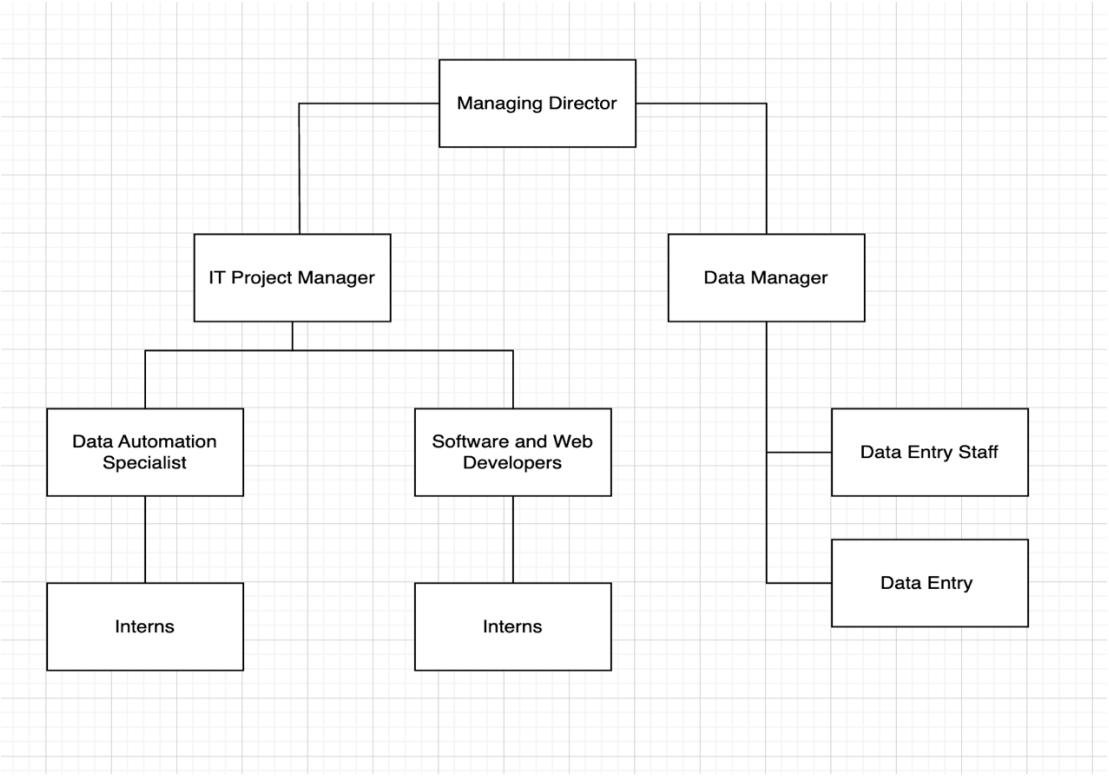


Figure 1: Hierarchical Structure of Organization

2.3 Working Domains of Organization

Alternative Technology Pvt. Ltd. offers a variety of services like Web Development, Mobile Application Development, Digital Marketing, and others such as:

1. Website Development
    - Java
    - Python
2. Mobile Application Development
    - Flutter
3. Database
    - Microsoft SQL Server
    - SQLite
4. Task Management
    - Trello
5. Code Management
    - Bitbucket and Github
6. Deployment
    - AWS

2.4 Description of Intern Department

As a web scraping intern at Prime Vendor Nepal Pvt. Ltd., the intern will have many opportunities to learn and focus on web scraping. During the internship, the intern will get hands-on experience with different web scraping tools and work on various projects. This will help the intern understand how to extract and process data from websites.

At first, the intern will spend time getting to know the company and its current projects. After that, the intern will focus on the assigned project, working closely with other interns, junior developers, and senior developers. The senior developers will guide the intern and make sure the intern is working efficiently.

To succeed, it is important for the intern to stay productive and complete the tasks given by the supervisor. The intern will also have daily meetings with the supervisor to update them on progress and get feedback. This will help the intern learn and grow in the field of web scraping, preparing for future jobs.

The duration and relevant details of internship are as shown in the table 2 :

Table 2: Duration of Internship

Start Date	16th April
End Date	9th July
Total Duration	12 Weeks
Position	Web Scraping Intern
Supervisor	Mr. Nabin Adhikari
Office Hours	9 AM – 6 PM

2.5 Literature Review

Web scraping, also known as web data extraction, is a technique used for extracting information from websites. It is an essential tool for businesses, researchers, and developers who need to collect large amounts of data efficiently. This literature review examines the various methods, applications, challenges, and ethical considerations associated with web scraping.

Web scraping methods have evolved significantly over the years. Traditional techniques involve writing custom scripts using programming languages like Python, favored for its extensive libraries such as BeautifulSoup, Scrapy, and Selenium (Mitchell, 2015). These tools allow for the parsing of HTML and XML documents and can interact with JavaScript-driven web content. More advanced techniques include machine learning approaches to identify and extract data patterns from websites.

Despite its benefits, web scraping faces several challenges. One of the primary technical challenges is dealing with dynamic content loaded by JavaScript, which traditional scraping tools struggle to handle (Mihindukulasooriya et al., 2018). Websites frequently change their structures to deter scraping, requiring constant updates to scraping scripts. Performance issues can also arise, especially when scraping large datasets, as this can be resource-intensive and time-consuming.

Web scraping raises significant ethical and legal concerns. Many websites' terms of service prohibit automated data extraction, and scraping can lead to legal actions against the scraper. The case of LinkedIn vs. hiQ Labs, where LinkedIn sued hiQ for scraping user data, highlighted the legal risks involved. Ethically, scraping must consider data privacy and the potential misuse of collected data. Researchers advocate for ethical guidelines and best practices to ensure responsible use of web scraping (Krotov & Silva, 2018).

Web scraping is a powerful tool for data collection with diverse applications across different fields. While it offers significant benefits, it also poses technical, ethical, and legal challenges that must be addressed. Future research should focus on developing more robust and ethical scraping techniques and frameworks to balance the benefits of web scraping with the need for compliance with legal and ethical standards.

CHAPTER 3: INTERNSHIP ACTIVITIES

3.1 Roles and Responsibilities

As a web scraping intern at Prime Vendor Nepal Pvt. Ltd., the intern will play a crucial role in extracting and processing data from various websites. This role is an excellent opportunity to gain hands-on experience in web development, data analysis, and the practical application of web scraping techniques. The allocation of tasks assumes the intern's readiness for query development of web scraping with Java. On a weekly basis, a task review is conducted, and task assignment to the intern is the responsibility of the supervisor. Upon completing each task, the supervisor undertakes a comprehensive review and validation process.

Some of the assigned tasks and responsibilities during the internship period included:

- Gaining a foundational understanding of web scraping.
- Learn the basics of web scraping, including its applications and best practices.
- Understand the ethical and legal considerations associated with web scraping.
- Develop Java scripts and tools for generating web scraping tasks
- Scraping website by using different java library.
- Identify websites whether they are block by side or not.
- Identify website side type and write scripts according to side like List, grid, table.
- Utilize Selenium scripts to bypass blocking mechanisms and successfully scrape data from restricted websites.

3.2 Weekly Log

The following section provides a brief overview of the weekly logs for the intern report (table 3.1). Throughout the course of the internship, the intern diligently recorded their activities and tasks on a weekly basis, providing a detailed account of their journey, progress, and contributions to the organization. These logs serve as a valuable record of the intern's hands-on experiences, skill development, and project involvement during the 12-week internship program.

The logs for each week provide a glimpse into the intern's assigned tasks and responsibilities, offering valuable insights into how their role within the organization evolved over time. From initial onboarding and project familiarization to the successful

deployment of key projects, these weekly logs capture the dynamic nature of the internship experience.

Table 3.1: Weekly Log of Internship

Week	Activity	Activity/ Task Performed
1st Week	Orientation and Introduction	- Attend Orientation sessions - Meet the team members and supervisors - Set up the development environment and software
2nd Week	Basic Training and Initial Tasks	- Learn web scraping fundamentals - Study Java and relevant libraries - Start small scripting tasks
3rd Week	Developing Simple Web Scraping Scripts	- Write basic Java scripts for scraping - Experiment with scraping libraries - Receive feedback on initial scripts
4th Week	Handling Different Website Structure	- Analyze various website layouts - Develop scripts for different structures - Refine scripts with supervisor feedback
5th Week	Advanced Scraping Techniques	- Learn dynamic content handling - Use Selenium for scraping - Tackle more complex tasks under supervision
6th Week	Data Processing and Storage	- Learn data cleaning techniques - Store data in databases - Validate data quality and integrity
7th Week	Task Review and Improvement	- Review tasks with supervisors - Implement feedback - Optimize existing scripts and processes
8th Week	Collaboration and Project Work	- Collaborate on a larger project - Participate in project planning

		- Share progress during team meetings
9th Week	Final Project and Planning	- Plan final web scraping project - Define project scope and objectives - Start preliminary work on final project
10th Week	Final Project development	- Develop final web scraping project - Apply all learned techniques - Conduct testing and validation of the final project
11th Week	Final Project Completion and Presentation	- Complete final project - Prepare and deliver a project presentation - Receive a feedback and showcase achievements
12th Week	Evaluation and Wrap-Up	- Conduct internship review with supervisors - Discuss strengths, weakness and future development areas - Finalize documentation

3.3 Description of the Project(s) Involved During Internship

The project involved developing a robust web scraping solution using Java and Selenium to automate the extraction of specific information from websites. The extracted data was then stored in a MySQL database, with a focus on preventing duplicate entries. The project aimed to enhance data acquisition capabilities by automating the collection of valuable information from various web sources.

3.4 Task / Activities Performed

Throughout my internship, I focused on gaining a solid understanding of the fundamental workflow of web scraping. During this period, I worked on comprehending how web scraping functions operate. I learned how different parts of a website, such as tracker-enabled, tracker-disabled, and database components, all work together. Additionally, I

gained insights into bypassing scraping-disabled functions and handling anti-scraping measures.

Moreover, I developed skills in identifying the structure of web pages, extracting relevant data efficiently, and managing challenges related to dynamic content and CAPTCHA. I also explored various web scraping tools and libraries, enhancing my practical knowledge and technical proficiency in the field.

Different activities were assigned by the supervisor of the organization. The activities I performed as a Java intern are as follows:

Requirement Gathering

The project began with identifying target websites and determining the specific data points to be extracted, such as product details, prices, and descriptions. Analyzing the HTML structure of these websites was crucial to understanding where and how the required data was embedded. Navigate to the target website to start scraping data.

```
@Test
public void testScrapingEcommerce() {
    // Navigate to  e-commerce website
    driver.get("https://ecommerce-playground.lambdatest.io/");
```

Figure 2: Code Snippet of target website

Development of Scraping Scripts

During my internship, I developed web scraping scripts using Java and Selenium WebDriver to automate browser actions required for navigating and interacting with web pages. By leveraging XPath and CSS selectors, I accurately located and extracted relevant data from HTML elements. The process involved identifying elements, automating interactions such as clicking buttons and filling out forms, and handling dynamic content to ensure all necessary data was accessible. proficiency in data extraction and management.

```
wait.until(ExpectedConditions.presenceOfElementLocated(By.xpath( xpathExpression: "//a[contains(.,'Shop by Category')]")));
driver.findElement(By.xpath( xpathExpression: "//a[contains(.,'Shop by Category')]")).click();

wait.until(ExpectedConditions.presenceOfElementLocated(By.xpath( xpathExpression: "//span[contains(.,'Phone, Tablets & Ipod')]")));
driver.findElement(By.xpath( xpathExpression: "//span[contains(.,'Phone, Tablets & Ipod')]")).click();

wait.until(ExpectedConditions.presenceOfElementLocated(By.xpath( xpathExpression: "//*[@id='entry_212408']//div[@class='row']")));
WebElement allProducts = driver.findElement(By.xpath( xpathExpression: "//*[@id='entry_212408']//div[@class='row']"));
List<WebElement> productList = allProducts.findElements(By.xpath( xpathExpression: ".//div[contains(@class,'product-layout product-grid no-desc')]"));
```

Figure 3: Code Snippet of Script Development

Scrape required Data

Only extract the required data that necessary to extract from the website. For example, extract product price and product name from website.

```
for (WebElement product : productList) {
    try {
        WebElement detail = product.findElement(By.xpath( xpathExpression: ".//a[@class='text-ellipsis-2']"));
        WebElement price = product.findElement(By.xpath( xpathExpression: ".//span[@class='price-new']"));

        String productName = detail.getText().trim();
        String productPrice = price.getText().replaceAll( regex: "[^0-9.]",  replacement: "");
        String productImage = detail.getAttribute( name: "href").trim();

        JSONObject productMetaData = new JSONObject();
        productMetaData.put("product_image", productImage);
        productMetaData.put("product_name", productName);
        productMetaData.put("product_price", productPrice);

        scrapedData.put(productMetaData);
        insertDataIntoDatabase(detail.getAttribute( name: "href"), detail.getText(), price.getText());

    } catch (Exception e) {
        System.out.println("Element not found: " + e.getMessage());
```

Figure 4: Code Snippet of Required Elements

Database Integration

Using JDBC (Java Database Connectivity), I established a connection between the Java application and a MySQL database. JDBC provides a standard Java API for database-independent connectivity, enabling seamless interaction with the MySQL database management system. Designing a database schema involved structuring tables to efficiently store the scraped data. This included defining appropriate data types, primary keys, and relationships between tables if necessary. A well-designed schema ensures that data is organized logically and is efficiently retrievable for analysis and reporting purposes.

```java
3 usages
public class Database_Connection {

    // Database URL, username, and password
    1 usage
    private static final String URL = "jdbc:mysql://localhost:3306/Web_Scraping"; // Update with your database URL
    1 usage
    private static final String USER = "rabin"; // Update with your database username
    1 usage
    private static final String PASSWORD = "**********"; // Update with your database password

    // Database connection
    1 usage
    public static Connection getConnection() throws SQLException {
        return DriverManager.getConnection(URL, USER, PASSWORD);
    }
}
```

Figure 5: Code Snippet of Database Connection

Insert Into Database

SQL queries were implemented to insert the extracted data into the MySQL database. This involved constructing `INSERT` statements tailored to the schema's structure and ensuring no duplicate entries by performing a preliminary check. Specifically, the code first executes a SELECT COUNT `(*)` query to determine if a record already exists, and if not, proceeds with the insertion of new data using an INSERT query.

```java
1 usage
private void insertDataIntoDatabase(String imageUrl, String productName, String productPrice) {
    String checkQuery = "SELECT COUNT(*) FROM Scrape_data WHERE product_image = ? AND product_name = ? AND product_price = ?";
    String insertQuery = "INSERT INTO Scrape_data (product_image, product_name, product_price) VALUES (?, ?, ?)";
```

Figure 6: Code Snippet of Database Insert

Data Cleaning and Preprocessing

The extracted data underwent cleaning processes to ensure consistency and usability. This included removing unnecessary characters, formatting text appropriately (e.g., trimming whitespace), and standardizing data formats (e.g., date formats) to facilitate analysis and integration with other systems.

```java
String productName = detail.getText().trim();
String productPrice = price.getText().replaceAll( regex: "[^0-9.]", replacement: "");
String productImage = detail.getAttribute( name: "href").trim();
```

Figure 7: Code Snippet of Data cleaning

Duplicate Data Handling

Duplicate data was addressed through proactive measures within the database operations. By implementing checks before inserting new entries, the script verified whether data already existed based on unique identifiers or keys. This approach minimized redundancy and maintained the consistency of the database.

```
Database_Connection DatabaseHelper = new Database_Connection();
try (Connection connection = Database_Connection.getConnection();
     PreparedStatement checkStatement = connection.prepareStatement(checkQuery);
     PreparedStatement insertStatement = connection.prepareStatement(insertQuery)) {

    checkStatement.setString( parameterIndex: 1, imageUrl);
    checkStatement.setString( parameterIndex: 2, productName);
    checkStatement.setString( parameterIndex: 3, productPrice);

    ResultSet resultSet = checkStatement.executeQuery();
    if (resultSet.next() && resultSet.getInt( columnIndex: 1) > 0) {
        System.out.println("Duplicate data found: " + imageUrl + ", " + productName + ", " + productPrice);
    } else {
        insertStatement.setString( parameterIndex: 1, imageUrl);
        insertStatement.setString( parameterIndex: 2, productName);
        insertStatement.setString( parameterIndex: 3, productPrice);

        insertStatement.executeUpdate();
        System.out.println("Inserted data: " + imageUrl + ", " + productName + ", " + productPrice);
    }
```

Figure 8: Code Snippet of Duplicate Data Handling

Error Handling

Critical sections of the code were encapsulated within try-catch blocks to handle potential errors gracefully. This included database connections, SQL queries, and data processing routines. By anticipating and handling errors, the application-maintained stability and reliability during operation.

```
        try {

    } catch (SQLException e) {
        e.printStackTrace();
    }
```

Figure 9: Code Snippet of Error Handling

Testing

To manually test the web scraping and database integration script, first make sure Selenium WebDriver and MySQL are properly set up. Run the script to check if it correctly navigates the target website, interacts with the necessary elements, and extracts product details. Look at the console output for the JSON data and check that the script connects to the database without issues. Test the insertion of new product data to ensure unique entries are added and duplicates are identified and handled correctly. Check that any errors during data extraction or database operations are logged appropriately. Verify that the extracted data is cleaned and formatted properly before being inserted into the database.

3.5 Tools Used

IntelliJ IDEA:

As a Java I used IntelliJ IDEA to write and manage the code for my web scraping project. The IDE's user-friendly interface made it easy to organize my project files, navigate between classes, and refactor code when needed. I relied on IntelliJ's code editor to write clean and efficient Java code, leveraging features like syntax highlighting and real-time error detection to catch issues early. The built-in tools for running and debugging the code helped me test the scripts directly within the IDE, allowing for quick iterations and adjustments. Overall, IntelliJ IDEA streamlined the development process, making it more efficient to write and maintain the code.

MySQL

MySQL was used in the coding process to store and manage the data extracted by the web scraping scripts. It provided a robust and reliable database management system that allowed for efficient organization of the scraped data into structured tables. Using MySQL, I could execute SQL queries to insert, update, and retrieve data, ensuring the information was stored accurately and was easily accessible for further processing or analysis. This setup facilitated the smooth integration of the data handling aspects of the project, enabling effective management and validation of the collected information within the database.

XAMPP:

In the coding process, XAMPP was used to set up a local development environment that included a MySQL database. This environment allowed for real-time testing and debugging

of the web scraping scripts, ensuring seamless data integration with the database. Using XAMPP, I could easily manage the MySQL database through phpMyAdmin, perform SQL operations, and validate the data extracted by the scripts. This setup provided a convenient and efficient way to develop and refine the code in a controlled, local environment before moving to production.

CHAPTER 4: CONCLUSION AND LEARNING OUTCOMES

4.1 Conclusion

In conclusion, my internship was a valuable experience that greatly improved my technical skills and understanding of software development. I created web scraping scripts using Java and Selenium WebDriver to automate data extraction from websites. This included navigating web pages, interacting with elements using XPath and CSS selectors, and ensuring accurate data collection. I connected the Java application to a MySQL database using JDBC, designed an effective database schema, and implemented SQL queries for data insertion. I also handled duplicate data efficiently with SQL features like INSERT IGNORE or ON DUPLICATE KEY UPDATE. Additionally, I cleaned and preprocessed the data to ensure consistency and used try-catch blocks to manage exceptions and keep the application stable.

Working with senior team members was a crucial part of the project. Their feedback and guidance during code reviews, brainstorming sessions, and design discussions helped improve the code quality, identify issues early, and enhance the program's performance. I conducted thorough testing and validation to ensure the accuracy and reliability of the scraping scripts and database operations. Comprehensive documentation was prepared to explain the methodology, challenges, solutions, and results. Overall, the internship provided a well-rounded learning experience, combining technical development, problem-solving, and teamwork, and equipped me with essential skills for a successful career in software development.

4.2 Learning Outcomes

As a Java intern working on the development of a web scraping system, I had a transformative learning experience throughout the internship. It provided a unique opportunity to bridge theoretical knowledge with practical application. The hands-on experience in creating web scraping scripts, managing databases, and collaborating with senior team members significantly enhanced my technical proficiency and problem-solving skills. This internship not only solidified my understanding of Java and web technologies but also equipped me with the practical tools and confidence needed for future challenges in software development.

The challenges encountered during the project, such as ensuring secure data handling and implementing efficient database management, honed my problem-solving skills and fostered resilience in overcoming obstacles. Collaborating closely with experienced mentors and peers enriched my learning journey, offering invaluable insights and guidance. Moreover, the internship underscored the importance of project management and effective communication in a professional setting. It equipped me with the necessary skills to work under supervision, meet deadlines, and deliver high-quality outputs.

Through this experience, I gained technical proficiency in various aspects of backend development. Additionally, exposure to different testing methodologies and techniques broadened my understanding of software quality assurance practices. Overall, the internship provided a comprehensive learning experience that prepared me for future endeavors in backend development and equipped me with the skills and confidence to tackle real-world challenges in this domain.

REFERENCES

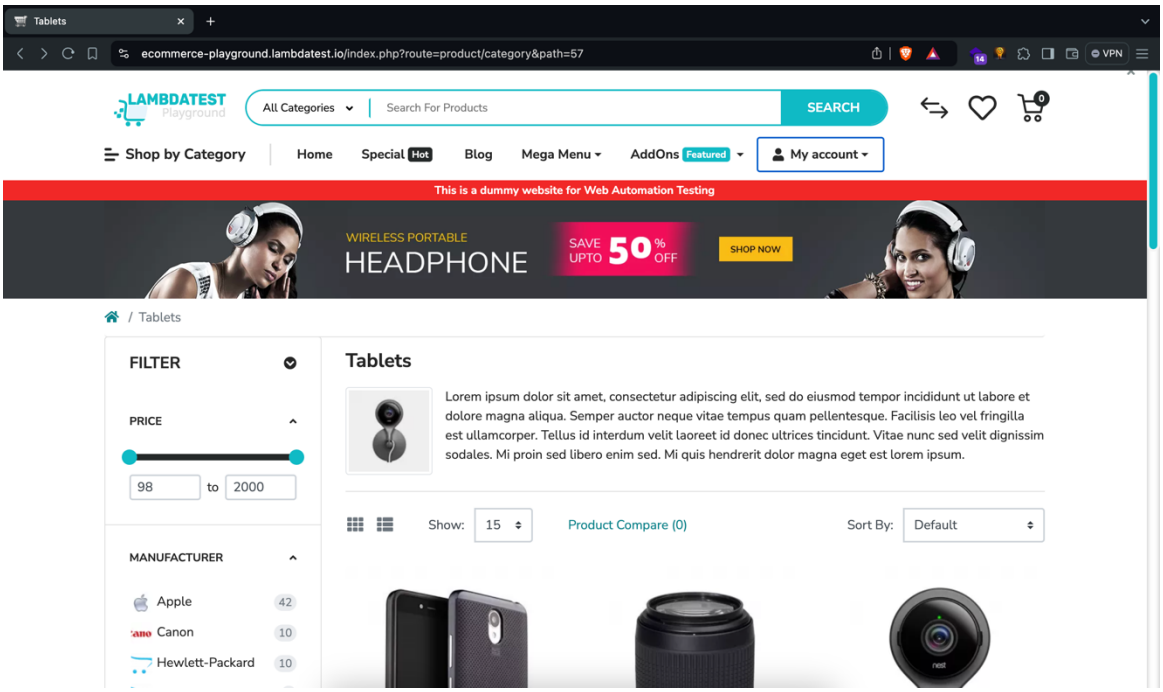
(N.d.). Retrieved from http://primevendornepal.com.np/

GeeksforGeeks. (2024). Introduction to web scraping. Retrieved from https://www.geeksforgeeks.org/introduction-to-web-scraping/

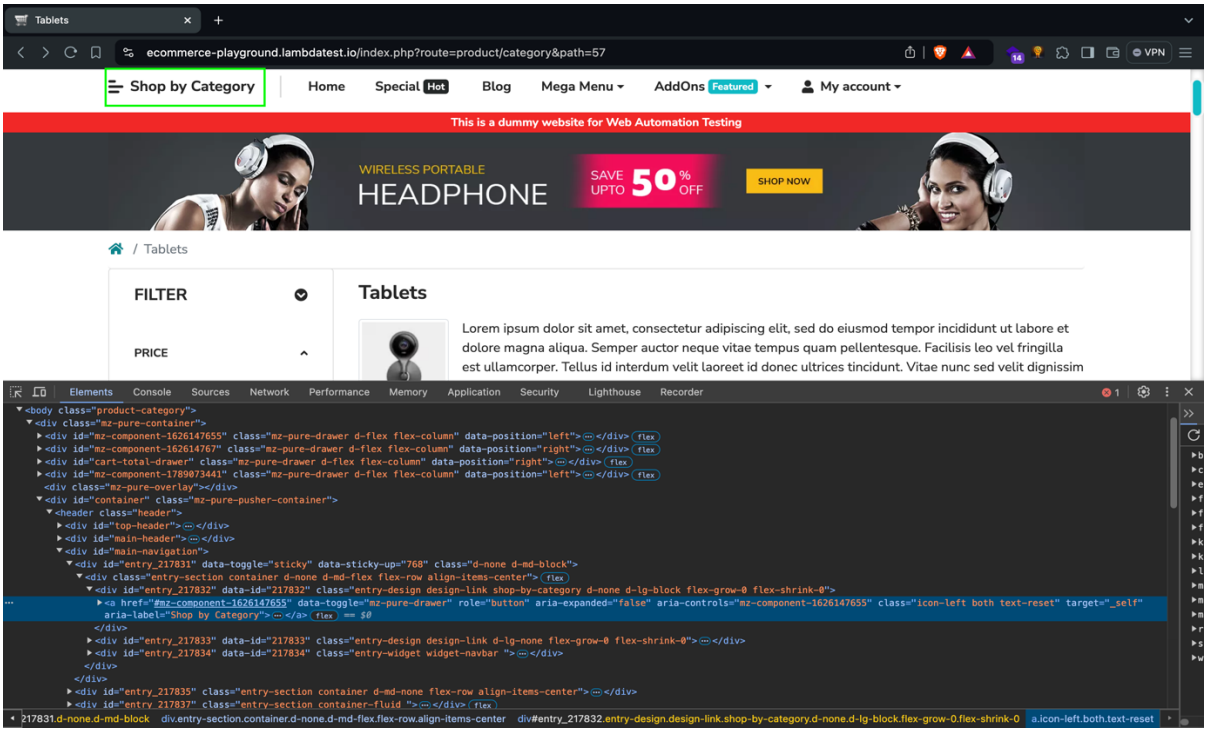
Sahin, K. (2022). Introduction to web scraping with Java. Retrieved from https://www.scrapingbee.com/blog/introduction-to-web-scraping-with-java/

Ferrara, E., De Meo, P., Fiumara, G., & Baumgartner, R. (2014). Web data extraction, applications and techniques: A survey. *Knowledge-Based Systems*, *70*, 301–323. https://doi.org/10.1016/j.knosys.2014.07.007

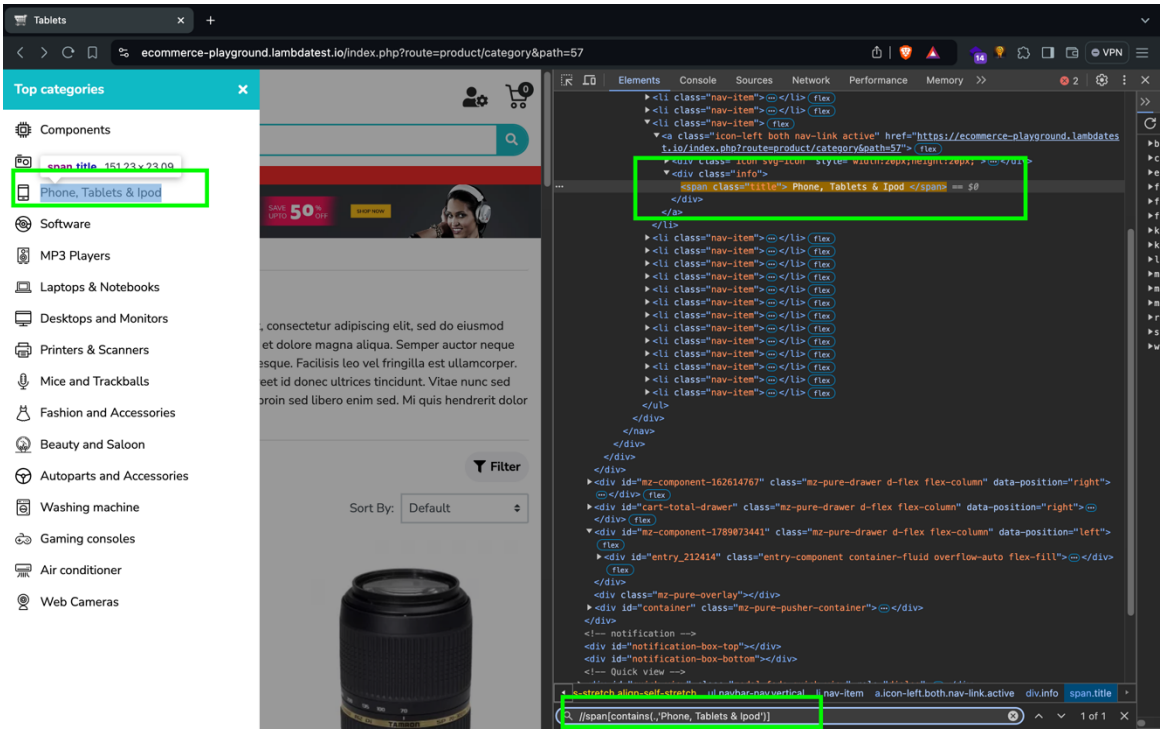
APPENDICE



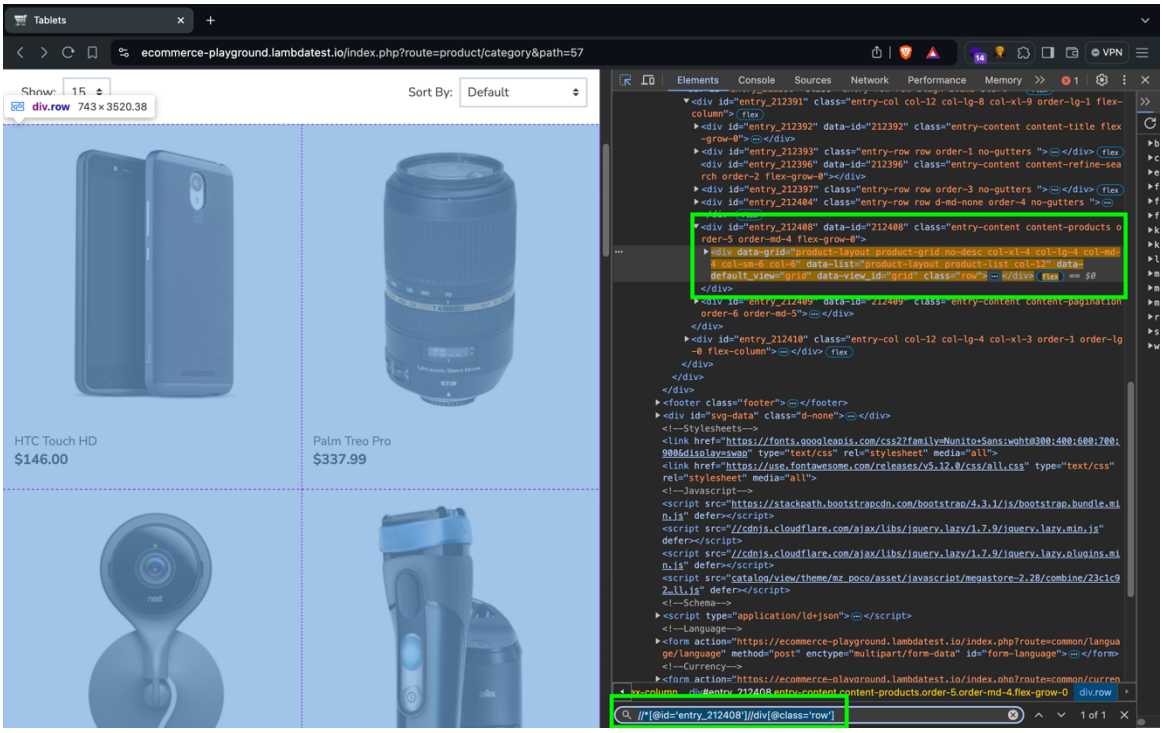
Home Page



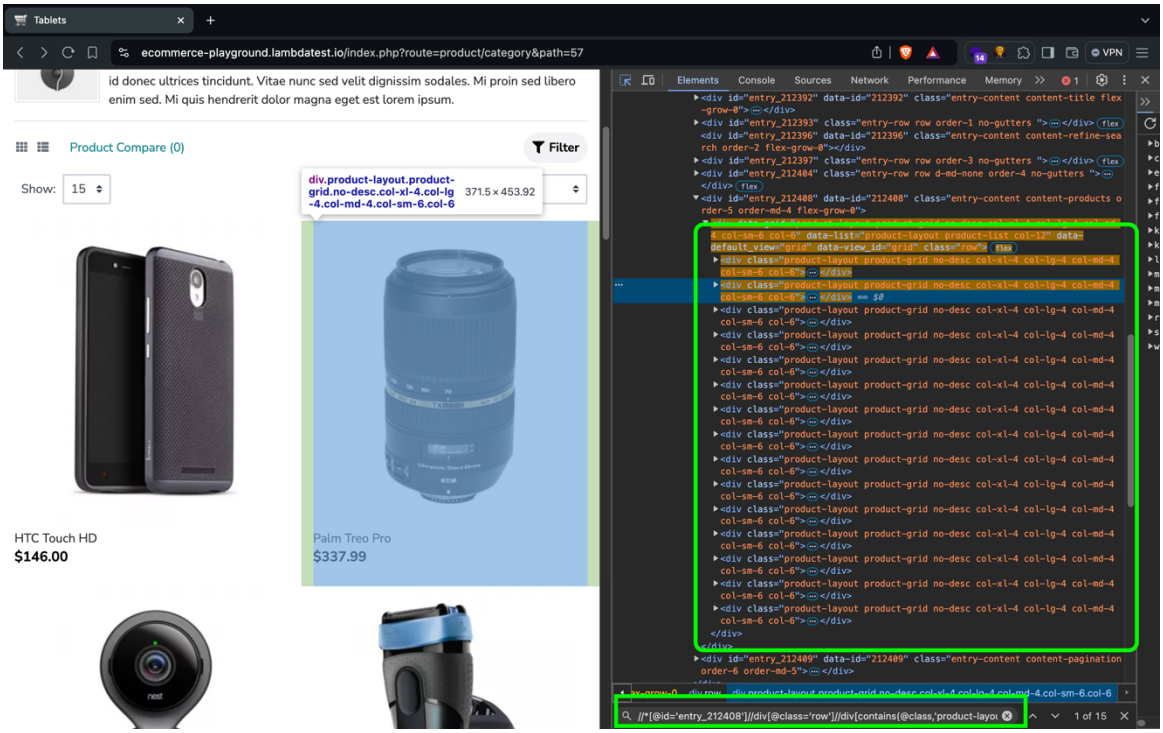
Selection Of Category



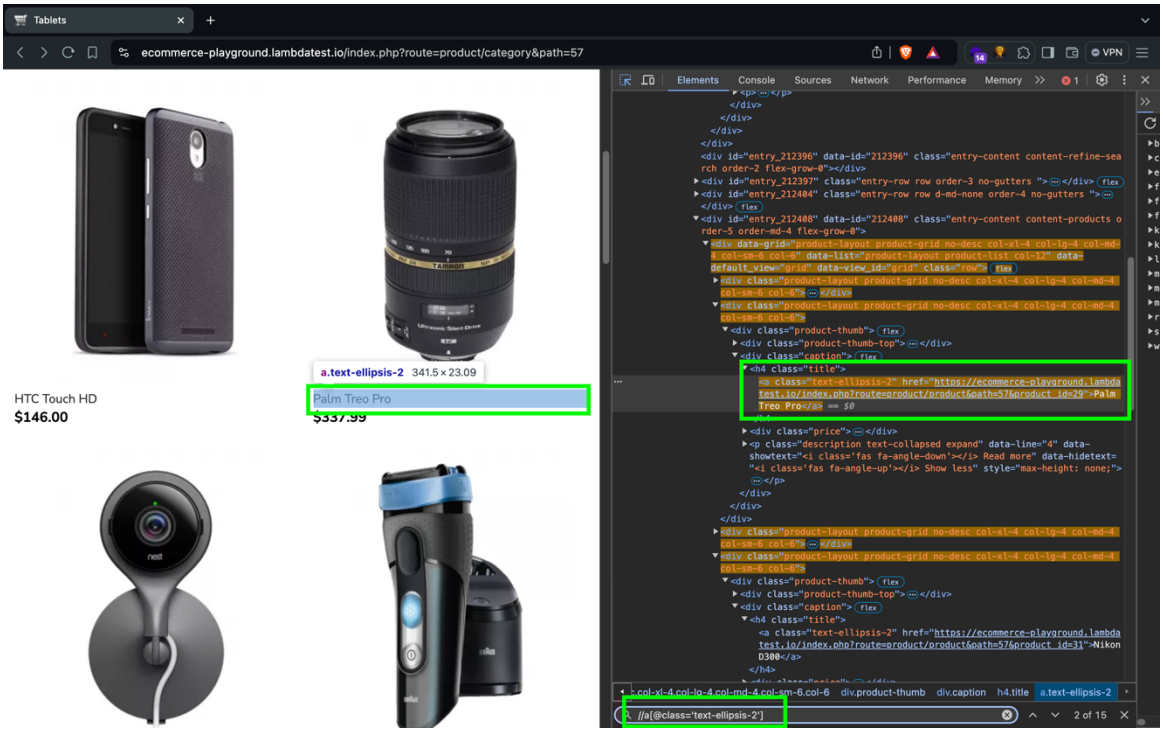
Select the Category Type



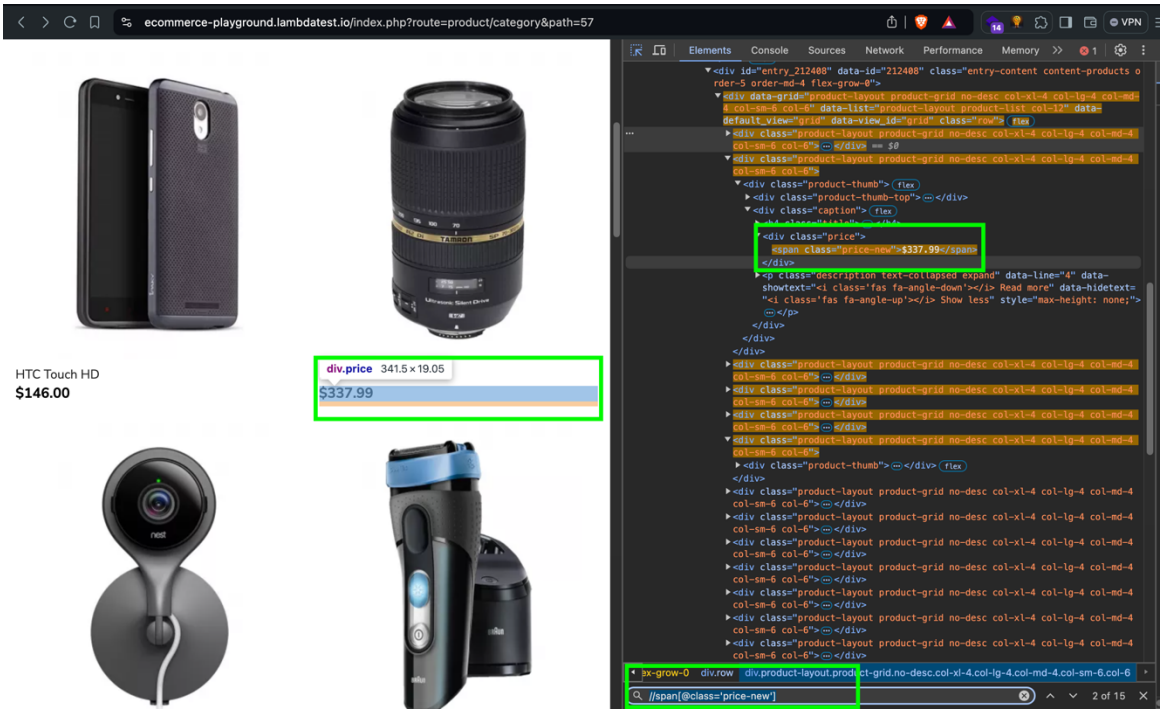
Load all the Products



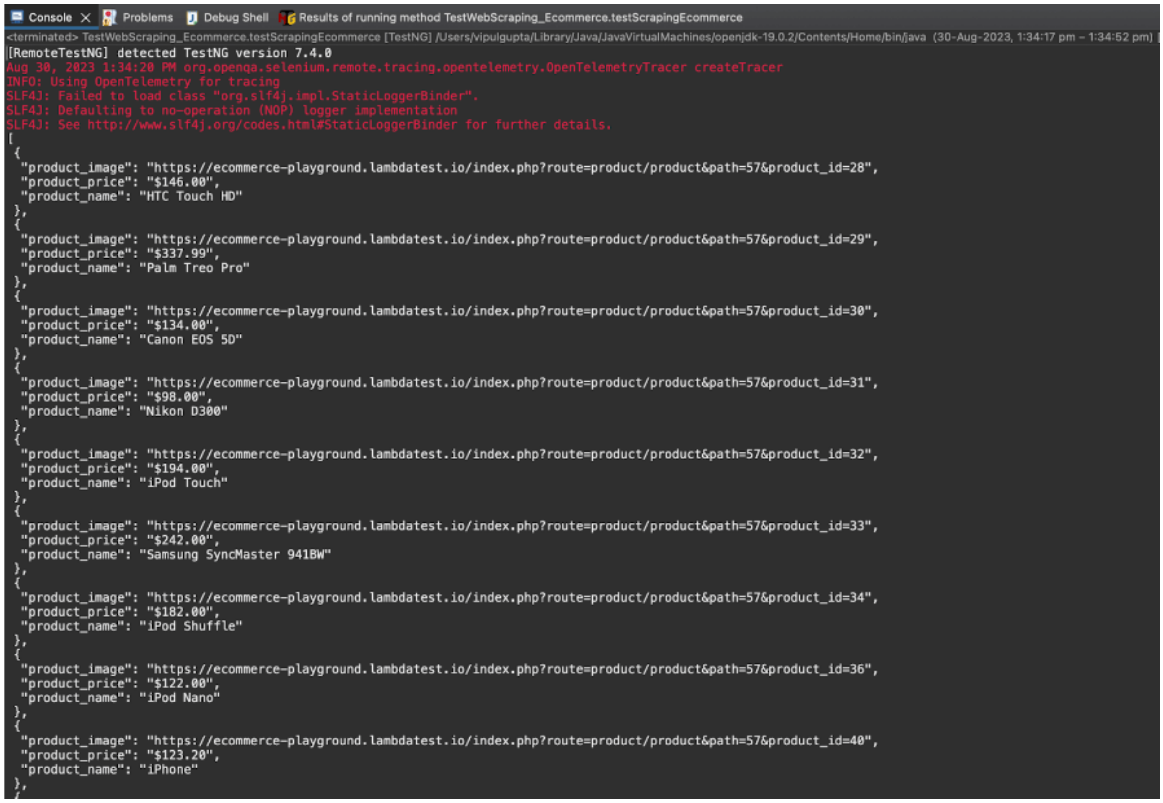
Select the Product



Select the Product Name



Select the Product Price



Console Output