

How to Use Web Scraper Chrome Extension to Extract Data

Jimna Jayan

June 12, 2024

Blog, Web Scraping

How to Use Web Scraper Chrome Extension to Extract Data

Web scraping is becoming a vital ingredient in business and marketing planning regardless of the industry. There are several ways to crawl the web for useful data depending on your requirements and budget. Did you know that your favorite web browser could also act as a great [web scraping tool](#)?

You can install the [Web Scraper extension from the chrome web store](#) to make it an easy-to-use data scraping tool. The best part is, that you can stay in the comfort zone of your browser while the scraping happens. This doesn't demand many technical skills, which makes it a good option when you need to do some quick data scraping. Let's get started with the tutorial on how to use the web scraper chrome extension to extract data.

About the Web Scraper Chrome Extension

[Web Scraper](#) is a [web data extractor extension](#) for [chrome](#) browsers made exclusively for web data scraping. You can set up a plan (sitemap) on how to navigate a website and specify the data to be extracted. The scraper will traverse the website according to the setup and extract the relevant data. It lets you export the extracted data to CSV.

Multiple pages can be scraped using the tool, making it even more powerful. It can even extract data from dynamic pages that use Javascript and Ajax.

What You Need

- Google Chrome browser
- A working internet connection

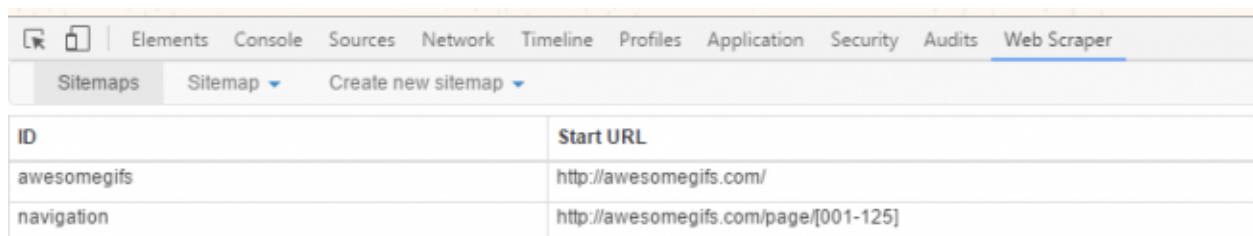
A. Installation and setup

- Web scraper chrome extension by using the link.
- For web scraper chrome extension download click on “Add”

Once this is done, you are ready to start scraping any website using your chrome browser. You just need to learn how to perform the scraping, which we are about to explain.

B. The Method

After installation, open the Google Chrome developer tools by pressing F12. (You can alternatively right-click on the screen and select inspect element). In the developer tools, you will find a new tab named ‘Web scraper’ as shown in the screenshot below.



Sitemaps	
Sitemap ▾ Create new sitemap ▾	
ID	Start URL
awesomегifs	http://awesomегifs.com/
navigation	http://awesomегifs.com/page/[001-125]

Now let's see how to use this on a live web page. We will use a site called *www.awesomегifs.com* for this tutorial. This site contains gif images and we will crawl these image URLs using our web scraper.

Step 1: Creating a Sitemap

- Go to *https://www.awesomегifs.com/*
- Open developer tools by right-clicking anywhere on the screen and then selecting inspect
- Click on the web scraper tab in developer tools
- Click on 'create new sitemap' and then select 'create sitemap'
- Give the sitemap a name and enter the URL of the site in the start URL field.
- Click on 'Create Sitemap'

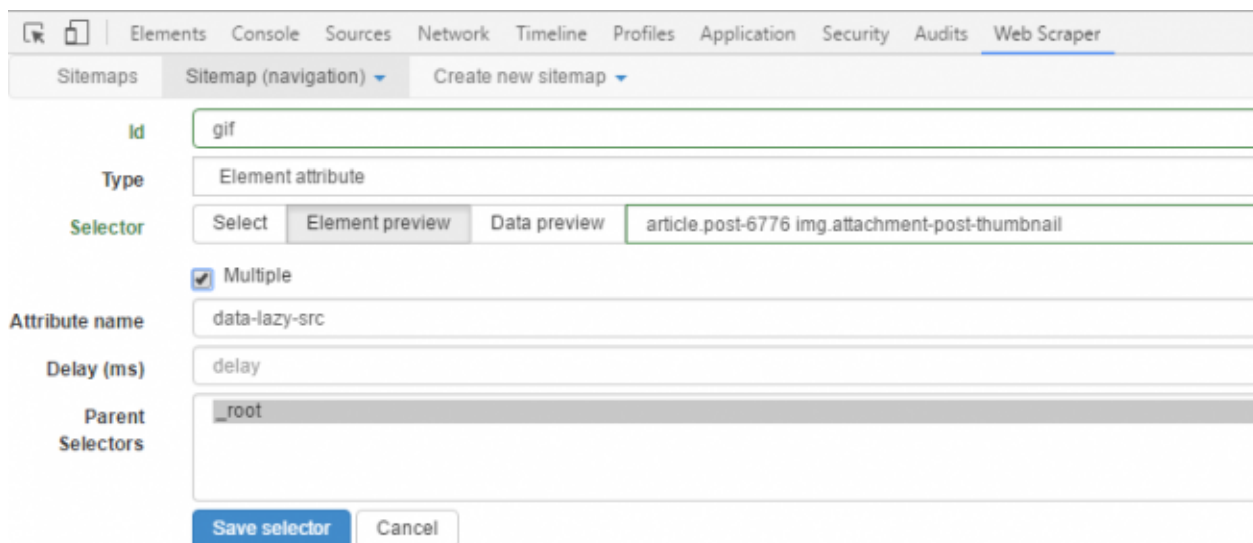
To crawl multiple pages from a website, we need to understand the pagination structure of that site. You can easily do that by clicking the 'Next' button a few times from the homepage. Doing this on *Awesomегifs.com* revealed that the pages are structured as *https://awesomегifs.com/page/1/*, *https://awesomегifs.com/page/2/*, and so on. To switch to a different page, you only have to change the number at the end of this URL. Now, we need the scraper to do this automatically.

To do this, create a new sitemap with the start URL as *https://awesomегifs.com/page/[001-125]*. The scraper will now open the URL repeatedly while incrementing the final value each time. This means the scraper will open pages starting from 1 to 125 and crawl the elements that we require from each page.

Step 2: Scraping Elements

Every time the scraper opens a page from the site, we need to extract some elements. In this case, it's the gif image URLs. First, you have to find the CSS selector matching the images. You can find the CSS selector by looking at the source file of the web page (CTRL+U). An easier way is to use the selector tool to click and select any element on the screen. Click on the Sitemap that you just created, and click on 'Add new selector'.

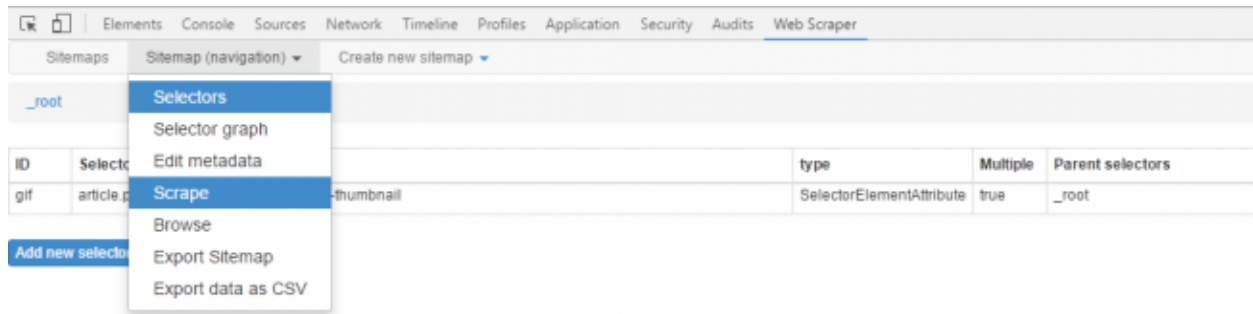
In the selector id field, give the selector a name. In the type field, you can select the type of data that you want to be extracted. Click on the select button and select any element on the web page that you want to be extracted. When you are done selecting, click on 'Done selecting'. It's easy as clicking on an icon with the mouse. You can check the 'multiple' checkbox to indicate that the element you want can be present multiple times on the page and that you want each instance of it to be scrapped.



The screenshot shows the 'Web Scraper' tab in a browser's developer tools. The 'Sitemaps' section is active, showing a 'Sitemap (navigation)' dropdown and a 'Create new sitemap' button. Below this, the 'Add new selector' form is visible. The form has the following fields and options:

- Id:** A text input field containing 'gif'.
- Type:** A dropdown menu set to 'Element attribute'.
- Selector:** A section with three tabs: 'Select', 'Element preview', and 'Data preview'. The 'Select' tab is active, showing a green box around the text 'article.post-6776 img.attachment-post-thumbnail'.
- Multiple:** A checkbox labeled 'Multiple' that is checked.
- Attribute name:** A text input field containing 'data-lazy-src'.
- Delay (ms):** A text input field containing 'delay'.
- Parent Selectors:** A list box containing '_root'.
- Buttons:** At the bottom, there are two buttons: 'Save selector' (blue) and 'Cancel' (white).

Now you can save the selector if everything looks good. To start the scraping process, just click on the sitemap tab and select 'Scrape'. A new window will pop up which will visit each page in the loop and crawl the required data. If you want to stop the data scraping process in between, just close this window and you will have the data that was extracted till then.



Once you stop scraping, go to the sitemap tab to browse the extracted data or export it to a CSV file. The only downside of such data extraction software is that you have to manually perform the scraping every time since it doesn't have many automation features built in.

If you want to crawl data on a large scale, it is better to go with a [data scraping service](#) instead of such free web scraper chrome extension data extraction tools like these. In the [second part](#) of this series, we will show you how to make a MySQL database using the extracted data. Stay tuned for that!

Frequently Asked Questions (FAQs)

How do I use Google Chrome Web scraper?

Using a web scraper in Google Chrome typically involves utilizing browser extensions designed for scraping tasks. These extensions can simplify the process of extracting data from websites without needing to write any code. Here's a general guide on how to use a basic web scraping extension in Google Chrome. While specific features might vary depending on the extension you choose, the overall process remains similar.

Step 1: Choose and Install a Web Scraping Extension

1. Find a Web Scraper Extension: Open the Google Chrome Web Store and search for web scraping extensions. Some popular options include Web Scraper (web-scraper.io) and Data Miner.
2. Install the Extension: Choose an extension that suits your needs, click on “Add to Chrome”, and then click “Add extension” in the popup to install it.

Step 2: Open the Target Website

- Navigate to the website you want to scrape in Google Chrome. Make sure the content you want to scrape is visible on the page.

Step 3: Launch the Web Scraper

- Click on the extension icon in the Chrome toolbar to open its interface. If it's your first time using the extension, there might be a tutorial or introduction. It's beneficial to go through this to understand the tool's features.

Step 4: Create a New Sitemap

- A sitemap within web scraping context is essentially a plan that tells the scraper what pages to scrape and what data to collect.
- Depending on the extension, you'll either select “Create new sitemap” or a similar option. You may need to give it a name and optionally the starting URL (the page you're currently on).

Step 5: Select Data to Scrape

- You'll then enter the selection phase, where you can click on elements of the webpage you want to scrape. This could include text, links, images, etc.
- As you select elements, the extension might offer options to refine your selection, ensuring you're capturing the right data. You can specify if you're collecting text, URLs, or other attributes.

Step 6: Define Data and Patterns

- For complex pages or to capture multiple items (like a list of products), you might need to define patterns or use the tool's pattern detection to ensure it recognizes similar elements across the page or multiple pages.

Step 7: Run the Scraper

- Once you've defined what data to scrape and where to find it, run the scraper. The extension will navigate the pages and collect the data according to your sitemap.

Step 8: Export the Data

- After the scraper completes its task, you can usually export the collected data in various formats, such as CSV or Excel, for further analysis or use.

Does Google ban web scraping?

Google's Stance on Scraping:

Google's Terms of Service do not explicitly mention "web scraping," but they include clauses that prohibit automated access to their services without permission. For example, the terms may restrict the use of robots, spiders, or scraping tools to access or extract data from their services. The intention here is to prevent excessive use of resources, protect against spam and abuse, and ensure the security and privacy of its users' data.

Detection and Enforcement:

Google employs various detection mechanisms to identify and block behavior it deems abusive or against its terms of service. This includes:

- Rate Limiting: Implementing rate limits on how many requests an IP address can make in a certain timeframe.
- CAPTCHAs: Presenting challenges to verify whether the user is human.
- Blocking IP Addresses: Temporarily or permanently banning IP addresses that exhibit suspicious behavior.

Consequences of Violation:

If Google detects unauthorized scraping activity, it might temporarily block the offending IP addresses from accessing its services. In more severe cases, or if the scraping causes significant strain on Google's infrastructure or involves the extraction of sensitive or protected data, legal action could be taken.

Ethical and Legal Considerations:

While scraping public data for personal use or research might seem harmless, doing so without permission on a scale that impacts service availability or violates copyright laws can have legal repercussions. It's essential to:

- Review and adhere to the terms of service of the website.
- Ensure that your data collection methods do not harm the website's service or access protected or private data without consent.
- Consider the ethical implications of collecting and using scraped data, especially personal information.

What is web scraper extension?

A web scraper extension is a browser add-on designed to simplify the process of extracting data from web pages. These extensions are particularly useful for individuals and professionals who need to collect information from the internet without writing custom code for web scraping. Here's a closer look at what web scraper extensions do, how they work, and their typical features:

Functionality

- **Automated Data Extraction:** Web scraper extensions automate the process of collecting data from websites. Users can select specific data they wish to extract, such as product details, prices, contact information, or any textual content displayed on a webpage.
- **Point-and-Click Interface:** Most of these extensions provide a user-friendly interface that allows users to select the data they want to scrape simply by clicking on the elements within the web page.
- **Data Organization:** The extracted data can be compiled into structured formats like CSV, Excel, or JSON, making it easy to analyze, share, or import into other applications.
- **Pagination Handling:** Advanced scraper extensions can navigate through pagination, allowing for the extraction of data from multiple pages of search results or listings automatically.

How They Work

1. **Installation:** Users first add the extension to their browser from the browser's extension store or marketplace.
2. **Configuration:** Upon navigating to a target web page, the user activates the extension and selects the data they wish to extract. This often involves defining a "sitemap" or plan that outlines which pages to visit and what data to collect.
3. **Data Selection:** The user typically enters a point-and-click mode where they can select specific page elements from which data should be extracted. The extension may offer options to refine the selection to ensure accuracy.
4. **Running the Scraper:** With the data points and pages defined, the user instructs the extension to start scraping. The tool then automatically visits the pages and extracts the specified data.
5. **Exporting Data:** Once the scraping process is complete, the user can export the collected data into a preferred format for further use.

Is web scraping illegal?

****1. General Legality:**

- Depends on Context: The legality of web scraping depends on various factors, including the website's terms of service, the nature of the data being scraped, and how the scraped data is used.

****2. Terms of Service (ToS):**

- Website Policies: Many websites have terms of service that explicitly prohibit scraping. Violating these terms can result in legal action or being banned from the site.

****3. Intellectual Property:**

- Copyright Issues: Scraping copyrighted content without permission can infringe on intellectual property rights. Using scraped content for commercial purposes without authorization can lead to legal consequences.

****4. Data Privacy:**

- Personal Data: Collecting personal data without consent can violate privacy laws such as GDPR (General Data Protection Regulation) in Europe or CCPA (California Consumer Privacy Act) in the United States. Sensitive personal information should be handled with utmost care.

****5. Ethical Considerations:**

- Respect and Fair Use: Ethical web scraping involves respecting the website's terms, data ownership, and user privacy. Scrapers should not overload servers or scrape data for malicious purposes.

****6. Legal Precedents:**

- Court Cases: There have been legal cases where web scraping was challenged in court. For example, in the case of *hiQ Labs, Inc. v. LinkedIn Corporation*, the court ruled that scraping publicly accessible data is not necessarily a violation of

the Computer Fraud and Abuse Act (CFAA). However, each case can have different outcomes based on specific circumstances.

****7. Practical Tips:**

- **Check Terms of Service:** Always review the terms of service of the website you intend to scrape.
- **Seek Permission:** When in doubt, seek permission from the website owner before scraping.
- **Respect Robots.txt:** Adhere to the guidelines specified in the `robots.txt` file of the website.
- **Use Data Responsibly:** Ensure that the scraped data is used in a manner that respects privacy and intellectual property laws.

In summary, while web scraping is not inherently illegal, it can become illegal depending on how it is performed and what data is being scraped. It is crucial to be aware of and comply with legal and ethical guidelines to avoid potential legal issues.

<https://www.promptcloud.com/blog/how-to-scrape-data-with-web-scraper-chrome/>