Introduction to Web Scraping

What is Web Scraping?

- **Web Scraping** is the automated process of extracting information from websites.
- Instead of manually copying data, a scraper **reads** and **parses** the webpage's code to collect the needed data.
- Used widely in data science for collecting:
  - Product prices
  - News articles
  - Job listings
  - Research datasets

Why Use Web Scraping?

- **Automate** repetitive data collection tasks
- Access data that is **not available via APIs**
- Build datasets for **Machine Learning models** and **Analytics**
- Monitor websites for **price changes**, **content updates**, or **news alerts**

When Not to Use Web Scraping

- If the site offers an **official API**, prefer that (it's cleaner and more stable).
- If scraping **violates** the site's **Terms of Service**.
- If the scraping **harms** the website (excessive requests can cause server overload).

HTML Basics for Web Scraping

What is HTML?

- **HTML (HyperText Markup Language)** structures content on the web.
- It's made up of **elements** (tags) like `<div>`, `<p>`, `<h1>`, `<a>`, etc.

Example:

```
<html>
  <body>
    <h1>Product Title</h1>
    <p class="price">$29.99</p>
    <a href="/buy-now">Buy Now</a>
  </body>
</html>
```

Important HTML Elements for Scraping

Tag	Meaning	Common Use
`<div>`	Division/Container	Group content
`<p>`	Paragraph	Text blocks
`<h1>`, `<h2>`, etc.	Headings	Titles and sections
`<a>`	Anchor (links)	URLs, navigation
`<img>`	Image	Pictures and icons
`<table>`	Table	Structured tabular data

Attributes Matter!

- HTML tags often have **attributes** like `id`, `class`, `href`, `src`.
- We use these attributes to **target** the correct elements.

Example:

```
<p class="price">$29.99</p>
```

Here, the `class="price"` attribute helps identify the price on the page.

Quick CSS Basics for Scraping

What is CSS?

- **CSS (Cascading Style Sheets)** controls the style and layout of HTML elements.
- For scraping, we mainly care about **CSS selectors** to **find and extract** data.

Common CSS Selectors

Selector	Meaning	Example
`.class`	Selects elements by class	`.price`, `.title`
`#id`	Selects an element by id	`#main`, `#product-title`
`tag`	Selects all elements of a tag	`h1`, `div`, `p`
`tag.class`	Selects a tag with class	`p.price`, `div.container`

Example: Selecting Elements

HTML:

```
<p class="price">$29.99</p>
```

CSS selector to target this:

```
p.price
```

In Python using BeautifulSoup:

```
soup.select_one("p.price")
```

Tools We'll Use for Web Scraping

- **`requests`** → To download the HTML content of a webpage.
- **`BeautifulSoup`** → To parse and extract data from the HTML.
- **(Optional)** **`Selenium`** → For websites that load data dynamically with JavaScript.

Summary

- Web scraping automates data extraction from websites.
- Understanding basic **HTML structure** and **CSS selectors** is crucial.
- Python provides powerful libraries to make web scraping easy.