

Puppeteer Web Scraping Documentation

Introduction

This documentation provides a guide on how to use the Puppeteer web scraping project to extract restaurant information from Google Maps. The project utilizes Puppeteer, Puppeteer Extra, Stealth Plugin, and Cheerio to navigate through Google Maps, search for restaurants in a specific location, extract restaurant links, and gather information about each restaurant.

Installation

To use this project, you need to have Node.js installed on your machine. You can install the required dependencies using npm or yarn:

```
npm install puppeteer puppeteer-extra puppeteer-extra-plugin-stealth cheerio
```

Usage

To use the web scraping functionality, simply call the **searchGoogleMaps()** function:

```
searchGoogleMaps();
```

Features

- Accepts Google Maps terms and conditions automatically.
- Searches for restaurants in a specified location.
- Extracts links to individual restaurant pages.
- Gathers information from each restaurant's page, such as website.

Code Overview

The code consists of the following main components:

1. **Initialization:** Initializes Puppeteer, Puppeteer Extra, and Stealth Plugin.
2. **Navigating to Google Maps:** Accepts terms and conditions and navigates to the search page.
3. **Auto Scrolling:** Scrolls the page to load all restaurant entries dynamically.

4. **Extracting Restaurant Links:** Extracts links to individual restaurant pages from the search results.
5. **Processing Each Restaurant Link:** Opens each restaurant's page and extracts information, such as the website.

Implementation Details

Initialization

```
const puppeteerExtra = require("puppeteer-extra"); const stealthPlugin = require("puppeteer-extra-plugin-stealth"); puppeteerExtra.use(stealthPlugin());
```

This section initializes Puppeteer Extra with the Stealth Plugin to ensure that the web scraping activities are undetectable.

Navigating to Google Maps

```
const browser = await puppeteerExtra.launch({ headless: false, executablePath: "", // Add your path here }); const page = await browser.newPage(); const query = "restaurants à proximité de Lille-Centre, Lille";
```

Launches a headless browser instance and opens a new page. It then defines a query for searching restaurants in a specific location.

Auto Scrolling

```
async function autoScroll(page) { // Function definition } await autoScroll(page);
```

Defines a function to automatically scroll the page to load all restaurant entries dynamically. It ensures that all restaurant links are loaded before extraction.

Extracting Restaurant Links

```
const html = await page.content(); const $ = cheerio.load(html); const restaurantLinks = [];  
$('a[href*="/maps/place/"]').each((i, el) => { // Extracting links });
```

Loads the HTML content of the page, parses it using Cheerio, and extracts links to individual restaurant pages based on the specified CSS selector.

Processing Each Restaurant Link

```
for (let i = 0; i < restaurantLinks.length; i++) { await processRestaurantLink(browser,  
restaurantLinks[i]); }
```

Iterates through each extracted restaurant link and processes it by opening the page and extracting information.

Conclusion

The Puppeteer web scraping project offers a convenient way to extract restaurant information from Google Maps. By following the provided documentation, developers can easily integrate this functionality into their projects and customize it as needed.