Scraper Command Documentation

Command: go_to

Navigate to a specific URL or path on the website.

Example:
```
{ "action": "go_to", "target": "https://example.com" }
```

Command: extract

Extract a single element's text content from the page.

Example:
```
{ "action": "extract", "selector": "//h1", "store_as": "title" }
```

Command: extract_multi

Extract multiple elements' text content as a list.

Example:
```
{ "action": "extract_multi", "selector": "//ul/li", "store_as": "items" }
```

Command: save

Save the scraped data into a file.

Example:
```
{ "action": "save", "filename": "output.json", "description": "Store results" }
```

Command: click

Click an element on the page (like a button or link).

Example:
```
{ "action": "click", "selector": "//button[@id='next']" }
```

Command: scroll

Scroll the page to load more content.

Example:
```
{ "action": "scroll", "direction": "down", "times": 3 }
```

Command: hover

Hover over an element to reveal hidden content.

Example:
```
{ "action": "hover", "selector": "//div[@class='menu']" }
```

Command: download

Download a file from a link on the page.

Example:
```
{ "action": "download", "selector": "//a[@id='report']", "filename": "report.pdf" }
```

Command: eval

Run custom JavaScript code in the browser context.

Example:
```
{ "action": "eval", "script": "document.title", "store_as": "page_title" }
```

Command: if

Run conditional steps only if a selector exists.

Example:
```
{ "action": "if", "selector": "//button[text()='Accept']", "steps": [ { "action":
"click", "selector": "//button[text()='Accept']" } ] }
```

Command: auth

Handle login by filling username, password and submitting form.

Example:
```
{ "action": "auth", "username": "myuser", "password": "mypassword",
"username_selector": "#username", "password_selector": "#password",
"submit_selector": "#loginBtn" }
```

Command: extract_attr

Extract an attribute value (like href or src) instead of text.

Example:
```
{ "action": "extract_attr", "selector": "//img", "attribute": "src", "store_as":
"images" }
```

Command: random_wait

Pause execution for a random time interval.

Example:
```
{ "action": "random_wait", "min": 2, "max": 6 }
```