

Screen Scraping Guide To Get Started On Your Own Project Using Ruby

Screen scraping is a powerful technique that allows you to extract data from websites. It's a common technique used for a variety of purposes, such as data mining, web research, and web automation. In this guide, we'll provide a comprehensive to screen scraping using Ruby, a popular programming language for web development.



Write Your Own Screen Scraper: A screen scraping guide to get started on your own project using Ruby and Nokogiri with examples

★★★★☆ 4 out of 5

Language	: English
File size	: 1239 KB
Text-to-Speech	: Enabled
Screen Reader	: Supported
Enhanced typesetting	: Enabled
Print length	: 12 pages
Lending	: Enabled



Prerequisites

- Basic understanding of Ruby programming language
- A text editor or IDE
- A web browser

Step 1: Choose a Ruby Gem

The first step in screen scraping with Ruby is to choose a Ruby gem. A Ruby gem is a reusable library of Ruby code that provides specific functionality. There are several Ruby gems available for screen scraping, but the most popular one is Nokogiri.

Nokogiri is a powerful HTML and XML parsing library that makes it easy to navigate and extract data from web pages. To install Nokogiri, run the following command in your terminal:

```
gem install nokogiri
```

Step 2: Make an HTTP Request

Once you have installed Nokogiri, you can start making HTTP requests to web pages. An HTTP request is a message that a client (such as your Ruby program) sends to a server (such as a website) to request data. To make an HTTP request in Ruby, you can use the `Net::HTTP` library.

```
require 'net/http'
```

```
uri = URI('https://example.com') res =  
Net::HTTP.get_response(uri)
```

The `Net::HTTP.get_response` method returns an HTTP response object. The response object contains the status code of the request, the headers of the response, and the body of the response. You can access the body of the response using the `body` attribute.

Step 3: Parse HTML

Once you have an HTTP response, you can start parsing the HTML. HTML is a markup language that describes the structure of a web page. To parse HTML in Ruby, you can use the Nokogiri library.

```
require 'nokogiri'
```

```
doc = Nokogiri::HTML(res.body)
```

The `Nokogiri::HTML` method parses the HTML in the response body and creates a document object. The document object represents the structure of the web page. You can use the document object to navigate and extract data from the web page.

Step 4: Extract Data

Once you have parsed the HTML, you can start extracting data from the web page. You can use the document object to access the different elements of the web page, such as the title, the headings, and the paragraphs.

```
title = doc.title headings = doc.css('h1, h2, h3') paragraphs  
= doc.css('p')
```

You can use the `title` attribute to access the title of the web page. You can use the `css` method to access the elements of the web page that match a CSS selector. The `css` method returns an array of elements. You can use the `text` attribute to access the text content of an element.

In this guide, we've provided a comprehensive to screen scraping using Ruby. We've shown you how to choose a Ruby gem, make an HTTP

request, parse HTML, and extract data from a web page. With this knowledge, you can start building your own screen scraping projects.

Additional Resources

- [Net::HTTP Documentation](#)
- [Nokogiri Tutorial](#)
- [Web Scraping with Ruby](#)

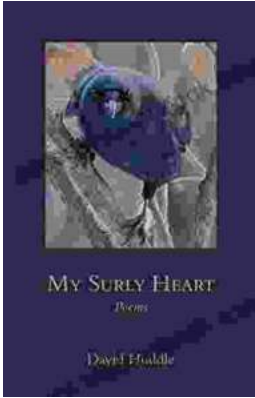


Write Your Own Screen Scraper: A screen scraping guide to get started on your own project using Ruby and Nokogiri with examples

★★★★☆ 4 out of 5

Language : English
File size : 1239 KB
Text-to-Speech : Enabled
Screen Reader : Supported
Enhanced typesetting : Enabled
Print length : 12 pages
Lending : Enabled





My Surly Heart: Poetic Expressions of Unrequited Love from Southern Messenger Poets

In the annals of American literature, the Southern Messenger holds a prominent place as a crucible where some of the most talented poets of the 19th...



Bleach Vol. 50: The Six Fullbringers - A Comprehensive Review

Bleach Vol. 50, titled "The Six Fullbringers," is the 50th installment in the acclaimed Bleach manga series by Tite Kubo. Released in 2010, this volume marks a significant...