All Collections > About Robots > Web Data Extraction Basics

Web Data Extraction Basics

Understand the basic web technologies used for extracting data

Written by Henrik Hofmeister
Updated over a week ago

Extracting data from most simple web pages can be done via the Extractor robot editor without any technical knowledge. Using the point-and-click interface, you simply point to the information you wish you extract, specify any required formatting and choose to which output field the information should be saved.

Read more about how to build Extractor robots:

- How to build an extractor
- What should I know about the extractor editor?
- What should I know about site navigation?

Some web pages, however, can require knowledge of web technologies like HTML, CSS selectors and JavaScript. Read on for details.

Element Paths (CSS Selectors)

Some web pages require you to know how to navigate the structure of the HTML page, the so-called *Document Object Model* (DOM), to find the element (also called *tag*) that holds the information you want. The typical way to navigate the DOM is by using *CSS selectors*, called *element paths* in Dexi.

For instance, to extract a price from a very simple HTML page:

```
 <html>

  <head>

    <meta charset="utf-8">

  </head>

  <body>

    <div>

      <h1>Price</h1>

      <p>$9.99</p>

    </div>

  </body>

</html>
```

... you could use the element path: `div > p`

In the Extractor robot editor, this can be used in e.g. an “Extract value” step to extract the price.

For other ways to find elements, see What should I know about elements, paths, and scopes?.

For general information on HTML, the DOM and CSS selectors, we refer you to a vast number of articles and tutorials available online. A couple of useful resources, we think, are:

- W3Schools - HTML Element Reference
- W3Schools - HTML Global Attributes
- JavaScript.info - DOM Tree
- MDN - CSS Selectors
- W3Schools - CSS Selectors Reference
- jsoup - online CSS selector tester

Dexi uses CSS version 3.

Robust Element Paths

Writing a good element path sometimes takes a bit of consideration: the more general/”wide” you make it, the more robust it is to web page changes but it also decreases the likelihood of finding the exact information you wish to extract.

As an example consider the following HTML snippet:

```
<div>

  <span>

    <div>

      <input type=”text” name=”username” id=”username-1298172391617”>

      <input type=”text” name=”password” id=”password-891291767394”>

    </div>

  </span>

</div>
```

An example of a robust element path would be:

`input[name="username"]`

...because:

1. It points to an element that most likely will continue to exist on the page.
2. It does not depend on changes to the structure of the page.

An example of a not-so-robust element path would be:

`div > span > div > input#username-1298172391617`

...because:

1. The id looks like a dynamic number that could easily change.
2. It is very dependent on the exact current structure of the page, e.g. if one of the `<div>` elements changes to a `<span>`, the element path is no longer valid.

When selecting elements in the Extractor editor, an element path is automatically generated. To make it more robust it is sometimes a good idea to manually change it using the considerations mentioned above.

JavaScript

Most modern web pages use JavaScript (JS) to some extent - and some make heavy use of it. It is used to make pages interactive and dynamic, allowing users to interact with the page and for the page to automatically load content, change appearance, etc.

Dexi fully supports JavaScript and is able to load complex pages e.g. with calendars, menus and much more on pages using technologies such as React and AngularJS (specific Extractor step types for extracting from the AngularJS model are available).

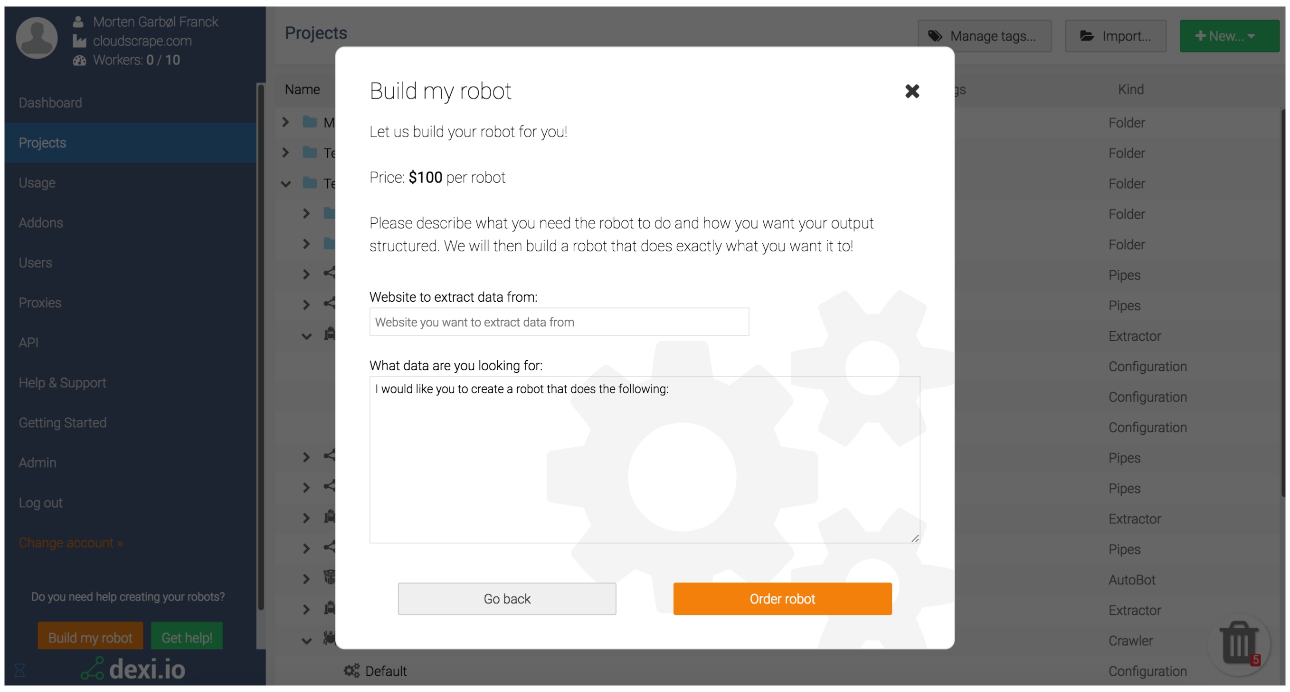
It is also possible to execute arbitrary JS code such as encoding/decoding data, using dates and accessing mathematical functions, just to name a few.

“Just get me the data, please”

If you are not technically inclined or if you perhaps don’t have the time to learn web technologies, we offer the build the robot for you. Simply tell us which information you want from which web page(s) and we build the necessary robot(s).

To request a robot build, please see our Robot Building page.

You can also log in to the platform and click the “Build my robot” button in the bottom left corner:



Build my robot

Let us build your robot for you!

Price: **$100** per robot

Please describe what you need the robot to do and how you want your output structured. We will then build a robot that does exactly what you want it to!

Website to extract data from:

Website you want to extract data from

What data are you looking for:

I would like you to create a robot that does the following:

If you need any other help, please write us at support@dexi.io.

Thank you for reading and enjoy dexi.io!

Did this answer your question?

😞 😐 😃