

A large, solid pink circle is centered on the slide, serving as a background for the main title text.

DATA SCIENCE BOOTCAMP

Session 7: Introduction to Web Scraping and API

SESSION 7 OUTLINE

- **WEB SCRAPING**
- **HTML**
- **APPLICATION PROGRAMMING INTERFACE (API)**

SESSION 7 OUTLINE

- **WEB SCRAPING**
- **HTML**
- **APPLICATION PROGRAMMING INTERFACE (API)**

WEB SCRAPING

Web scraping is the process of using bots to extract content and data from a website.

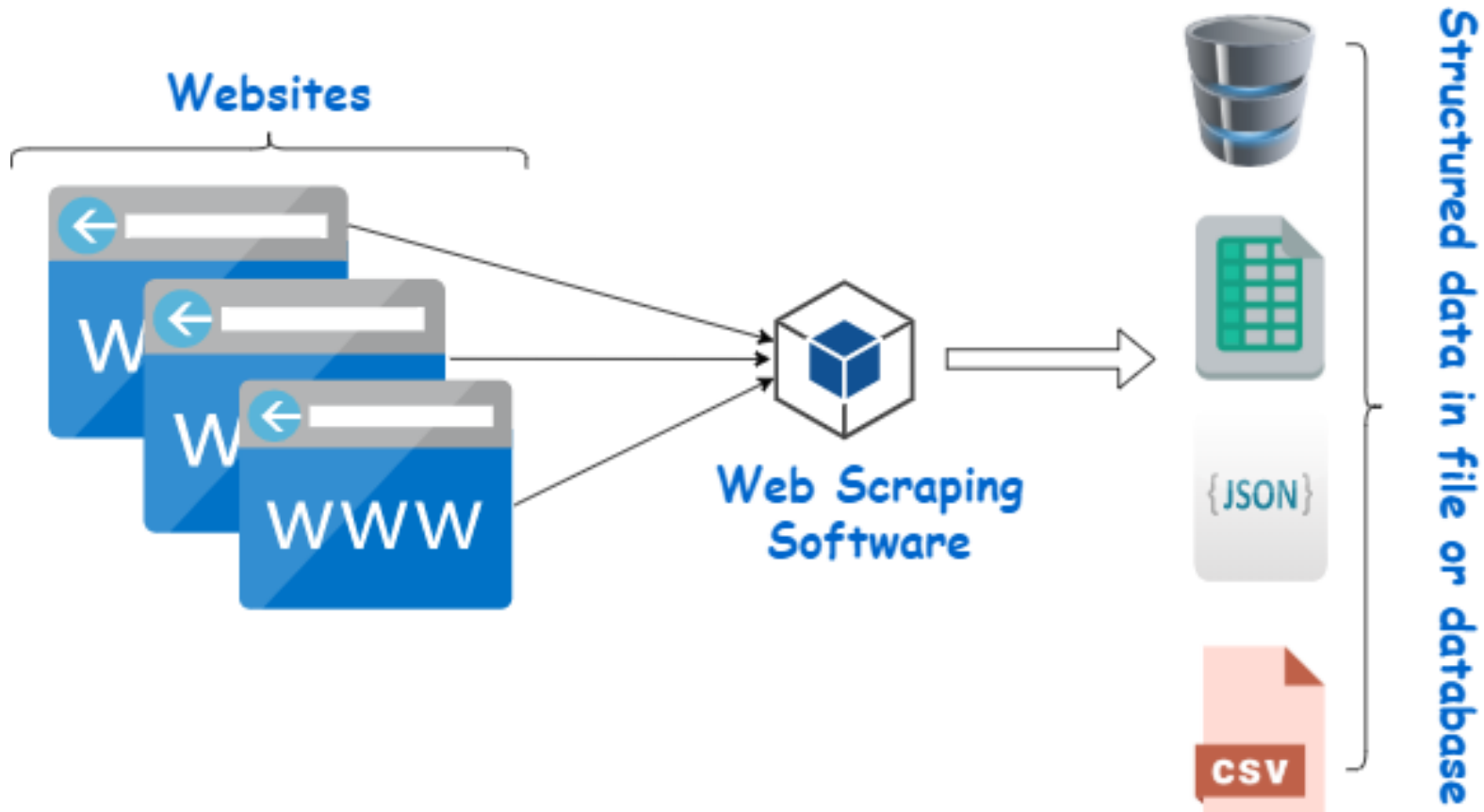
It extracts underlying HTML code and, with it, data stored in a database. The scraper can then replicate entire website content elsewhere.

WEB SCRAPING

Web scraping is the process of using bots to extract content and data from a website.

It **extracts underlying HTML code** and, with it, data stored in a database. The scraper can then **replicate entire website content** elsewhere.

WEB SCRAPING



SESSION 7 OUTLINE

- **WEB SCRAPING**
- **HTML**
- **APPLICATION PROGRAMMING INTERFACE (API)**

HTML

HTML is the standard markup language for Web pages.

With HTML you can create your own Website.

HTML

HTML is the standard **markup language** for **Web pages**.

With HTML you can create your own Website.

HTML Example

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

HTML Example - Headings

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Heading 1

Heading 2

Heading 3

Heading 4

Heading 5

Heading 6

`<h1>Heading 1</h1>`

`<h2>Heading 2</h2>`

`<h3>Heading 3</h3>`

`<h4>Heading 4</h4>`

`<h5>Heading 5</h5>`

`<h6>Heading 6</h6>`

HTML Example - Paragraph

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

This is a paragraph.

This is a paragraph.

```
<p>This is a paragraph.</p>
```

```
<p>This is another paragraph.</p>
```

HTML Tables

```
<table>
  <tr>
    <th>Company</th>
    <th>Contact</th>
    <th>Country</th>
  </tr>
  <tr>
    <td>Alfreds Futterkiste</td>
    <td>Maria Anders</td>
    <td>Germany</td>
  </tr>
  <tr>
    <td>Centro comercial Moctezuma</td>
    <td>Francisco Chang</td>
    <td>Mexico</td>
  </tr>
</table>
```

Company	Contact	Country
Alfreds Futterkiste	Maria Anders	Germany
Centro comercial Moctezuma	Francisco Chang	Mexico

HTML Lists

```
<ul>
```

```
  <li>Coffee</li>
```

```
  <li>Tea</li>
```

```
  <li>Milk</li>
```

```
</ul>
```

- Coffee
- Tea
- Milk

```
<ol>
```

```
  <li>Coffee</li>
```

```
  <li>Tea</li>
```

```
  <li>Milk</li>
```

```
</ol>
```

1. Coffee
2. Tea
3. Milk

HTML Block and Inline

```
<p>Hello World</p>
```

Hello World

```
<div>Hello World</div>
```

Hello World

```
<span>Hello World</span>
```

This is an inline span Hello World element inside a paragraph.

HTML Attributes

```
<a href="https://www.w3schools.com">Visit W3Schools</a>
```

[Visit W3Schools](https://www.w3schools.com)

HTML Id

```
<!DOCTYPE html>
<html>
<head>
<style>
#myHeader {
  background-color: lightblue;
  color: black;
  padding: 40px;
  text-align: center;
}
</style>
</head>
<body>

<h1 id="myHeader">My Header</h1>

</body>
</html>
```

The id Attribute

Use CSS to style an element with the id "myHeader":



My Header

HTML Classes

```
<!DOCTYPE html>
<html>
<head>
<style>
.note {
  font-size: 120%;
  color: red;
}
</style>
</head>
<body>
```

```
<h1>My <span class="note">Important</span> Heading</h1>
<p>This is some <span class="note">important</span> text.</p>

</body>
</html>
```

My **Important** Heading

This is some **important** text.

SESSION 7 OUTLINE

- **WEB SCRAPING**
- **HTML**
- **APPLICATION PROGRAMMING INTERFACE (API)**

API

API stands for **Application **P**rogramming **I**nterface.**

In the context of APIs, the word Application refers to any software with a distinct function. Interface can be thought of as a contract of service between two applications. This contract defines how the two communicate with each other using requests and responses.

API

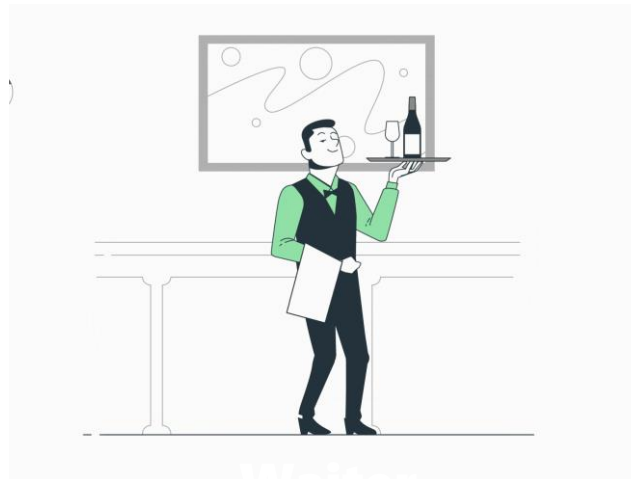
How API works



API



**FRONT-END
(APPLICATION)**



**API
(ENDPOINTS)**



**BACK-END
(SERVER/DATABASE)**