

# dom-serializer

build passing

Renders a [domhandler](#) DOM node or an array of domhandler DOM nodes to a string.

```
import render from "dom-serializer";  
  
// OR  
  
const render = require("dom-  
    serializer").default;
```

## render

Name	Type	Description	Default	Value	node	Node   Node[]	-	Node to be rendered.	Changes serialization behavior	options <a href="#">DomSerializerOptions</a> { }	Returns: string	
<b>Parameters:</b>												

Renders a DOM node or an array of DOM nodes to a string.  
Can be thought of as the equivalent of the outerHTML of the passed node(s).

- `render(node: Node | Node[], options?: Options): string`

## API

# Options

## decodeEntities

- **Optional decodeEntities:** *boolean*

Encode characters that are either reserved in HTML or XML, or are outside of the ASCII range.

default true

---

## emptyAttrs

- **Optional emptyAttrs:** *boolean*

Print an empty attribute's value.

default xmlMode

example With emptyAttrs:

false:<input checked>

example With emptyAttrs:

true:<input checked="">

---

## selfClosingTags

- **Optional selfClosingTags:** *boolean*

Print self-closing tags for tags without contents.

<code>domhandler</code>	Handler for <code>htmllparser2</code> that turns documents into a DOM
<code>domutils</code>	Utilities for working with domhandler's DOM
<code>css-select</code>	domhandler's DOM
<code>Name</code>	Description

## Ecosystem

Name	Description
<code>cheerio</code>	CSS selector engine, comparable with <code>domhandler</code> 's DOM
<code>dom-serializer</code>	Serializer for <code>domhandler</code> 's DOM
<code>LICENSE: MIT</code>	

<code>xmlMode</code>	<ul style="list-style-type: none"> <li>• <code>Optional xmlMode: boolean   "foreign"</code></li> </ul> <p>Treat the input as an XML document; enables the <code>emptyAttrs</code> and <code>selfClosingTags</code> options.</p> <p>If the value is <code>"foreign"</code>, it will try to correct mixed-case attribute names.</p> <p>default false</p>
----------------------	--