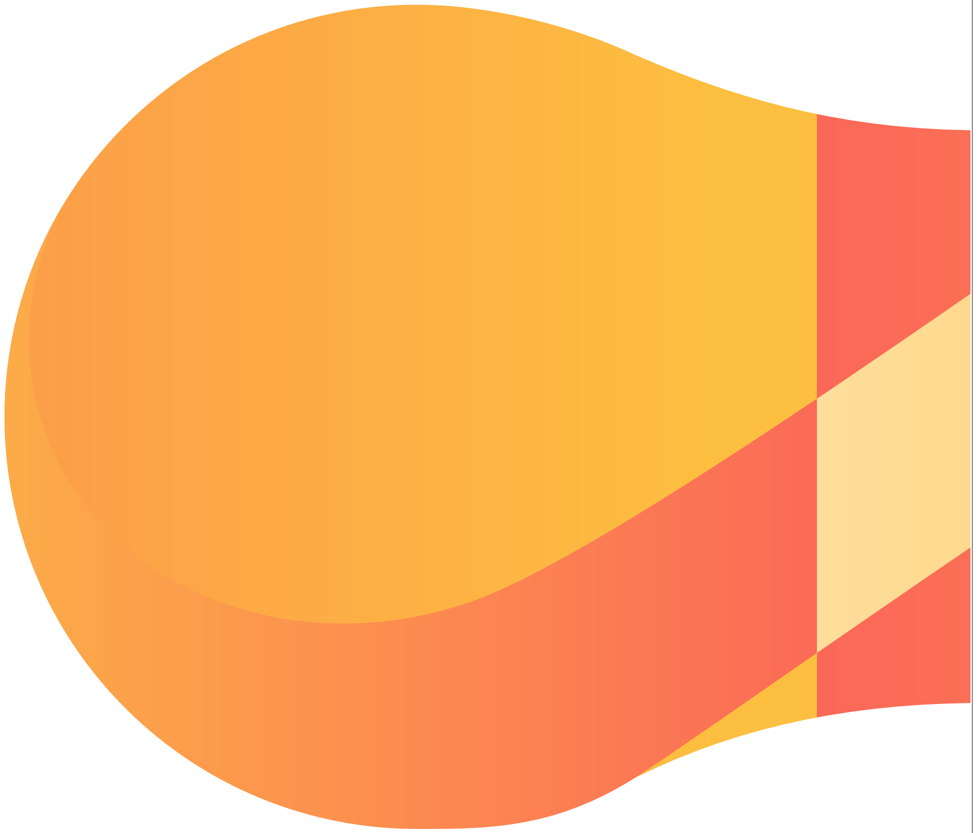

PostHTML Render

Renders a PostHTML Tree to HTML/XML

Install

```
npm i -D posthtml-render
```

Usage

NodeJS

```
import { render } from
        ''posthtml-render;

const tree = [];

const node = {};

node.tag = 'ul';
node.attrs = { class: 'list' };
node.content = [
 'one',
  'two',
  'three'
].map((content) => ({ tag: 'li',
        content }));
```

```
tree.push(node);
```

```
const html = render(tree, options);
```

```
<ul class="list">
  <li>one</li>
  <li>two</li>
  <li>three</li>
</ul>
```

Options

Name	Type	Default	Description
singleTags	{Array<String|RegExp>}	[]	Specify custom single tags (self closing)
closingSingleTag	{String}	>	Specify the single tag closing format
quoteAllAttributes	{Boolean}	true	Put double quotes around all tags, even

```
1
```

Attribute values are wrapped in single quote:

```
<img src='https://example.com/
        example.png'
        onload='testFunc("test")'>
```

```
0
```

Quote style is based on attribute values (an alternative for `replaceQuote` option):

```
<img src="https://example.com/
        example.png"
        onload='testFunc("test")'>
```

true (Default)

```html
<i src="index.js"></i>
```

false

```html
<i src=index.js></i>
```

replaceQuote

Replaces quotes in attribute values with `&quote;`.

true (Default)

```html
<img src="<?php echo
        $foo[&quote;bar&quote;] ?>">
```

false

```html
<img src="<?php echo $foo["bar"] ?>">
```

quoteStyle

2 (Default)

Attribute values are wrapped in double quotes:

```html
<img src="https://example.com/
        example.png"
        onload="testFunc("test")">
```

Name	Type	Default	Descriptio
			when not necessary.
`replaceQuote`	`{Boolean}`	`true`	Replaces quotes in attribute values with `&quote;`.
`quoteStyle`	`{0 or 1 or 2}`	`2`	Specify the style of quote arround the attribute values

singleTags

Specify custom single tags (self closing)

{String}

```js
const render = require('posthtml-
        render')

const tree = [ { tag: 'name' } ]
const options = { singleTags: [
        'name' ] }

const html = render(tree, options)
```

result.html

```
<name>
```

```
{RegExp}
```

```
const render = require('posthtml-
        render')

const tree = [ { tag: '%=title%' } ]
const options = { singleTags: [ /^%.*%
        $/ ] }

const html = render(tree, options)
```

result.html

```
<%=title%>
```

closingSingleTag

Specify the single tag closing format

```
Formats
```

```
const render = require('posthtml-
        render')

const tree = [ { tag: 'img' } ]
```

```
'tag'
```

```
const html = render(tree, {
        closingSingleTag: 'tag' })
```

```
<custom></custom>
```

```
'slash'
```

```
const html = render(tree, {
        closingSingleTag: 'slash' })
```

```
<custom />
```

```
'default' (Default)
```

```
const html = render(tree)
```

```
<img>
```

```
'closeAs'
```

```
const tree = [ {
  tag: 'custom',
  closeAs:
        'default' // Available types:
        `tag` | `slash` | `default`
} ]
const html = render(tree, {
        closingSingleTag: 'closeAs' })
```

```
<custom>
```

quoteAllAttributes

Specify if all attributes should be quoted.