

# Package: midas (via r-universe)

October 10, 2024

**Type** Package

**Title** Turn HTML 'Shiny'

**Version** 1.1.0

**Author** Nicholas Jhirad

**Maintainer** Nicholas Jhirad <lostnfound@gmail.com>

**Description** Contains functions for converting existing HTML/JavaScript source into equivalent 'shiny' functions. Bootstraps the process of making new 'shiny' functions by allowing us to turn HTML snippets directly into R functions.

**Imports** shiny (>= 1.0.3), xml2 (>= 1.1.1),

**Suggests** formatR (>= 1.5)

**BugReports** <https://github.com/shapenaji/midas/issues>

**License** GPL-3

**LazyData** true

**Encoding** UTF-8

**RoxygenNote** 6.0.1

**Repository** <https://shapenaji.r-universe.dev>

**RemoteUrl** <https://github.com/shapenaji/midas>

**RemoteRef** HEAD

**RemoteSha** 85dbc81056e020593c4e945a4673fc70f3daff87

## Contents

html_to_list . . . . .	2
midas_touch . . . . .	2
turn_shiny . . . . .	3

<b>Index</b>	<b>4</b>
--------------	----------

---

html_to_list	<i>Uses xml2 to parse html text into a list</i>
--------------	---

---

**Description**

Uses xml2 to parse html text into a list

**Usage**

```
html_to_list(x)
```

**Arguments**

x                    string of html

**Value**

html as an R list

**Examples**

```
# create some test data
html <- '<div class="example"><h3>test</h3></div>'
html_to_list(html)
```

---

midas_touch	<i>turns an xml2 list into a shiny function call</i>
-------------	--

---

**Description**

turns an xml2 list into a shiny function call

**Usage**

```
midas_touch(root)
```

**Arguments**

root                    an html list (from xml2 as\_list). Note: If the list is particularly deep, you may need to set option(expressions = SOMETHING BIG)

**Value**

a function call that produces the equivalent shiny objects

---

turn_shiny	<i>Turns a string of html into the equivalent shiny code</i>
------------	--

---

**Description**

Turns a string of html into the equivalent shiny code

**Usage**

```
turn_shiny(html, remove_newlines = TRUE, file = NULL)
```

**Arguments**

html	a string of complete html
remove_newlines	whether or not to remove newlines from the string
file	default (NULL) is to return the object, if this is set it will write to file

**Value**

a function call that produces the equivalent shiny objects

**Examples**

```
html <- '<div class="example"><h3>test</h3></div>'
turn_shiny(html)
eval(turn_shiny(html))
```

# Index

[html\\_to\\_list](#), 2

[midas\\_touch](#), 2

[turn\\_shiny](#), 3