# Web scraping

## Importing page

**minimal\_html(**"html code"**):** Lets you create an xml\_document from literal HTML.

**read\_html(**"url"**):** This returns a xml\_document2 object which you’ll then manipulate using rvest functions.

Logo

Description automatically generated with low confidence

Text

Description automatically generated

|  |  |
| --- | --- |
| Graphical user interface, text, application  Description automatically generated |  |

Graphical user interface, text, application

Description automatically generated

## Filtering the page

### rvest functions

html %>% **html\_children("**tag"**):** Selects all the children of a tag.

html %>%**html\_element(**"tag" or ".class\_name " or "#tag\_id"**):** Can select the elements of the first matching tag, class\_name or tag\_id. When applied to a node set (tag), always returns a vector the same length as the input, using a ***"missing"*** element where needed.

html %>% html\_element(‘ccs\_selector1, ccs\_selector2’ or ‘\*’)

html %>% **html\_elements(**"tag" or ".class\_name" or ".tag\_id"**):** Can select the elements of all matching tags, class\_names or tag\_ids. When applied to a node set (tag), returns all matching elements beneath any of the inputs, flattening results into a new node set.

### CSS selectors and combinators

|  |  |
| --- | --- |
| Text  Description automatically generated | Table  Description automatically generated  p:last-child  , == OR  .x.y == .x Y .y |

**space** **(descendant combinator**)**:** selects all the descendants of a certain HTML element, whether they are direct descendants or elements further down the tree.

Graphical user interface, text, application

Description automatically generated

**> (child combinator):** It only selects direct descendants of a parent element.

Graphical user interface, text, application, website

Description automatically generated

**+ (adjacent sibling combinator):** We can used to reference to the second sibling of a tag.

Graphical user interface, text, application

Description automatically generated

**~ (general sibling combinator):** We can used to reference to all siblings of a tag.

Graphical user interface, text, application

Description automatically generated

**\*:** Allow us to get all the siblings after div.first tag.

Graphical user interface, application, website

Description automatically generated

### XPATH

|  |  |
| --- | --- |
| **XPATH:** '*//***p**' | **XPATH:** '//**div**/**p**' |
| **meaning:** Selects all **p** tags | **meaning:** From all **div** selects direct child **p** |
|  |  |
| **XPATH:** ' //**div**[**p**[@class=**"third"**]] | **XPATH:** ' //**\***[@id=**"third"**]//**p**' |
| **meaning:** Select all **div** with **p** with class **third** | **meaning:** From all **tags** with id **third** select child **p** |
|  |  |
| **XPATH:** '//**div**[position() = 2]/**\***[position() >= 2]' | |
| **meaning:** From all **div** in second place selects direct child **tags** in second or grader place | |
|  | |
| **XPATH:** ''//**div**[count(**h2**) = 1 and count(**p**) > 1]'' | |
| **meaning:** Select all **div** with one **h2** and more than one **p** | |
|  | |
| **XPATH:** programming\_html %>%  html\_elements(xpath = '//**li**//**em**[text() = "twice"]') %>%  html\_elements(xpath = '..') | |
| **meaning:** From all **li** select child **em** with “twice” text inside then select parent | |
|  | |

## Extracting information

html %>% **html\_text()**: Shows the text as it was written in the source code.

html %>% **html\_text2()**: Shows the text in the way it would be displayed in the browser.

html %>% **html\_attr():** Gets a single attribute.

html %>% html\_elements("a") %>% html\_attr("href")

html %>% html\_elements("img") %>% html\_attr("src")

html %>% **html\_attrs():** Gets all attributes.

html %>% html\_node("table") %>% **html\_table()**: If we found a table tag we can extract its info.

* If the table has the *<th></th>* we should set **header=TRUE**

## Best practice

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

# Excel files

temp <- tempfile(fileext = ".xls")

download.file("http://www.systemicpeace.org/inscr/p5v2018.xls", destfile = temp, mode = 'wb')

Polity5 <- readxl::read\_xls(temp)