

# Package ‘scraEP’

November 15, 2017

**Type** Package

**Title** Scrape European Parliament Careers

**Version** 1.0

**Date** 2017-11-15

**Author** Julien Boelaert <jubo.stats@gmail.com>

**Maintainer** Julien Boelaert <jubo.stats@gmail.com>

**Description** A utility to webscrape the in-house careers of members of the European parliament, from its website <<http://www.europarl.europa.eu>>.

**License** GPL (>= 3)

**Imports** XML, RCurl

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2017-11-15 13:35:04 UTC

## R topics documented:

scraEP-package . . . . .	1
scraEP . . . . .	2
wiki . . . . .	3
xscrape . . . . .	3

<b>Index</b>	<b>6</b>
--------------	----------

---

scraEP-package	<i>Scrape the careers of all members of European parliament.</i>
----------------	--

---

## Description

Fetch all in-house career information of MEPs from the European parliament’s website, and put them in a data frame.

## Details

Package: scraEP  
 Type: Package  
 Version: 1.0  
 Date: 2017-11-15  
 License: GPL (>=3)

Function `scraEP` downloads and extracts career information from the EP's website.

Function `xscrape` is a general tool to extract information from html pages into data frames using XPath queries.

### Author(s)

Julien Boelaert <jubo.stats@gmail.com>

---

scraEP	<i>Extract all (in-house) career information of MEPs from the European parliament's website.</i>
--------	--

---

### Description

This function downloads all career information from the European parliament's website <<http://www.europarl.europa.eu>> or from local html files, and extracts it into a data frame.

### Usage

```
scraEP(local.html= NA, save.html= NA, max.try= 20)
```

### Arguments

<code>local.html</code>	a character string, indicating the local directory from which the MEP's html files are imported (existing files will be overwritten). If NA (default), the career information is downloaded from the EP's website.
<code>save.html</code>	a character string, indicating the local directory in which the downloaded html files should be saved. If NA (default), the downloaded files are not saved to local filesystem.
<code>max.try</code>	numeric: maximum number of tries to download html files from the EP's website.

### Details

Downloading all the pages can take a long time, be sure to store the result in an object.

### Value

A `data.frame`, where each row is a career step recorded on the EP's website.

**Author(s)**

Julien Boelaert <jubo.stats@gmail.com>

---

wiki	<i>Wikipedia page for R.</i>
------	------------------------------

---

**Description**

Toy example to extract data using xscrape.

**Usage**

```
data(wiki)
```

**Format**

Object `wiki` is a character vector.

**Details**

Object `wiki` is a raw webpage, containing the source of the ‘R (programming language)’ article on English wikipedia (<[https://en.wikipedia.org/wiki/R\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/R_(programming_language))>, retrieved on 15/11/2017).

**Author(s)**

Julien Boelaert <jubo.stats@gmail.com>

---

xscrape	<i>Extract information from webpages to a data.frame, using XPath queries.</i>
---------	--

---

**Description**

This function transforms an html page (or list of pages) into a `data.frame`, extracting nodes specified by their XPath.

**Usage**

```
xscrape(pages, col.xpath, row.xpath= "/html", collapse= " | ")
```

**Arguments**

pages	an object of class XMLDocument (as returned by function <code>htmlParse</code> ), or list of such objects. These are the webpages that information is be extracted from.
col.xpath	a character vector of XPath queries. Each element of this vector will be a column in the resulting data.frame. If the vector is named, these names are given to the result's.
row.xpath	(optional) a character string, containing an XPath query. This functions as an intermediary node: if specified, each result of this XPath query (on each page) becomes a row in the resulting data.frame. If not specified (default), the intermediary nodes are whole html pages, so that each page becomes a row in the result.
collapse	(optional) a character string, containing the separator that will be used in case a col.xpath query yields multiple results.

**Details**

If a col.xpath query designs a full node, only its text is extracted. If it designs an attribute (eg ends with `'/@href'` for weblinks), only the attribute's value is extracted.

If a col.xpath query matches no elements in a page, returned value is NA. If it matches multiple elements, they are concatenated into a single character string, separated by collapse.

**Value**

A data.frame, where each row corresponds to an intermediary node (either a full page or an XML node within a page, specified by row.xpath), and each column corresponds to the text of an col.xpath query.

**Author(s)**

Julien Boelaert <jubo.stats@gmail.com>

**Examples**

```
## Extract all external links and their titles from a wikipedia page
data(wiki)
wiki.parse <- XML::htmlParse(wiki)
links <- xscrape(wiki.parse,
                row.xpath= "//a[starts-with(./@href, 'http')]",
                col.xpath= c(title= ".", link= "./@href"))

## Not run:
## Convert results from a search for 'R' on duckduckgo.com
duck <- XML::htmlParse("http://duckduckgo.com/html?q=R")
results <- xscrape(duck,
                  row.xpath= "//div[contains(@class, 'result__body')]",
                  col.xpath= c(title= "./h2",
                               snippet= ".//*[ @class='result__snippet']",
                               url= ".//a[@class='result__url']/@href"))
```

*xscrape*

5

## End(Not run)

# Index

\*Topic **datasets**

wiki, [3](#)

\*Topic **package**

scraEP-package, [1](#)

scraEP, [2](#)

scraEP-package, [1](#)

wiki, [3](#)

xscrape, [3](#)