

Custom Keyboard Maestro Plug-in

NAME

- Harvest a default results list, or or custom list, from a Safari page.

VERSION

- 0.1

SYNOPSIS

- Finds a list of elements by xPath search pattern
- returns a text list in the specified string pattern
 - (Javascript string syntax)

REQUIREMENTS

- Yosemite
(The core script harvestList.sh is mainly written in Javascript for Applications)

OPTIONS

Type of page and list

- Sample presets
- Search result list with links
 - DuckDuckGo
 - Google
 - Yahoo
 - Bing
- Kindle notes list
 - Booklist from https://kindle.amazon.com/your_highlights
- Links to recent posts on a Discourse forum
- Custom list settings
- xpath
- javascript lambda for output lines
- Checkstring for front page url
- target url (or front page if blank)
- List description
- List title string
- On no match
 - empty list
 - notify

Output format

- Simple text lines
- JSON array string
- JSON of the property object

INSTALLATION

- Drag the .zip file onto the Keyboard Maestro icon in the OS X toolbar.
- (if updating a previous version of the action, first manually remove the previous copy from the custom actions folder)
 - ~/Library/Application Support/Keyboard Maestro/Keyboard Maestro Actions

EXAMPLES (Sample settings for particular web-sites)

Note that unless **custom** mode is chosen, the action will automatically detect the following websites, and use the following settings for each:

Essentially, the XPath expression specifies which element type to harvest, and the string expression (javascript syntax) specifies, as a function of x (the matched element) the format for each line.

In addition to the HTML attributes of each element, x will always have an x.text property, which will either be an DOM .text property, or derived from the .textContent property by replacing multiple spaces with single spaces.

Google search results

https://www.google.com/
XPath: //h3[@class='r']/a
λx: '- [' + x.text + '](' + x.href + ')'

Google Scholar search results

https://scholar.google.com/
XPath: //h3[@class='gs_rt']/a
λx: '- [' + x.text + '](' + x.href + ')'

Google Books search results

https://books.google.com/
XPath: //h3[@class='r']/a
λx: '- [' + x.text + '](' + x.href + ')'

DuckDuckGo search results

https://duckduckgo.com/
XPath: //h2[@class='result__title']/a
λx: '- [' + x.text + '](' + x.href + ')'

Yahoo search results

https://us.search.yahoo.com/
XPath: //h3[@class='title']/a[@class=' td-u']
λx: '- [' + x.text + '](' + x.href + ')'

Bing search results

https://www.bing.com/
XPath: //li/h2/a
λx: '- [' + x.text + '](' + x.href + ')'

JSTOR search results

http://www.jstor.org/
XPath: //div[@class='title']/a
λx: '- [' + x.text + '](http://www.jstor.org' + x.href + ')'

StackOverflow search results

http://stackoverflow.com/
XPath: //div[@class='result-link']/span/a
λx: '- [' + x.text + '](http://www.stackoverflow.com' + x.href + ')'

PubMed search results

http://www.ncbi.nlm.nih.gov/
XPath: //p[@class='title']/a
λx: '- [' + x.text + '](http://www.ncbi.nlm.nih.gov' + x.href + ')'

WorldCat search results

http://www.worldcat.org/
XPath: //div[@class='name']/a
λx: '- [' + x.text + '](http://www.worldcat.org' + x.href + ')'

Kindle books with highlights

https://kindle.amazon.com/your_highlights/
XPath: //span[@class='title']/a
λx: '- [' + x.text + '](https://kindle.amazon.com' + x.href + ')'

London Library Catalogue Search

http://pmt-eu.hosted.exlibrisgroup.com/
XPath: //h2[@class='EXLResultTitle']/a
λx: '- [' + x.text + '](http://pmt-eu.hosted.exlibrisgroup.com/primo_library/libweb/action/' + x.href + ')'

British Library Catalogue Search

http://catalogue.bl.uk/
XPath: //h2[@class='EXLResultTitle']/a
λx: '- [' + x.text + '](http://catalogue.bl.uk/primo_library/libweb/action/' + x.href + ')'

Keyboard Maestro forum posts

http://forum.keyboardmaestro.com/
XPath: //td[@class='main-link clearfix']/a
λx: '- [' + x.text + '](http://forum.keyboardmaestro.com' + x.href + ')'

FoldingText forum posts

http://support.foldingtext.com/
XPath: //td[@class='main-link clearfix']/a
λx: '- [' + x.text + '](http://support.foldingtext.com' + x.href + ')'

MLA search results

http://www.mla.org/
XPath: //a[@class='gs-title']
λx: '- [' + x.text + '](' + x.href + ')'

Amazon search results

http://www.amazon.com/
XPath: //a[@class='a-link-normal s-access-detail-page a-text-normal']
λx: '- [' + x.title + '](' + x.href + ')'

AbeBooks search results

http://www.abebooks.com/
XPath: //h2[@itemprop='offers']/a
λx: '- [' + x.title + '](http://www.abebooks.com' + x.href + ')'

AddAll book search results

http://www.addall.com/
XPath: //dt/b/font/a
λx: '- [' + x.text + '](http://www.addall.com' + x.href + ')'

CONTACT

- (For example to suggest the addition of any further search engines or databases for **automatic** mode)
- Rob Trew - Twitter @ComplexPoint