



- [Signup and Pricing](#)
- [Explore GitHub](#)
- [Features](#)
- [Blog](#)
- [Login](#)

## [mynymml](#) / [harmony](#)

- [Watch](#)
- [Fork](#)
- - [483](#)
  - [10](#)
- [Code](#)
- [Network](#)
- [Pull Requests 2](#)
- [Issues 7](#)
- [Stats & Graphs](#)
- [Tags 6](#)
- [Downloads 0](#)
- [branch: master](#)  
Switch Branches/Tags  
Filter branches/tags  
  - [Branches](#)
  - [Tags](#)

### [master](#)

- [Files](#)
- [Commits](#)
- [Branches 1](#)

### [harmony](#) / [lib](#) / [harmony](#) / **page.rb**



[mynymml](#) 2 years ago

[Add Page#to\\_html convenience method](#)

## 1 contributor

100644 124 lines (108 sloc) 2.799 kb

- [Edit this file](#)
- [Raw](#)
- [Blame](#)
- [History](#)

```
1 require 'pathname'
2 require 'tempfile'
3
4 require 'johnson/tracemonkey'
5 require 'envjs/runtime'
6
7 module Harmony
8   class Page
9
10    # Window factory
11    #
12    # @private
13    module Window #:nodoc:
14      extend self
15
16      # Cache the initial runtime. Parsing env.js (done automatically when
17      # Envjs::Runtime is extended) takes a while, so we only want to do this
18      # once.
19      #
20      # @private
21      BASE_RUNTIME = Johnson::Runtime.new
22      BASE_RUNTIME.extend(Envjs::Runtime)
23
24      def from_uri(uri)
25        BASE_RUNTIME.evaluate("window.open('#{uri}')" )
26      end
27
28      def from_document(document)
29        Tempfile.open('harmony') {|f| f << document; @path = f.path }
30        from_uri("file://#{@path}")
31      end
32
33      def blank
34        from_uri('about:blank')
35      end
36    end
37
38    # Create page from remote document.
39    #
40    # @example
41    #
42    #   Page.fetch('http://montrealrb.org')
43    #   Page.fetch('http://localhost:3000')
```

```
44 # Page.fetch('file:///home/mynynml/www/foo/index.html')
45 #
46 # @param [String] uri
47 # uri to fetch document from
48 #
49 # @return [Page]
50 # new page object preloaded with fetched document
51 #
52 def self.fetch(uri)
53   page = new
54   page.instance_variable_set(:@window, Window.from_uri(uri))
55   page
56 end
57
58 # Create new page containing given document.
59 #
60 # @param [String] document
61 # HTML document. Defaults to an "about:blank" window, with the basic
62 # structure: `<head><title></title></head><body></body></html>`
63 #
64 def initialize(document=nil)
65   @window = Window.from_document(document) if document
66 end
67
68 # Load one or more javascript files in page's context
69 #
70 # @param [#to_s, #to_s, ...] paths
71 # paths to js file
72 # @return [Page] self
73 #
74 def load(*paths)
75   paths.flatten.each do |path|
76     window.load(path.to_s)
77   end
78   self
79 end
80
81 # Evaluate Javascript code within this page's context.
82 #
83 # @param [String] code
84 # javascript code to execute
85 #
86 # @return [Object]
87 # last javascript statement's value, cast to a ruby object
88 #
89 def execute_js(code)
90   window.evaluate(code)
91 end
92 alias :x :execute_js
93
94 # DOM document's `window` object. Equivalent to the return value of
95 # `page.execute_js('window')`
96 #
97 # @return [Object]
98 # window DOM object
```

```
99   #
100  def window
101    @window ||= Window.blank
102  end
103
104  # Convenience method, equivalent to the return value of
105  # `page.execute_js('window.document')`
106  #
107  # @return [Object]
108  #   document DOM object
109  #
110  def document
111    window.document
112  end
113
114  # Page as html document
115  #
116  # @return [String] html
117  #
118  def to_html
119    document.innerHTML
120  end
121 end
122end
123
```

## GitHub Links

### GitHub

- [About](#)
- [Blog](#)
- [Features](#)
- [Contact & Support](#)
- [Training](#)
- [GitHub Enterprise](#)
- [Site Status](#)

### Tools

- [Gauges: Analyze web traffic](#)
- [Speaker Deck: Presentations](#)
- [Gist: Code snippets](#)
- [GitHub for Mac](#)
- [Issues for iPhone](#)
- [Job Board](#)

### Extras

- [GitHub Shop](#)
- [The Octodex](#)

## Documentation

- [GitHub Help](#)
- [Developer API](#)
- [GitHub Flavored Markdown](#)
- [GitHub Pages](#)
  
- [Terms of Service](#)
- [Privacy](#)
- [Security](#)

© 2012 GitHub Inc. All rights reserved.



Powered by the [Dedicated Servers](#) and [Cloud Computing](#) of Rackspace Hosting®

# Markdown Cheat Sheet

## Format Text

### Headers

```
# This is an <h1> tag
## This is an <h2> tag
##### This is an <h6> tag
```

### Text styles

```
*This text will be italic*
_This will also be italic_
**This text will be bold**
__This will also be bold__

*You **can** combine them*
```

## Lists

### Unordered

```
* Item 1
* Item 2
  * Item 2a
  * Item 2b
```

## Ordered

1. Item 1
2. Item 2
3. Item 3
  - \* Item 3a
  - \* Item 3b

## Miscellaneous

### Images

![GitHub Logo](/images/logo.png)

Format: ![Alt Text](url)

### Links

<http://github.com> - automatic!

[GitHub](http://github.com)

### Blockquotes

As Kanye West said:

> We're living the future so

> the present is our past.

## Code Examples in Markdown

### Syntax highlighting with [GFM](#)

```
```javascript
function fancyAlert(arg) {
  if(arg) {
    $.facebox({div: '#foo'})
  }
}
```
```

### Or, indent your code 4 spaces

Here is a Python code example  
without syntax highlighting:

```
def foo:
    if not bar:
        return true
```

### Inline code for comments

I think you should use an

`<abbr>` element here instead.

Something went wrong with that request. Please try again. [Dismiss](#)

## Looking for the GitHub logo?

- **GitHub Logo**



[Download](#)

- **The Octocat**



[Download](#)