

Grimoire — *Smooth CoffeeScript*



This literate program is *interactive* in its HTML₅ form. Edit a CoffeeScript segment to try it. You can see the generated JavaScript as you modify a CoffeeScript function by typing 'show name' after its definition.

Grimoire

A Grimoire is a book of magical incantations. This program got the name because it is full of magic. Not the superstitious kind — but magical constants, assumptions and implicit definitions — follow its style and conventions at your own peril. As it is with magic, this program does not actually work, it is all sleight of hand.

There is a little touch of self-referential magic in here: This literate program documents the internals of its own kind of literate program documents including itself.

If you have some CoffeeScript that would be more spiffy packaged in an interactive HTML₅ document, then be warned: As in magical potions, the ingredients are exotic and intoxicating. They include `ssam` (streaming sam rc script) from `plan9port`, `pandoc` (the Haskell universal format converter) and the enchanted `CodeMirror` — the embedded editor that shows your code as it *really* is.

The Crystal Ball of Hindsight One night, under the full moon, I might make this more approachable. There isn't anything going on in the `ssam` structural regular expressions or the `ssam` pipeline that could not be done in, say, a CoffeeScript program. It is this way because the `plan9` toolset is eminent for prototyping and experiments and `pandoc` does most of the work. Creating these documents is a little something I am doing for fun. Building a professional quality publishing pipeline — while interesting — would be an order of magnitude more work. This is not like any commercial application I have ever seen or written.

The program attempts to change as little as possible in the `markdown` format, but meddles — quite unwisely — in the internals of the HTML₅ created by `pandoc`. This is to create one document that is simultaneously a `markdown` document, a literate CoffeeScript program, and an interactive HTML₅ application.

Editor Commandline Commands

I use `acme` to stir the ingredients, but you can use any editor/command line you like — don't blame me for the result. Speaking of don't blame me: If this program does not work for you, then it is not a bug, it is because you don't believe in magic — because *seriously* you don't, do you?

Below are the commands I use to extract CoffeeScript code, execute it and to format these Grimoire documents. To execute the commands; middle-button select them in the `acme` environment (if you are reading the markdown source code). As you can see, the magical constants start right here. To use the commands with another document, snarf them and `s/grimoire/new-document-name/g`.

Simple Extraction This command extracts CoffeeScript to an editor buffer — only bird tracks followed by the capitalization that you see below are extracted. This is used to avoid extracting code blocks that are spelled in all capitals i.e `~~~~~ { .COFFEESCRIPT }`. In the online environment those blocks become read-only and are not executed.

```
Edit ,x/^~~~+[ ]*{\.[cC]offee[sS]cript.*}$/+ ,/^~~~+$/-p
```

Extract and Run Standalone The next command extracts the CoffeeScript and prepend definitions that are provided by the interactive environment but do not exist in the standard standalone environment. Creates a CoffeeScript file. Compiles to a JavaScript file. Runs the CoffeeScript to produce an HTML and an output file. You can edit this bit if the document is not producing HTML. Opens the HTML in your web browser and plumbs it back into `acme`.

```
Edit ,>ssam -n 'x/^~~~+[ ]*{\.[cC]offee[sS]cript.*}$/+ ,/^~~~+$/-' | cat embed-standalone.coffee - |
```

Create PDF with TeX The following command produces a PDF version of the document. It uses XeTeX which in this case must be present on your system together with too many TeX packages to list (try a full 1.6Gb install if you have the bandwidth or spend hours as I do: start from a basic TeX and add individual packages one-by-one in the Tex Live Utility). It is not required unless you want a PDF version. By the way XeTeX has some trouble with Unicode characters, sometimes. Anyway it assumes that the command above has been run, the `VerbatimInput` lines near the end of this document (you can only see them in the markdown not in the HTML because they are TeX commands) uses the created files to insert program output and JavaScript translation into the PDF. You can delete those lines if you don't want that.

```
Edit ,>markdown2pdf --listings --xetex '--template=pandoc-template.tex' -o grimoire.pdf; open grimoire.pdf
```

Create an HTML₅ Application The last command produces an interactive HTML₅ document. It does assume that the 2nd commands above has been run sometime earlier (to bootstrap itself). The command below works as a straight pipeline. First running the markdown through `pandoc` to produce HTML₅ with `section` and `mathml` elements¹. It creates a standalone document based on the template in `pandoc-template.html` and includes different CSS files for styling itself and its embedded editor. It also includes itself with the `-B (include-before-body)` option, not the interactive HTML that you might be reading, but the output-HTML that the interactive HTML produces.

The next four stages are all `ssam` substitutions in the HTML that `pandoc` produced. Talk about being dependent on internal implementation details — the `pandoc` used is version 1.8.2.

- (1) The relevant `<code>` elements (based on their class and spelling) are made contenteditable. That is only used to allow them to get focus and could possibly be replaced by pseudo onclick handlers. They are also set to *not* be `spellchecked`, which should be obvious for `<code>` elements. But even with that flag, Firefox still spellchecks the CoffeeScript in the elements displaying a lot of ugly squiggles.
- (2) Sections with an id that starts with `view-solution` are hidden and given a `reveal` event handler. The hidden part goes on to the next section, which in the markdown can be marked with a header `####` without a text. That bit is now depending on `pandoc` not optimizing those superfluous headers away. But it makes it possible for solutions to have multiple code blocks and images in them.
- (3) This part replaces the 1st and only the 1st image reference in a document with a canvas element. This is something that should be changed: a canvas element isn't usable without scripting and this could be done in scripting.
- (4) The last defaults images to be centered instead of left aligned. This should be done in CSS instead.

```
Edit ,>pandoc -f markdown -t html -S -5 --mathml --section-divs --css pandoc-template.css --css co
```

Interactive Environment

```
webfragment = ->
```

```
div id: "feature-detect", ->
  div id: "feature-javascript", style: "color:#FF0000; display: block", ->
    text "No JavaScript &rarr; no output and no interactivity."
  div id: "feature-mathml", style: "color:#FF0000; display: none", ->
    text "No MathML 1998 &rarr; math is not readable."
  div id: "feature-canvas", style: "color:#FF0000; display: none", ->
    text "No Canvas &rarr; graphical output is not rendered."
  div id: "feature-contenteditable", style: "color:#FF0000; display: none", ->
    text "No ContentEditable &rarr; CoffeeScript sections can not be changed."
```

¹MathML is at this time not rendered in the two consumer browsers, Microsoft Internet Explorer and Google Chrome. I have tried MathJax but that resulted in Opera displaying garbage instead of equations. I found that it was unacceptable that Opera which does show W3C standard MathML get degraded, just because Google and Microsoft have chosen not to implement the standard. Of course in a commercial project the decision would likely be the other way round because of Opera's tiny market share compared with IE and Chrome. MathML renders fine in Firefox, Opera, and Safari on OS X & iOS.

```

input class: 'field', type: 'button', value: 'Adjust layout', onclick: ->
  @value = if toggleLayout() then 'Layout: fixed' else 'Layout: freeflow'
input class: 'field', type: 'button', value: 'Select editor', onclick: ->
  @value = if switchEditor() then 'Editor: CodeMirror' else 'Editor: plain text'
input class: 'field', type: 'button', value: 'Evaluation', onclick: ->
  @value = if switchEvaluation() then ↑ Evaluate else 'Auto Evaluate'

```

External scripts These scripts should be the same as those listed in the application manifest, `pandoc-template.appcache` that is mentioned in the `pandoc-template.html` file.

```

script src: 'node_modules/coffee-script.js'
script src: 'node_modules/coffee-kup.js'
script src: 'node_modules/underscore.js'
script src: 'node_modules/qc.js'
script src: 'codemirror/codemirror.js'
script src: 'codemirror/coffeescript.js'

```

The Interactive Environment

```
coffeescript ->
```

```

@reveal = (instance) -> # Very pandoc specific: requires --section-divs
  instance.getElementsByTagName('section')[0]?.style.display = 'block'
  instance.getElementsByTagName('h5')[0]?.innerHTML = 'Solution'
  instance.onclick = undefined

```

```

@switchEditor = ->
  localStorage?.editor = if @useCodeMirror then 'TextArea' else 'CodeMirror'
  @useCodeMirror = not @useCodeMirror
@switchEvaluation = ->
  localStorage?.evaluation =
    if localStorage?.evaluation is 'manual'
      'automatic'
    else
      'manual'
  localStorage?.evaluation is 'manual'
@toggleLayout = ->
  fixedLayout = document.getElementById('page').style.maxWidth is ''
  localStorage?.fixedLayout = fixedLayout
  switchLayout fixedLayout
switchLayout = (fixedLayout) -> # Should be in CSS
  document.getElementById('page').style.maxWidth =
    if fixedLayout then '600px' else ''
  s = document.getElementById('page').style
  if fixedLayout
    s.webkitHyphens = 'auto'
    s.mozHyphens = 'auto'
    s.msHyphens = 'auto'
    s.hyphens = 'auto'
    s.textAlign = 'justify'
  else
    s.webkitHyphens = ''
    s.mozHyphens = ''
    s.msHyphens = ''
    s.hyphens = ''
    s.textAlign = ''
  fixedLayout

```

```

featureDetect = ->
  adjustCoverPosition = (idFeature) -> # Should be in CSS
    canvasCover = document.getElementById('drawCanvas')
    topPos = parseInt canvasCover?.style.top
    lineHeight = document.getElementById(idFeature)?.scrollHeight
    canvasCover?.style.top = (topPos + lineHeight) + 'px'

  mathmlDetect = ->
    (e = document.createElement 'div').innerHTML = '<math></math>'
    passed = e.firstChild and 'namespaceURI' of e.firstChild and
      e.firstChild.namespaceURI is 'http://www.w3.org/1998/Math/MathML'
    # TODO Needs a better test. Testing on Chrome directly because
    # it reports it supports MathML even though that is not the case.
    passed and not /Chrome/.test navigator.userAgent

  document.getElementById('feature-javascript')?.style.display = "none"
  used = (document.getElementsByTagName('math')?.length > 0)
  if used and mathmlDetect() is false
    document.getElementById('feature-mathml')?.style.display = "block"
    adjustCoverPosition 'feature-mathml'
  unless document.createElement('canvas').getContext?
    document.getElementById('feature-canvas')?.style.display = "block"
    adjustCoverPosition 'feature-canvas'
  unless 'isContentEditable' of document.documentElement
    document.getElementById('feature-contenteditable')?.style.display = "block"
    adjustCoverPosition 'feature-contenteditable'

```

```

window.onload = ->
  featureDetect()
  switchLayout(on) if localStorage?.fixedLayout isnt 'false'
  @useCodeMirror = localStorage?.editor is 'CodeMirror'

```

Runtime canvas support As mentioned in the command used to create an [HTML5 application, item #3](#) this should not be done automatically.

```

canvas = document.getElementById('drawCanvas')
window.ctx = canvas.getContext '2d' if canvas?

clearCanvas = ->
  if window.ctx?
    window.ctx.clearRect 0, 0,
      window.ctx.canvas.width, window.ctx.canvas.height
drawCanvas = (draw) ->
  clearCanvas()
  draw window.ctx

```

Runtime output support The `dataurl` in `addFrame` is not displayed in IE9 nor is its `innerHTML`.

```

getParent = (child) -> child?.parentElement ? child?.parentNode

addElement = (parent, text) ->
  newelem = document.createElement 'code'
  newelem.setAttribute 'class', 'sourceCode output'
  newelem.innerHTML = text
  parent.appendChild newelem

separator = (parent) ->
  if parent.getElementsByClassName('output').length is 0
    addElement parent, '<hr><br>'

```

```

addFrame = (parent, width, height, content) ->
  newelem = document.createElement 'iframe'
  newelem.setAttribute 'class', 'output'
  newelem.setAttribute 'width', width
  newelem.setAttribute 'height', height
  newelem.setAttribute 'src',
    "data:text/html;charset=utf-8,{encodeURIComponent content}"
  newelem.innerHTML =
    '''<div id="feature-dataurl" style="color:#FF0000; display: block">
      No or insufficient Data URL support &rarr;
      web page output is not rendered in this browser.</div>'''
  parent.appendChild newelem

showHere = (atTag, obj, shallow = false, symbol = '&rarr;') ->
  displayShallow = (o) ->
    """#{{"\n #{k}: #{v}" for own k,v of o}\n}"""
  msg = switch typeof obj
    when 'undefined', 'string', 'function'
      obj
    when 'object'
      if shallow
        displayShallow obj
      else
        try
          JSON.stringify obj
        catch err
          displayShallow obj
    else obj?.toString()
  parentTag = getParent atTag
  separator parentTag
  addElement parentTag, "#{symbol} #{msg}<br>"
  obj

showDocumentHere = (atTag, content, width = 300, height = 300) ->
  parentTag = getParent atTag
  separator parentTag
  addFrame parentTag, width, height, content
  return

```

CoffeeScript Evaluation This is called during startup for a full evaluation. It is also triggered by changes in one of the editor types, where it can be full evaluation with code stitching for short articles or incremental i.e. one code block for a whole book.

```

evaluateSource = (field = null) ->

```

```

show = (obj, shallow = false, symbol = '&rarr;') ->
  showHere currentTag, obj, shallow, symbol
  return # Suppress display of the return value
view = (obj, shallow = false, symbol = '&rarr;') ->
  showHere currentTag, obj, shallow, symbol
showDocument = (content, width = 300, height = 300) ->
  showDocumentHere currentTag, content, width, height
addErrorElement = (text) ->
  show """<span class="er">#{text}</span>"""

```

```

# Naming convention: 'button-' <type> '-' <codeTagId>
getCurrentTagId = (id) -> id.match(/button-\w+-(.*)/)[1]

```

```

runOnDemand = (func) ->
  show "<button id='button-run-#{currentTag.id}'> Run </button>"
  document.getElementById("button-run-#{currentTag.id}").onclick = ->
    currentTag = document.getElementById get_current_tag_id @id
    view = show = (obj, shallow = false, symbol = '&rArr;') ->
      showHere currentTag, obj, shallow, symbol
    showDocument = (content, width = 300, height = 300) ->
      showDocumentHere currentTag, content, width, height
    func()
  return

confirm = (message, func) ->
  show "#{message}" +
    " <button id='button-yes-#{currentTag.id}'> Yes </button>" +
    " <button id='button-no-#{currentTag.id}'> No </button>"
  document.getElementById("button-yes-#{currentTag.id}").onclick = ->
    currentTag = document.getElementById get_current_tag_id @id
    view = show = (obj, shallow = false, symbol = '&rArr;') ->
      showHere currentTag, obj, shallow, symbol
    showDocument = (content, width = 300, height = 300) ->
      showDocumentHere currentTag, content, width, height
    func true
  document.getElementById("button-no-#{currentTag.id}").onclick = ->
    currentTag = document.getElementById get_current_tag_id @id
    view = show = (obj, shallow = false, symbol = '&rArr;') ->
      showHere currentTag, obj, shallow, symbol
    showDocument = (content, width = 300, height = 300) ->
      showDocumentHere currentTag, content, width, height
    func false
  return

prompt = (message, defaultValue, func) ->
  show "#{message}" +
    " <input id='input-prompt-#{currentTag.id}' value='#{defaultValue}' />" +
    " <button id='button-prompt-#{currentTag.id}'> Go </button>"
  document.getElementById("button-prompt-#{currentTag.id}").onclick = ->
    currentTag = document.getElementById get_current_tag_id @id
    view = show = (obj, shallow = false, symbol = '&rArr;') ->
      showHere currentTag, obj, shallow, symbol
    showDocument = (content, width = 300, height = 300) ->
      showDocumentHere currentTag, content, width, height
    func document.getElementById("input-prompt-#{get_current_tag_id @id}").value
  return

```

QuickCheck You may ask: “What are these completely unrelated QuickCheck helpers doing here?” Well, it could be that I am simply trying to distract you while I pull something out of my sleeve — or it could be that I didn’t get round to moving them into `qc.js`, which as its name implies is written in JavaScript (and it is not a CommonJS module either).

Incidentally JavaScript could learn a thing or two from CoffeeScript. For example: it could use a back-tick feature for embedding CoffeeScript into it. Until the day when these helpers are moved to a better resting place, at least they are not doing much harm here ... unless you summon them. QuickCheck is a minor imp, using randomness to cause havoc in your code. Several examples are shown in ‘Smooth CoffeeScript’ which is why the helpers are here. Let me know if you should have converted `qc.js` to CoffeeScript or have written something better.

```

# HTML colored output for QuickCheck.
class HtmlListener extends ConsoleListener
  constructor: (@maxCollected = 10) ->
    log: (str) -> show str
    passed: (str) -> # print message as OK

```

```

    show ""<span class="kw">#{str}</span>""
invalid: (str) -> # print message as warning
    show ""<span class="dt">#{str}</span>""
failure: (str) -> # print message as error
    show ""<span class="al">#{str}</span>""
done: ->
    show 'Completed test.'
    resetProps() # Chain here if needed

# Enhanced noteArg returning its argument so it can be used inline.
Case::note = (a) -> @noteArg a; a
# Same as Case::note but also logs the noted args.
Case::noteVerbose = (a) -> @noteArg a; show @args; a

# Helper to declare a named test property for
# a function func taking types as arguments.
# Property is passed the testcase, the arguments
# and the result of calling func, it must return
# a boolean indicating success or failure.
testPure = (func, types, name, property) ->
    declare name, types, (c, a...) ->
        c.assert property c, a..., c.note func a...
    return

# Default qc configuration with 100 pass and 1000 invalid tests
qcConfig = new Config 100, 1000

# Test all known properties
test = (msg, func) ->
    _ .each [msg, func, runAllProps qcConfig, new HtmlListener],
    (o) -> unless _ .isUndefined o then show o

```

Traversing and unescaping The code for the field or the document is obtained with tags that relate the code back to its HTML element. The tag is needed to direct output to its place of origin. When an editor is instantiated for a `<code>` element it changes to a `<textarea>` so matching is on the class of a `<pre>`'s `firstChild`.

```

getCode = (codeTag) ->
    if codeTag.value?
        segment = codeTag.value
    else
        segment = codeTag.innerHTML
    #console.debug "From: #{segment}"
    code = segment.replace /<br>/g, '\n'
    code = code.replace /<[/]?span[^\>]*>/g, ''
    code = code.replace /[&]gt;/g, '>'
    code = code.replace /[&]lt;/g, '<'
    code = code.replace /[&]nbsp;/g, ' '
    #console.debug "To: #{code}"
    codeTag:codeTag, code:code

if field? and @incrementalEvaluation
    segments = [getCode field]
else
    # Obtain each code segment with an ownership tag
    segments =
        for codeSegment in window.document.getElementsByTagName 'pre'
            if codeSegment.className is 'sourceCode'
                codeTag = codeSegment.firstChild
                if codeTag.className is 'sourceCode coffeescript' \
                    or codeTag.className is 'sourceCode CoffeeScript'

```



```
getCode codeTag
```

Stitch together continuing code segments. This is done based on a test of whether they begin with whitespace. For example a class with methods in different code blocks or a program that is split up in sections as in this document. This only works during full evaluation.

```
segments = (segment for segment in segments when segment?)
for segment, i in segments
  if i > 0 and /^[ \s]/.test segment.code
    segment.code = segments[i-1].code + '\n' + segment.code
    segments[i-1] = undefined
segments = (segment for segment in segments when segment?)
```

Clear output A performance test showed that the vast majority of time was consumed by this little loop: deleting previously shown output from the DOM. It made full code reevaluation an impossibility for anything but short articles. It was tested in three different browsers and `removeChild` was unbelievably slow in all of them. I don't know of alternatives for quickly removing elements, so the evaluation model for large documents was changed to incremental.

```
for segment in segments
  outList = getParent(segment.codeTag).getElementsByClassName 'output'
  outLength = outList?.length
  while outList and outLength > 0
    elem = outList[--outLength]
    getParent(elem).removeChild elem
```

Evaluation No real sandbox here, so the names and loop construct was changed to minimize the chance of conflict with the code that is being executed. Since clearing output made full reevaluation impossible, a small system where variables that are mentioned in `globalNames` are automatically promoted from the execution environment to the global environment was set up. This makes the named previous definitions available in subsequent code blocks during incremental evaluation (where there is no code stitching).

```
# Evaluate code segments
incredibleLength = segments.length
for incredibleIndex in [0...incredibleLength] by 1
  incredibleStart = new Date()
  currentSegment = segments[incredibleIndex]
  currentTag = currentSegment.codeTag
  if currentTag.id is ''
    currentTag.id = "codeTag#{incredibleIndex}"
  do (currentTag) ->
    try
      draw = undefined
      incredibleResult = eval CoffeeScript.compile \
        currentSegment.code, bare:on
      drawCanvas draw if draw?
      if incredibleResult isnt undefined and
        typeof incredibleResult isnt 'function'
        show incredibleResult, on, '&crarr;'
      if globalNames?
        promoteToGlobal = (magic) ->
          try
            magicValue = eval magic
            catch error
              return # doesn't exist
            this[magic] = magicValue
            for magic in globalNames.split ' '
              promoteToGlobal magic
      catch error
        console.log error.message
```

```

        addErrorElement error
    return
    if showTiming? and showTiming is on
        incredibleTime = Number(0.001 *
            (new Date() - incredibleStart)).toFixed 3
        show "#{incredibleTime}s", on, ''
    return

```

```

# Textarea editor code evaluation
keyDownEvaluation = (evt) ->
    return unless evt?
    if localStorage?.evaluation is 'manual'
        if evt.keyCode is 13 and evt.shiftKey # ↑ Shift / Enter
            evt.preventDefault()
            evaluateSource evt.currentTarget
    if evt.keyCode is 13 and not evt.shiftKey
        evt.target?.rows++
    return
keyUpEvaluation = (evt) ->
    return unless evt?
    unless localStorage?.evaluation is 'manual'
        evaluateSource evt.currentTarget unless evt.keyCode in
            [16,37,38,39,40] # Ignore shift and arrow keys

# CodeMirror editor code evaluation
cmEvaluation = (editor) ->
    editor.save()
    evaluateSource editor.getTextArea()
cmAutoEvaluation = (editor) ->
    unless localStorage?.evaluation is 'manual'
        cmEvaluation editor
cmManualEvaluation = (editor) ->
    if localStorage?.evaluation is 'manual'
        cmEvaluation editor

```

Create an editor by converting a `<code>` element to a `<textarea>`. If CodeMirror is enabled then the `<textarea>` is used as a basis for that editor.

```

createEditor = (codeElement, text) ->
    newelem = document.createElement 'textarea'
    newelem.setAttribute 'id', codeElement.id
    newelem.setAttribute 'class', codeElement.getAttribute('class')
    countLines = (c for c in text when c is '\n').length + 1
    newelem.setAttribute 'rows', countLines
    newelem.setAttribute 'style', 'width: 98%;'
    newelem.setAttribute 'autofocus', 'true'
    newelem.setAttribute 'spellcheck', 'false'
    newelem.innerHTML = text
    newelem.addEventListener 'keydown', keyDownEvaluation, false
    newelem.addEventListener 'keyup', keyUpEvaluation, false
    getParent(codeElement).replaceChild newelem, codeElement
    if @useCodeMirror
        elemCodeMirror = CodeMirror.fromTextArea newelem,
            onChange: cmAutoEvaluation
            extraKeys: {'Shift-Enter': cmManualEvaluation}
            lineNumbers: true
            theme: 'panthème'

```

Setup editor activation

```

activateEditor = ->
    @removeEventListener 'focus', activateEditor, false

```

```

sourcecode = @innerHTML.toString().replace /<\/?span[^\>]*>/g, ''
sourcecode = sourcecode.replace /<br[ ]*[^>]*>/g, '\n'
createEditor this, sourcecode

# Activate text editor when code gets focus
for segment in document.getElementsByTagName 'pre'
  segment.firstChild.addEventListener 'focus', activateEditor, false

```

Complete evaluation works for smaller articles, but not for a whole book. This is primarily because getting rid of old output elements with the DOM `removeChild` is very slow.

So set `@incrementalEvaluation` to `on` in larger texts.

A suitable place is the same — likely hidden — section where `@globalNames` is initialized. Code stitching is only in effect when `@incrementalEvaluation` is `off`.

```

@incrementalEvaluation = off
evaluateSource()

```

Bootstrap Create the embeddable HTML/JavaScript fragment as output. This does not actually bootstrap Grimoire. If it did then as soon as an error happened during editing, the whole thing would come tumbling down. To bootstrap use the [Editor Commandline Commands](#) or copy the code from the output field.

Is it possible to copyright magic? Let's say that the HTML/JavaScript code that is produced by this document is public domain or MIT licensed. Then you can use that any way you want to at your own risk. This document itself is licensed as CC-BYSA. To get the all the files this depend on go to the [interactive](#) directory in the repository branch [gh-pages](#).

```

kup = if exports? then require 'coffeekup' else window.CoffeeKup

display = if exports?
  console.log
else
  (html) -> show _.escape html

display kup.render webfragment, format:on

```

Output

```

1 <div id="feature-detect">
2   <div id="feature-javascript" style="color:#FF0000; display: block">
3     No JavaScript &rarr; no output and no interactivity. </div>
4   <div id="feature-mathml" style="color:#FF0000; display: none">
5     No MathML 1998 &rarr; math is not readable. </div>
6   <div id="feature-canvas" style="color:#FF0000; display: none">
7     No Canvas &rarr; graphical output is not rendered. </div>
8   <div id="feature-contenteditable" style="color:#FF0000; display: none">
9     No ContentEditable &rarr; CoffeeScript sections can not be changed. </div>
10 </div>
11 <input class="field" type="button" value="Adjust layout" onclick="(function () {
12   return this.value = toggleLayout() ? 'Layout: fixed' : 'Layout: freeflow';
13   }).call(this);" />
14 <input class="field" type="button" value="Select editor" onclick="(function () {
15   return this.value = switchEditor() ? 'Editor: CodeMirror' : 'Editor: plain text';
16   }).call(this);" />
17 <input class="field" type="button" value="Evaluation" onclick="(function () {
18   return this.value = switchEvaluation() ? '↑ Evaluate' : 'Auto Evaluate';
19   }).call(this);" />
20 <script src="node_modules/coffee-script.js"></script>

```

```

21 <script src="node_modules/coffeescript.js"></script>
22 <script src="node_modules/underscore.js"></script>
23 <script src="node_modules/qc.js"></script>
24 <script src="codemirror/codemirror.js"></script>
25 <script src="codemirror/coffeescript.js"></script>
26 <script>var __slice = Array.prototype.slice;var __hasProp = Object.prototype.hasOwnProperty;var __bind = function(fn, me){ return function() {
27     var featureDetect, switchLayout;
28     this.reveal = function(instance) {
29         var _ref, _ref2;
30         if ((_ref = instance.getElementsByTagName('section')[0]) != null) {
31             _ref.style.display = 'block';
32         }
33         if ((_ref2 = instance.getElementsByTagName('h5')[0]) != null) {
34             _ref2.innerHTML = 'Solution';
35         }
36         return instance.onclick = void 0;
37     };
38     this.switchEditor = function() {
39         if (typeof localStorage != "undefined" && localStorage != null) {
40             localStorage.editor = this.useCodeMirror ? 'TextArea' : 'CodeMirror';
41         }
42         return this.useCodeMirror = !this.useCodeMirror;
43     };
44     this.switchEvaluation = function() {
45         if (typeof localStorage != "undefined" && localStorage != null) {
46             localStorage.evaluation = (typeof localStorage != "undefined" && localStorage != null ? localStorage.evaluation : void 0);
47         }
48         return (typeof localStorage != "undefined" && localStorage != null ? localStorage.evaluation : void 0) === 'manual';
49     };
50     this.toggleLayout = function() {
51         var fixedLayout;
52         fixedLayout = document.getElementById('page').style.maxWidth === '';
53         if (typeof localStorage != "undefined" && localStorage != null) {
54             localStorage.fixedLayout = fixedLayout;
55         }
56         return switchLayout(fixedLayout);
57     };
58     switchLayout = function(fixedLayout) {
59         var s;
60         document.getElementById('page').style.maxWidth = fixedLayout ? '600px' : '';
61         s = document.getElementById('page').style;
62         if (fixedLayout) {
63             s.webkitHyphens = 'auto';
64             s.mozHyphens = 'auto';
65             s.msHyphens = 'auto';
66             s.hyphens = 'auto';
67             s.textAlign = 'justify';
68         } else {
69             s.webkitHyphens = '';
70             s.mozHyphens = '';
71             s.msHyphens = '';
72             s.hyphens = '';
73             s.textAlign = '';
74         }
75         return fixedLayout;
76     };
77     featureDetect = function() {
78         var adjustCoverPosition, mathmlDetect, used, _ref, _ref2, _ref3, _ref4, _ref5;
79         adjustCoverPosition = function(idFeature) {
80             var canvasCover, lineHeight, topPos, _ref;
81             canvasCover = document.getElementById('drawCanvas');
82             topPos = parseInt(canvasCover != null ? canvasCover.style.top : void 0);
83             lineHeight = (_ref = document.getElementById(idFeature)) != null ? _ref.scrollHeight : void 0;
84             return canvasCover != null ? canvasCover.style.top = (topPos + lineHeight) + 'px' : void 0;
85         };
86         mathmlDetect = function() {
87             var e, passed;
88             (e = document.createElement('div')).innerHTML = '<math></math>';
89             passed = e.firstChild && 'namespaceURI' in e.firstChild && e.firstChild.namespaceURI === 'http://www.w3.org/1998/Math/MathML';
90             return passed && !/Chrome/.test(navigator.userAgent);
91         };
92         if ((_ref = document.getElementById('feature-javascript')) != null) {

```

```

93     _ref.style.display = "none";
94 }
95 used = ((_ref2 = document.getElementsByTagName('math')) != null ? _ref2.length : void 0) > 0;
96 if (used && mathmlDetect() === false) {
97     if ((_ref3 = document.getElementById('feature-mathml')) != null) {
98         _ref3.style.display = "block";
99     }
100     adjustCoverPosition('feature-mathml');
101 }
102 if (document.createElement('canvas').getContext() == null) {
103     if ((_ref4 = document.getElementById('feature-canvas')) != null) {
104         _ref4.style.display = "block";
105     }
106     adjustCoverPosition('feature-canvas');
107 }
108 if (!('isContentEditable' in document.documentElement)) {
109     if ((_ref5 = document.getElementById('feature-contenteditable')) != null) {
110         _ref5.style.display = "block";
111     }
112     return adjustCoverPosition('feature-contenteditable');
113 }
114 };
115 return window.onload = function() {
116     var activateEditor, addElement, addFrame, canvas, clearCanvas, cmAutoEvaluation, cmEvaluation, cmManualEvaluation, createEditor;
117     featureDetect();
118     if ((typeof localStorage !== "undefined" && localStorage !== null ? localStorage.fixedLayout : void 0) !== 'false') {
119         switchLayout(true);
120     }
121     this.useCodeMirror = (typeof localStorage !== "undefined" && localStorage !== null ? localStorage.editor : void 0) === 'CodeMirror';
122     canvas = document.getElementById('drawCanvas');
123     if (canvas != null) window.ctx = canvas.getContext('2d');
124     clearCanvas = function() {
125         if (window.ctx != null) {
126             return window.ctx.clearRect(0, 0, window.ctx.canvas.width, window.ctx.canvas.height);
127         }
128     };
129     drawCanvas = function(draw) {
130         clearCanvas();
131         return draw(window.ctx);
132     };
133     getParent = function(child) {
134         var _ref;
135         return (_ref = child != null ? child.parentElement : void 0) != null ? _ref : child != null ? child.parentNode : void 0;
136     };
137     addElement = function(parent, text) {
138         var newelem;
139         newelem = document.createElement('code');
140         newelem.setAttribute('class', 'sourceCode output');
141         newelem.innerHTML = text;
142         return parent.appendChild(newelem);
143     };
144     separator = function(parent) {
145         if (parent.getElementsByClassName('output').length === 0) {
146             return addElement(parent, '<hr><br>');
147         }
148     };
149     addFrame = function(parent, width, height, content) {
150         var newelem;
151         newelem = document.createElement('iframe');
152         newelem.setAttribute('class', 'output');
153         newelem.setAttribute('width', width);
154         newelem.setAttribute('height', height);
155         newelem.setAttribute('src', "data:text/html;charset=utf-8," + (encodeURIComponent(content)));
156         newelem.innerHTML = '<div id="feature-dataurl" style="color:#FF0000; display: block">\nNo or insufficient Data URL support\n';
157         return parent.appendChild(newelem);
158     };
159     showHere = function(atTag, obj, shallow, symbol) {
160         var displayShallow, msg, parentTag;
161         if (shallow == null) shallow = false;
162         if (symbol == null) symbol = '&rarr;';
163         displayShallow = function(o) {
164             var k, v;

```

```

165     return "{" + ((function() {
166         var _results;
167         _results = [];
168         for (k in o) {
169             if (!__hasProp.call(o, k)) continue;
170             v = o[k];
171             _results.push("\n  " + k + ": " + v);
172         }
173         return _results;
174     })() + "\n}";
175 };
176 msg = (function() {
177     switch (typeof obj) {
178         case 'undefined':
179         case 'string':
180         case 'function':
181             return obj;
182         case 'object':
183             if (shallow) {
184                 return displayShallow(obj);
185             } else {
186                 try {
187                     return JSON.stringify(obj);
188                 } catch (err) {
189                     return displayShallow(obj);
190                 }
191             }
192             break;
193         default:
194             return obj != null ? obj.toString() : void 0;
195     }
196 })();
197 parentTag = getParent(atTag);
198 separator(parentTag);
199 addElement(parentTag, "" + symbol + " " + msg + "<br>");
200 return obj;
201 };
202 showDocumentHere = function(atTag, content, width, height) {
203     var parentTag;
204     if (width == null) width = 300;
205     if (height == null) height = 300;
206     parentTag = getParent(atTag);
207     separator(parentTag);
208     addFrame(parentTag, width, height, content);
209 };
210 evaluateSource = function(field) {
211     var HtmlListener, addErrorElement, codeSegment, codeTag, confirm, currentSegment, currentTag, elem, getCode, getCurrentTag;
212     if (field == null) field = null;
213     show = function(obj, shallow, symbol) {
214         if (shallow == null) shallow = false;
215         if (symbol == null) symbol = '&rarr;';
216         showHere(currentTag, obj, shallow, symbol);
217     };
218     view = function(obj, shallow, symbol) {
219         if (shallow == null) shallow = false;
220         if (symbol == null) symbol = '&rarr;';
221         return showHere(currentTag, obj, shallow, symbol);
222     };
223     showDocument = function(content, width, height) {
224         if (width == null) width = 300;
225         if (height == null) height = 300;
226         return showDocumentHere(currentTag, content, width, height);
227     };
228     addErrorElement = function(text) {
229         return show("<span class='er'>" + text + "</span>");
230     };
231     getCurrentTagId = function(id) {
232         return id.match(/button-\w+-(.*)/)[1];
233     };
234     runOnDemand = function(func) {
235         show("<button id='button-run-" + currentTag.id + "'> Run </button>");
236         document.getElementById("button-run-" + currentTag.id).onclick = function() {

```

```

237     var currentTag;
238     currentTag = document.getElementById(getCurrentTagId(this.id));
239     view = show = function(obj, shallow, symbol) {
240         if (shallow == null) shallow = false;
241         if (symbol == null) symbol = '&rArr;';
242         return showHere(currentTag, obj, shallow, symbol);
243     };
244     showDocument = function(content, width, height) {
245         if (width == null) width = 300;
246         if (height == null) height = 300;
247         return showDocumentHere(currentTag, content, width, height);
248     };
249     return func();
250 };
251 };
252 confirm = function(message, func) {
253     show(("" + message) + (" <button id='button-yes-' + currentTag.id + "'> Yes </button>") + (" <button id='button-no-' +
254     document.getElementById("button-yes-" + currentTag.id).onclick = function() {
255         var currentTag;
256         currentTag = document.getElementById(getCurrentTagId(this.id));
257         view = show = function(obj, shallow, symbol) {
258             if (shallow == null) shallow = false;
259             if (symbol == null) symbol = '&rArr;';
260             return showHere(currentTag, obj, shallow, symbol);
261         };
262         showDocument = function(content, width, height) {
263             if (width == null) width = 300;
264             if (height == null) height = 300;
265             return showDocumentHere(currentTag, content, width, height);
266         };
267         return func(true);
268     };
269     document.getElementById("button-no-" + currentTag.id).onclick = function() {
270         var currentTag;
271         currentTag = document.getElementById(getCurrentTagId(this.id));
272         view = show = function(obj, shallow, symbol) {
273             if (shallow == null) shallow = false;
274             if (symbol == null) symbol = '&rArr;';
275             return showHere(currentTag, obj, shallow, symbol);
276         };
277         showDocument = function(content, width, height) {
278             if (width == null) width = 300;
279             if (height == null) height = 300;
280             return showDocumentHere(currentTag, content, width, height);
281         };
282         return func(false);
283     };
284 };
285 prompt = function(message, defaultValue, func) {
286     show(("" + message) + (" <input id='input-prompt-' + currentTag.id + "' value='" + defaultValue + "' />") + (" <button
287     document.getElementById("button-prompt-" + currentTag.id).onclick = function() {
288         var currentTag;
289         currentTag = document.getElementById(getCurrentTagId(this.id));
290         view = show = function(obj, shallow, symbol) {
291             if (shallow == null) shallow = false;
292             if (symbol == null) symbol = '&rArr;';
293             return showHere(currentTag, obj, shallow, symbol);
294         };
295         showDocument = function(content, width, height) {
296             if (width == null) width = 300;
297             if (height == null) height = 300;
298             return showDocumentHere(currentTag, content, width, height);
299         };
300         return func(document.getElementById("input-prompt-" + (getCurrentTagId(this.id))).value);
301     };
302 };
303 HtmlListener = (function(_super) {
304
305     __extends(HtmlListener, _super);
306
307     function HtmlListener(maxCollected) {
308         this.maxCollected = maxCollected != null ? maxCollected : 10;

```

```

309     }
310
311     HtmlListener.prototype.log = function(str) {
312         return show(str);
313     };
314
315     HtmlListener.prototype.passed = function(str) {
316         return show("<span class=\"kw\">" + str + "</span>");
317     };
318
319     HtmlListener.prototype.invalid = function(str) {
320         return show("<span class=\"dt\">" + str + "</span>");
321     };
322
323     HtmlListener.prototype.failure = function(str) {
324         return show("<span class=\"al\">" + str + "</span>");
325     };
326
327     HtmlListener.prototype.done = function() {
328         show('Completed test. ');
329         return resetProps();
330     };
331
332     return HtmlListener;
333
334 })(ConsoleListener);
335 Case.prototype.note = function(a) {
336     this.noteArg(a);
337     return a;
338 };
339 Case.prototype.noteVerbose = function(a) {
340     this.noteArg(a);
341     show(this.args);
342     return a;
343 };
344 testPure = function(func, types, name, property) {
345     declare(name, types, function() {
346         var a, c;
347         c = arguments[0], a = 2 <= arguments.length ? __slice.call(arguments, 1) : [];
348         return c.assert(property.apply(null, [c].concat(__slice.call(a), [c.note(func.apply(null, a))]]));
349     });
350 };
351 qcConfig = new Config(100, 1000);
352 test = function(msg, func) {
353     return __.each([msg, func, runAllProps(qcConfig, new HtmlListener)], function(o) {
354         if (!_.isUndefined(o)) return show(o);
355     });
356 };
357 getCode = function(codeTag) {
358     var code, segment;
359     if (codeTag.value != null) {
360         segment = codeTag.value;
361     } else {
362         segment = codeTag.innerHTML;
363     }
364     code = segment.replace(/<br>/g, '\n');
365     code = code.replace(/<[/\]?span[>]*>/g, '');
366     code = code.replace(/&gt;/g, '>');
367     code = code.replace(/&lt;/g, '<');
368     code = code.replace(/&nbsp;/g, ' ');
369     return {
370         codeTag: codeTag,
371         code: code
372     };
373 };
374 if ((field != null) && this.incrementalEvaluation) {
375     segments = [getCode(field)];
376 } else {
377     segments = (function() {
378         var _i, _len, _ref, _results;
379         _ref = window.document.getElementsByTagName('pre');
380         _results = [];

```



```

381     for (_i = 0, _len = _ref.length; _i < _len; _i++) {
382         codeSegment = _ref[_i];
383         if (codeSegment.className === 'sourceCode') {
384             codeTag = codeSegment.firstChild;
385             if (codeTag.className === 'sourceCode coffeescript' || codeTag.className === 'sourceCode CoffeeScript') {
386                 _results.push(getCode(codeTag));
387             } else {
388                 _results.push(void 0);
389             }
390         } else {
391             _results.push(void 0);
392         }
393     }
394     return _results;
395 })();
396 segments = (function() {
397     var _i, _len, _results;
398     _results = [];
399     for (_i = 0, _len = segments.length; _i < _len; _i++) {
400         segment = segments[_i];
401         if (segment != null) _results.push(segment);
402     }
403     return _results;
404 })();
405 for (i = 0, _len = segments.length; i < _len; i++) {
406     segment = segments[i];
407     if (i > 0 && /^[^s]/.test(segment.code)) {
408         segment.code = segments[i - 1].code + '\n' + segment.code;
409         segments[i - 1] = void 0;
410     }
411 }
412 segments = (function() {
413     var _i, _len2, _results;
414     _results = [];
415     for (_i = 0, _len2 = segments.length; _i < _len2; _i++) {
416         segment = segments[_i];
417         if (segment != null) _results.push(segment);
418     }
419     return _results;
420 })();
421 }
422 for (_i = 0, _len2 = segments.length; _i < _len2; _i++) {
423     segment = segments[_i];
424     outList = getParent(segment.codeTag).getElementsByClassName('output');
425     outLength = outList != null ? outList.length : void 0;
426     while (outList && outLength > 0) {
427         elem = outList[--outLength];
428         getParent(elem).removeChild(elem);
429     }
430 }
431 incredibleLength = segments.length;
432 _fn = function(currentTag) {
433     var draw, incredibleResult, magic, promoteToGlobal, _j, _len3, _ref;
434     try {
435         draw = void 0;
436         incredibleResult = eval(CoffeeScript.compile(currentSegment.code, {
437             bare: true
438         }));
439         if (draw != null) drawCanvas(draw);
440         if (incredibleResult !== void 0 && typeof incredibleResult !== 'function') {
441             show(incredibleResult, true, '&crarr;');
442         }
443         if (typeof globalNames !== "undefined" && globalNames !== null) {
444             promoteToGlobal = function(magic) {
445                 var magicValue;
446                 try {
447                     magicValue = eval(magic);
448                 } catch (error) {
449                     return;
450                 }
451                 return this[magic] = magicValue;
452             };

```

```

453         _ref = globalNames.split(' ');
454         for (_j = 0, _len3 = _ref.length; _j < _len3; _j++) {
455             magic = _ref[_j];
456             promoteToGlobal(magic);
457         }
458     }
459     } catch (error) {
460         console.log(error.message);
461         addErrorElement(error);
462     }
463 };
464 for (incredibleIndex = 0; incredibleIndex < incredibleLength; incredibleIndex += 1) {
465     incredibleStart = new Date();
466     currentSegment = segments[incredibleIndex];
467     currentTag = currentSegment.codeTag;
468     if (currentTag.id === '') currentTag.id = "codeTag" + incredibleIndex;
469     _fn(currentTag);
470     if ((typeof showTiming !== "undefined" && showTiming !== null) && showTiming === true) {
471         incredibleTime = Number(0.001 * (new Date() - incredibleStart)).toFixed(3);
472         show(" " + incredibleTime + "s", true, ' ');
473     }
474 }
475 };
476 keyDownEvaluation = function(evt) {
477     var _ref;
478     if (evt == null) return;
479     if ((typeof localStorage !== "undefined" && localStorage !== null ? localStorage.evaluation : void 0) === 'manual') {
480         if (evt.keyCode === 13 && evt.shiftKey) {
481             evt.preventDefault();
482             evaluateSource(evt.currentTarget);
483         }
484     }
485     if (evt.keyCode === 13 && !evt.shiftKey) {
486         if ((_ref = evt.target) != null) _ref.rows++;
487     }
488 };
489 keyUpEvaluation = function(evt) {
490     var _ref;
491     if (evt == null) return;
492     if ((typeof localStorage !== "undefined" && localStorage !== null ? localStorage.evaluation : void 0) !== 'manual') {
493         if ((_ref = evt.keyCode) !== 16 && _ref !== 37 && _ref !== 38 && _ref !== 39 && _ref !== 40) {
494             return evaluateSource(evt.currentTarget);
495         }
496     }
497 };
498 cmEvaluation = function(editor) {
499     editor.save();
500     return evaluateSource(editor.getTextArea());
501 };
502 cmAutoEvaluation = function(editor) {
503     if ((typeof localStorage !== "undefined" && localStorage !== null ? localStorage.evaluation : void 0) !== 'manual') {
504         return cmEvaluation(editor);
505     }
506 };
507 cmManualEvaluation = function(editor) {
508     if ((typeof localStorage !== "undefined" && localStorage !== null ? localStorage.evaluation : void 0) === 'manual') {
509         return cmEvaluation(editor);
510     }
511 };
512 createEditor = function(codeElement, text) {
513     var c, countLines, elemCodeMirror, newelem;
514     newelem = document.createElement('textare');
515     newelem.setAttribute('id', codeElement.id);
516     newelem.setAttribute('class', codeElement.getAttribute('class'));
517     countLines = ((function() {
518         var _i, _len, _results;
519         _results = [];
520         for (_i = 0, _len = text.length; _i < _len; _i++) {
521             c = text[_i];
522             if (c === '\n') _results.push(c);
523         }
524         return _results;

```

```

525     })).length + 1;
526     newelem.setAttribute('rows', countLines);
527     newelem.setAttribute('style', 'width: 98%;');
528     newelem.setAttribute('autofocus', 'true');
529     newelem.setAttribute('spellcheck', 'false');
530     newelem.innerHTML = text;
531     newelem.addEventListener('keydown', keyDownEvaluation, false);
532     newelem.addEventListener('keyup', keyUpEvaluation, false);
533     getParent(codeElement).replaceChild(newelem, codeElement);
534     if (this.useCodeMirror) {
535         return elemCodeMirror = CodeMirror.fromTextArea(newelem, {
536             onChange: cmAutoEvaluation,
537             extraKeys: {
538                 'Shift-Enter': cmManualEvaluation
539             },
540             lineNumbers: true,
541             theme: 'pantheme'
542         });
543     }
544 };
545 activateEditor = function() {
546     var sourcecode;
547     this.removeEventListener('focus', activateEditor, false);
548     sourcecode = this.innerHTML.toString().replace(/<\/?span[^\>]*>/g, '');
549     sourcecode = sourcecode.replace(/<br[ ]*^[^\>]*>/g, '\n');
550     return createEditor(this, sourcecode);
551 };
552 _ref = document.getElementsByTagName('pre');
553 for (_i = 0, _len = _ref.length; _i < _len; _i++) {
554     segment = _ref[_i];
555     segment.firstChild.addEventListener('focus', activateEditor, false);
556 }
557 this.incrementalEvaluation = false;
558 return evaluateSource();
559 };
560 }).call(this);</script>
561

```

JavaScript

```

1  (function() {
2      var display, kup, show, showDocument, webfragment,
3          __hasProp = Object.prototype.hasOwnProperty,
4          __extends = function(child, parent) { for (var key in parent) { if (__hasProp.call(parent, key)) child[key] = parent[key]; } fun
5          __slice = Array.prototype.slice;
6
7      show = console.log;
8
9      showDocument = function(doc, width, height) {
10         return show(doc);
11     };
12
13     webfragment = function() {
14         div({
15             id: "feature-detect"
16         }, function() {
17             div({
18                 id: "feature-javascript",
19                 style: "color:#FF0000; display: block"
20             }, function() {
21                 return text("No JavaScript &rarr; no output and no interactivity.");
22             });
23             div({
24                 id: "feature-mathml",
25                 style: "color:#FF0000; display: none"
26             }, function() {
27                 return text("No MathML 1998 &rarr; math is not readable.");
28             });
29             div({
30                 id: "feature-canvas",

```

```

31     style: "color:#FF0000; display: none"
32 }, function() {
33     return text("No Canvas &arr; graphical output is not rendered.");
34 });
35 return div({
36     id: "feature-contenteditable",
37     style: "color:#FF0000; display: none"
38 }, function() {
39     return text("No ContentEditable &arr; CoffeeScript sections can not be changed.");
40 });
41 });
42 input({
43     "class": 'field',
44     type: 'button',
45     value: 'Adjust layout',
46     onclick: function() {
47         return this.value = toggleLayout() ? 'Layout: fixed' : 'Layout: freeflow';
48     }
49 });
50 input({
51     "class": 'field',
52     type: 'button',
53     value: 'Select editor',
54     onclick: function() {
55         return this.value = switchEditor() ? 'Editor: CodeMirror' : 'Editor: plain text';
56     }
57 });
58 input({
59     "class": 'field',
60     type: 'button',
61     value: 'Evaluation',
62     onclick: function() {
63         return this.value = switchEvaluation() ? '↑ Evaluate' : 'Auto Evaluate';
64     }
65 });
66 script({
67     src: 'node_modules/coffee-script.js'
68 });
69 script({
70     src: 'node_modules/coffeeup.js'
71 });
72 script({
73     src: 'node_modules/underscore.js'
74 });
75 script({
76     src: 'node_modules/qc.js'
77 });
78 script({
79     src: 'codemirror/codemirror.js'
80 });
81 script({
82     src: 'codemirror/coffeescript.js'
83 });
84 return coffeescript(function() {
85     var featureDetect, switchLayout;
86     this.reveal = function(instance) {
87         var _ref, _ref2;
88         if ((_ref = instance.getElementsByTagName('section')[0]) != null) {
89             _ref.style.display = 'block';
90         }
91         if ((_ref2 = instance.getElementsByTagName('h5')[0]) != null) {
92             _ref2.innerHTML = 'Solution';
93         }
94         return instance.onclick = void 0;
95     };
96     this.switchEditor = function() {
97         if (typeof localStorage != "undefined" && localStorage != null) {
98             localStorage.editor = this.useCodeMirror ? 'TextArea' : 'CodeMirror';
99         }
100         return this.useCodeMirror = !this.useCodeMirror;
101     };
102     this.switchEvaluation = function() {

```

```

103     if (typeof localStorage !== "undefined" && localStorage !== null) {
104         localStorage.evaluation = (typeof localStorage !== "undefined" && localStorage !== null ? localStorage.evaluation : void 0);
105     }
106     return (typeof localStorage !== "undefined" && localStorage !== null ? localStorage.evaluation : void 0) === 'manual';
107 };
108 this.toggleLayout = function() {
109     var fixedLayout;
110     fixedLayout = document.getElementById('page').style.maxWidth === '';
111     if (typeof localStorage !== "undefined" && localStorage !== null) {
112         localStorage.fixedLayout = fixedLayout;
113     }
114     return switchLayout(fixedLayout);
115 };
116 switchLayout = function(fixedLayout) {
117     var s;
118     document.getElementById('page').style.maxWidth = fixedLayout ? '600px' : '';
119     s = document.getElementById('page').style;
120     if (fixedLayout) {
121         s.webkitHyphens = 'auto';
122         s.mozHyphens = 'auto';
123         s.msHyphens = 'auto';
124         s.hyphens = 'auto';
125         s.textAlign = 'justify';
126     } else {
127         s.webkitHyphens = '';
128         s.mozHyphens = '';
129         s.msHyphens = '';
130         s.hyphens = '';
131         s.textAlign = '';
132     }
133     return fixedLayout;
134 };
135 featureDetect = function() {
136     var adjustCoverPosition, mathmlDetect, used, _ref, _ref2, _ref3, _ref4, _ref5;
137     adjustCoverPosition = function(idFeature) {
138         var canvasCover, lineHeight, topPos, _ref;
139         canvasCover = document.getElementById('drawCanvas');
140         topPos = parseInt(canvasCover != null ? canvasCover.style.top : void 0);
141         lineHeight = (_ref = document.getElementById(idFeature)) != null ? _ref.scrollHeight : void 0;
142         return canvasCover != null ? canvasCover.style.top = (topPos + lineHeight) + 'px' : void 0;
143     };
144     mathmlDetect = function() {
145         var e, passed;
146         (e = document.createElement('div')).innerHTML = '<math></math>';
147         passed = e.firstChild && 'namespaceURI' in e.firstChild && e.firstChild.namespaceURI === 'http://www.w3.org/1998/Math/MathML';
148         return passed && !/Chrome/.test(navigator.userAgent);
149     };
150     if ((_ref = document.getElementById('feature-javascript')) != null) {
151         _ref.style.display = "none";
152     }
153     used = ((_ref2 = document.getElementsByTagName('math')) != null ? _ref2.length : void 0) > 0;
154     if (used && mathmlDetect() === false) {
155         if ((_ref3 = document.getElementById('feature-mathml')) != null) {
156             _ref3.style.display = "block";
157         }
158         adjustCoverPosition('feature-mathml');
159     }
160     if (document.createElement('canvas').getContext == null) {
161         if ((_ref4 = document.getElementById('feature-canvas')) != null) {
162             _ref4.style.display = "block";
163         }
164         adjustCoverPosition('feature-canvas');
165     }
166     if (!('isContentEditable' in document.documentElement)) {
167         if ((_ref5 = document.getElementById('feature-contenteditable')) != null) {
168             _ref5.style.display = "block";
169         }
170         return adjustCoverPosition('feature-contenteditable');
171     }
172 };
173 return window.onload = function() {
174     var activateEditor, addElement, addFrame, canvas, clearCanvas, cmAutoEvaluation, cmEvaluation, cmManualEvaluation, createEditor,

```

```

175     featureDetect();
176     if ((typeof localStorage !== "undefined" && localStorage !== null ? localStorage.fixedLayout : void 0) !== 'false') {
177         switchLayout(true);
178     }
179     this.useCodeMirror = (typeof localStorage !== "undefined" && localStorage !== null ? localStorage.editor : void 0) === 'CodeMirror';
180     canvas = document.getElementById('drawCanvas');
181     if (canvas != null) window.ctx = canvas.getContext('2d');
182     clearCanvas = function() {
183         if (window.ctx != null) {
184             return window.ctx.clearRect(0, 0, window.ctx.canvas.width, window.ctx.canvas.height);
185         }
186     };
187     drawCanvas = function(draw) {
188         clearCanvas();
189         return draw(window.ctx);
190     };
191     getParent = function(child) {
192         var _ref;
193         return (_ref = child != null ? child.parentElement : void 0) != null ? _ref : child != null ? child.parentNode : void 0;
194     };
195     addElement = function(parent, text) {
196         var newelem;
197         newelem = document.createElement('code');
198         newelem.setAttribute('class', 'sourceCode output');
199         newelem.innerHTML = text;
200         return parent.appendChild(newelem);
201     };
202     separator = function(parent) {
203         if (parent.getElementsByTagName('output').length === 0) {
204             return addElement(parent, '<hr><br>');
205         }
206     };
207     newFrame = function(parent, width, height, content) {
208         var newelem;
209         newelem = document.createElement('iframe');
210         newelem.setAttribute('class', 'output');
211         newelem.setAttribute('width', width);
212         newelem.setAttribute('height', height);
213         newelem.setAttribute('src', "data:text/html;charset=utf-8," + (encodeURIComponent(content)));
214         newelem.innerHTML = '<div id="feature-dataurl" style="color:#FF0000; display: block">\nNo or insufficient Data URL support\n';
215         return parent.appendChild(newelem);
216     };
217     showHere = function(atTag, obj, shallow, symbol) {
218         var displayShallow, msg, parentTag;
219         if (shallow == null) shallow = false;
220         if (symbol == null) symbol = '&#x2192;';
221         displayShallow = function(o) {
222             var k, v;
223             return "{" + ((function() {
224                 var _results;
225                 _results = [];
226                 for (k in o) {
227                     if (!__hasProp.call(o, k)) continue;
228                     v = o[k];
229                     _results.push("\n  " + k + ": " + v);
230                 }
231                 return _results;
232             })()) + "\n}";
233         };
234         msg = (function() {
235             switch (typeof obj) {
236                 case 'undefined':
237                 case 'string':
238                 case 'function':
239                     return obj;
240                 case 'object':
241                     if (shallow) {
242                         return displayShallow(obj);
243                     } else {
244                         try {
245                             return JSON.stringify(obj);
246                         } catch (err) {

```

```

247         return displayShallow(obj);
248     }
249 }
250 break;
251 default:
252     return obj != null ? obj.toString() : void 0;
253 }
254 })();
255 parentTag = getParent(atTag);
256 separator(parentTag);
257 addElement(parentTag, "" + symbol + " " + msg + "<br>");
258 return obj;
259 };
260 showDocumentHere = function(atTag, content, width, height) {
261     var parentTag;
262     if (width == null) width = 300;
263     if (height == null) height = 300;
264     parentTag = getParent(atTag);
265     separator(parentTag);
266     addFrame(parentTag, width, height, content);
267 };
268 evaluateSource = function(field) {
269     var HtmlListener, addErrorElement, codeSegment, codeTag, confirm, currentSegment, currentTag, elem, getCode, getCurrentTag;
270     if (field == null) field = null;
271     show = function(obj, shallow, symbol) {
272         if (shallow == null) shallow = false;
273         if (symbol == null) symbol = '&rarr;';
274         showHere(currentTag, obj, shallow, symbol);
275     };
276     view = function(obj, shallow, symbol) {
277         if (shallow == null) shallow = false;
278         if (symbol == null) symbol = '&rarr;';
279         return showHere(currentTag, obj, shallow, symbol);
280     };
281     showDocument = function(content, width, height) {
282         if (width == null) width = 300;
283         if (height == null) height = 300;
284         return showDocumentHere(currentTag, content, width, height);
285     };
286     addErrorElement = function(text) {
287         return show("<span class='er'>" + text + "</span>");
288     };
289     getCurrentTagId = function(id) {
290         return id.match(/button-\w+-(.*)/)[1];
291     };
292     runOnDemand = function(func) {
293         show("<button id='button-run-' + currentTag.id + '> Run </button>");
294         document.getElementById("button-run-" + currentTag.id).onclick = function() {
295             var currentTag;
296             currentTag = document.getElementById(getCurrentTagId(this.id));
297             view = show = function(obj, shallow, symbol) {
298                 if (shallow == null) shallow = false;
299                 if (symbol == null) symbol = '&rArr;';
300                 return showHere(currentTag, obj, shallow, symbol);
301             };
302             showDocument = function(content, width, height) {
303                 if (width == null) width = 300;
304                 if (height == null) height = 300;
305                 return showDocumentHere(currentTag, content, width, height);
306             };
307             return func();
308         };
309     };
310     confirm = function(message, func) {
311         show(("" + message) + (" <button id='button-yes-' + currentTag.id + '> Yes </button>") + (" <button id='button-no-' +
312         document.getElementById("button-yes-" + currentTag.id).onclick = function() {
313             var currentTag;
314             currentTag = document.getElementById(getCurrentTagId(this.id));
315             view = show = function(obj, shallow, symbol) {
316                 if (shallow == null) shallow = false;
317                 if (symbol == null) symbol = '&rArr;';
318                 return showHere(currentTag, obj, shallow, symbol);

```

```

319     };
320     showDocument = function(content, width, height) {
321         if (width == null) width = 300;
322         if (height == null) height = 300;
323         return showDocumentHere(currentTag, content, width, height);
324     };
325     return func(true);
326 };
327 document.getElementById("button-no-" + currentTag.id).onclick = function() {
328     var currentTag;
329     currentTag = document.getElementById(getCurrentTagId(this.id));
330     view = show = function(obj, shallow, symbol) {
331         if (shallow == null) shallow = false;
332         if (symbol == null) symbol = '&Arr;';
333         return showHere(currentTag, obj, shallow, symbol);
334     };
335     showDocument = function(content, width, height) {
336         if (width == null) width = 300;
337         if (height == null) height = 300;
338         return showDocumentHere(currentTag, content, width, height);
339     };
340     return func(false);
341 };
342 };
343 prompt = function(message, defaultValue, func) {
344     show(("" + message) + (" <input id='input-prompt-" + currentTag.id + "' value='" + defaultValue + "' />") + (" <button
345     document.getElementById("button-prompt-" + currentTag.id).onclick = function() {
346         var currentTag;
347         currentTag = document.getElementById(getCurrentTagId(this.id));
348         view = show = function(obj, shallow, symbol) {
349             if (shallow == null) shallow = false;
350             if (symbol == null) symbol = '&Arr;';
351             return showHere(currentTag, obj, shallow, symbol);
352         };
353         showDocument = function(content, width, height) {
354             if (width == null) width = 300;
355             if (height == null) height = 300;
356             return showDocumentHere(currentTag, content, width, height);
357         };
358         return func(document.getElementById("input-prompt-" + (getCurrentTagId(this.id))).value);
359     };
360 };
361 HtmlListener = (function(_super) {
362
363     __extends(HtmlListener, _super);
364
365     function HtmlListener(maxCollected) {
366         this.maxCollected = maxCollected != null ? maxCollected : 10;
367     }
368
369     HtmlListener.prototype.log = function(str) {
370         return show(str);
371     };
372
373     HtmlListener.prototype.passed = function(str) {
374         return show("<span class='kw'" + str + "</span>");
375     };
376
377     HtmlListener.prototype.invalid = function(str) {
378         return show("<span class='dt'" + str + "</span>");
379     };
380
381     HtmlListener.prototype.failure = function(str) {
382         return show("<span class='al'" + str + "</span>");
383     };
384
385     HtmlListener.prototype.done = function() {
386         show('Completed test. ');
387         return resetProps();
388     };
389
390     return HtmlListener;

```



```

391
392    })(ConsoleListener);
393    Case.prototype.note = function(a) {
394        this.noteArg(a);
395        return a;
396    };
397    Case.prototype.noteVerbose = function(a) {
398        this.noteArg(a);
399        show(this.args);
400        return a;
401    };
402    testPure = function(func, types, name, property) {
403        declare(name, types, function() {
404            var a, c;
405            c = arguments[0], a = 2 <= arguments.length ? __slice.call(arguments, 1) : [];
406            return c.assert(property.apply(null, [c].concat(__slice.call(a), [c.note(func.apply(null, a))]]));
407        });
408    };
409    qcConfig = new Config(100, 1000);
410    test = function(msg, func) {
411        return _.each([msg, func, runAllProps(qcConfig, new HtmlListener)], function(o) {
412            if (!_.isUndefined(o)) return show(o);
413        });
414    };
415    getCode = function(codeTag) {
416        var code, segment;
417        if (codeTag.value != null) {
418            segment = codeTag.value;
419        } else {
420            segment = codeTag.innerHTML;
421        }
422        code = segment.replace(/<br>/g, '\n');
423        code = code.replace(/<[/\]?span[^\>]*>/g, '');
424        code = code.replace(/&]gt;/g, '>');
425        code = code.replace(/&]lt;/g, '<');
426        code = code.replace(/&]nbsp;/g, ' ');
427        return {
428            codeTag: codeTag,
429            code: code
430        };
431    };
432    if ((field != null) && this.incrementalEvaluation) {
433        segments = [getCode(field)];
434    } else {
435        segments = (function() {
436            var _i, _len, _ref, _results;
437            _ref = window.document.getElementsByTagName('pre');
438            _results = [];
439            for (_i = 0, _len = _ref.length; _i < _len; _i++) {
440                codeSegment = _ref[_i];
441                if (codeSegment.className === 'sourceCode') {
442                    codeTag = codeSegment.firstChild;
443                    if (codeTag.className === 'sourceCode coffeescript' || codeTag.className === 'sourceCode CoffeeScript') {
444                        _results.push(getCode(codeTag));
445                    } else {
446                        _results.push(void 0);
447                    }
448                } else {
449                    _results.push(void 0);
450                }
451            }
452            return _results;
453        })();
454        segments = (function() {
455            var _i, _len, _results;
456            _results = [];
457            for (_i = 0, _len = segments.length; _i < _len; _i++) {
458                segment = segments[_i];
459                if (segment != null) _results.push(segment);
460            }
461            return _results;
462        })();

```

```

463     for (i = 0, _len = segments.length; i < _len; i++) {
464         segment = segments[i];
465         if (i > 0 && /^[\\s]/.test(segment.code)) {
466             segment.code = segments[i - 1].code + '\\n' + segment.code;
467             segments[i - 1] = void 0;
468         }
469     }
470     segments = (function() {
471         var _i, _len2, _results;
472         _results = [];
473         for (_i = 0, _len2 = segments.length; _i < _len2; _i++) {
474             segment = segments[_i];
475             if (segment != null) _results.push(segment);
476         }
477         return _results;
478     })();
479 }
480 for (_i = 0, _len2 = segments.length; _i < _len2; _i++) {
481     segment = segments[_i];
482     outList = getParent(segment.codeTag).getElementsByClassName('output');
483     outLength = outList != null ? outList.length : void 0;
484     while (outList && outLength > 0) {
485         elem = outList[--outLength];
486         getParent(elem).removeChild(elem);
487     }
488 }
489 incredibleLength = segments.length;
490 _fn = function(currentTag) {
491     var draw, incredibleResult, magic, promoteToGlobal, _j, _len3, _ref;
492     try {
493         draw = void 0;
494         incredibleResult = eval(CoffeeScript.compile(currentSegment.code, {
495             bare: true
496         }));
497         if (draw != null) drawCanvas(draw);
498         if (incredibleResult !== void 0 && typeof incredibleResult !== 'function') {
499             show(incredibleResult, true, '&crarr;');
500         }
501         if (typeof globalNames !== "undefined" && globalNames !== null) {
502             promoteToGlobal = function(magic) {
503                 var magicValue;
504                 try {
505                     magicValue = eval(magic);
506                 } catch (error) {
507                     return;
508                 }
509                 return this[magic] = magicValue;
510             };
511             _ref = globalNames.split(' ');
512             for (_j = 0, _len3 = _ref.length; _j < _len3; _j++) {
513                 magic = _ref[_j];
514                 promoteToGlobal(magic);
515             }
516         }
517     } catch (error) {
518         console.log(error.message);
519         addErrorElement(error);
520     }
521 };
522 for (incredibleIndex = 0; incredibleIndex < incredibleLength; incredibleIndex += 1) {
523     incredibleStart = new Date();
524     currentSegment = segments[incredibleIndex];
525     currentTag = currentSegment.codeTag;
526     if (currentTag.id === '') currentTag.id = "codeTag" + incredibleIndex;
527     _fn(currentTag);
528     if ((typeof showTiming !== "undefined" && showTiming !== null) && showTiming === true) {
529         incredibleTime = Number(0.001 * (new Date() - incredibleStart)).toFixed(3);
530         show("" + incredibleTime + "s", true, ' ');
531     }
532 }
533 };
534 keyDownEvaluation = function(evt) {

```

```

535     var _ref;
536     if (evt == null) return;
537     if ((typeof localStorage !== "undefined" && localStorage !== null ? localStorage.evaluation : void 0) === 'manual') {
538         if (evt.keyCode === 13 && evt.shiftKey) {
539             evt.preventDefault();
540             evaluateSource(evt.currentTarget);
541         }
542     }
543     if (evt.keyCode === 13 && !evt.shiftKey) {
544         if ((_ref = evt.target) != null) _ref.rows++;
545     }
546 };
547 keyUpEvaluation = function(evt) {
548     var _ref;
549     if (evt == null) return;
550     if ((typeof localStorage !== "undefined" && localStorage !== null ? localStorage.evaluation : void 0) !== 'manual') {
551         if ((_ref = evt.keyCode) !== 16 && _ref !== 37 && _ref !== 38 && _ref !== 39 && _ref !== 40) {
552             return evaluateSource(evt.currentTarget);
553         }
554     }
555 };
556 cmEvaluation = function(editor) {
557     editor.save();
558     return evaluateSource(editor.getTextArea());
559 };
560 cmAutoEvaluation = function(editor) {
561     if ((typeof localStorage !== "undefined" && localStorage !== null ? localStorage.evaluation : void 0) !== 'manual') {
562         return cmEvaluation(editor);
563     }
564 };
565 cmManualEvaluation = function(editor) {
566     if ((typeof localStorage !== "undefined" && localStorage !== null ? localStorage.evaluation : void 0) === 'manual') {
567         return cmEvaluation(editor);
568     }
569 };
570 createEditor = function(codeElement, text) {
571     var c, countLines, elemCodeMirror, newelem;
572     newelem = document.createElement('textarea');
573     newelem.setAttribute('id', codeElement.id);
574     newelem.setAttribute('class', codeElement.getAttribute('class'));
575     countLines = ((function() {
576         var _i, _len, _results;
577         _results = [];
578         for (_i = 0, _len = text.length; _i < _len; _i++) {
579             c = text[_i];
580             if (c === '\n') _results.push(c);
581         }
582         return _results;
583     })()).length + 1;
584     newelem.setAttribute('rows', countLines);
585     newelem.setAttribute('style', 'width: 98%;');
586     newelem.setAttribute('autofocus', 'true');
587     newelem.setAttribute('spellcheck', 'false');
588     newelem.innerHTML = text;
589     newelem.addEventListener('keydown', keyDownEvaluation, false);
590     newelem.addEventListener('keyup', keyUpEvaluation, false);
591     getParent(codeElement).replaceChild(newelem, codeElement);
592     if (this.useCodeMirror) {
593         return elemCodeMirror = CodeMirror.fromTextArea(newelem, {
594             onChange: cmAutoEvaluation,
595             extraKeys: {
596                 'Shift-Enter': cmManualEvaluation
597             },
598             lineNumbers: true,
599             theme: 'pantheme'
600         });
601     }
602 };
603 activateEditor = function() {
604     var sourcecode;
605     this.removeEventListener('focus', activateEditor, false);
606     sourcecode = this.innerHTML.toString().replace(/<\/?span[^\>]*>/g, '');

```

```

607         sourcecode = sourcecode.replace(/<br[ ]*[^>]*>/g, '\n');
608         return createEditor(this, sourcecode);
609     };
610     _ref = document.getElementsByTagName('pre');
611     for (_i = 0, _len = _ref.length; _i < _len; _i++) {
612         segment = _ref[_i];
613         segment.firstChild.addEventListener('focus', activateEditor, false);
614     }
615     this.incrementalEvaluation = false;
616     return evaluateSource();
617 };
618 });
619 };
620
621 kup = typeof exports !== "undefined" && exports !== null ? require('coffeekup') : window.CoffeeKup;
622
623 display = typeof exports !== "undefined" && exports !== null ? console.log : function(html) {
624     return show(_.$escape(html));
625 };
626
627 display(kup.render(webfragment, {
628     format: true
629 }));
630
631 }).call(this);

```

Formats [Standalone CoffeeScript](#) [Markdown](#) [PDF](#) [HTML](#)

License [Creative Commons Attribution Share Alike](#) by autotelicum © 2555/2012