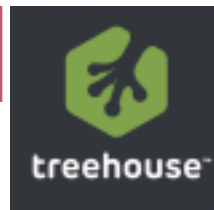


Learn

Handlebars.js Part 3: Tips and Tricks



Treehouse

writes on April 13, 2011



So we're about done learning about Handlebars.js, but I've got a few examples of tricks I've found while using Handlebars that I figured I would share. If you haven't read them yet, I'd highly recommend reading [Part 1](#) and [Part 2](#) of the series on Handlebars before you dive into this post.

The `debug` Helper

It can be easy at times to get confused about where you are in the stack in a Handlebars.js template. I usually keep a `debug` helper around to help me figure that out.

```
1 Handlebars.registerHelper("debug", function(optionalValue) {  
2   console.log("Current Context");  
3   console.log("=====");  
4   console.log(this);  
5  
6   if (optionalValue) {  
7     console.log("Value");  
8     console.log("=====");  
9     console.log(optionalValue);  
10  }  
11 });
```

You could then use that helper in any template like:

```
1 {{debug}}
```

or like:

```
1 {{debug someValue}}
```

and you'll see output in the JavaScript console letting you know what's going on:

```
1 Current Context
2 =====
3 email: "alan@test.com"
4 first_name: "Alan"
5 last_name: "Johnson"
6 member_since: "Mar 25, 2011"
7 phone: "1234567890"
8 stripeClass: "even"
9 __proto__: Object
10 Value
11 =====
12 Alan
```

A jQuery Plugin

It tends to be a pretty regular pattern with Handlebars to compile a template, process it with some data, and then fill a DOM element with the results. Here's a quick jQuery plugin that does all of that for you:

```
1 <html>
2   <head>
3     <title>What's Up?</title>
4   </head>
5   <body>
6     <div id="content">
```

```

7       This is where the content goes!
8     </div>
9     <script id="template" type="text/handlebars">
10       Hello, {{name}}!
11     </script>
12     <script type="text/javascript" src="jquery.js"></script>
13     <script src="handlebars-0.9.0.pre.5.js" type="text/javascript"></
14     <script type="text/javascript">
15       (function($) {
16         var compiled = {};
17         $.fn.handlebars = function(template, data) {
18           if (template instanceof jQuery) {
19             template = $(template).html();
20           }
21
22           compiled[template] = Handlebars.compile(template);
23           this.html(compiled[template](data));
24         };
25       })(jQuery);
26
27       $('#content').handlebars($('#template'), { name: "Alan" });
28     </script>
29   </body>
30 </html>

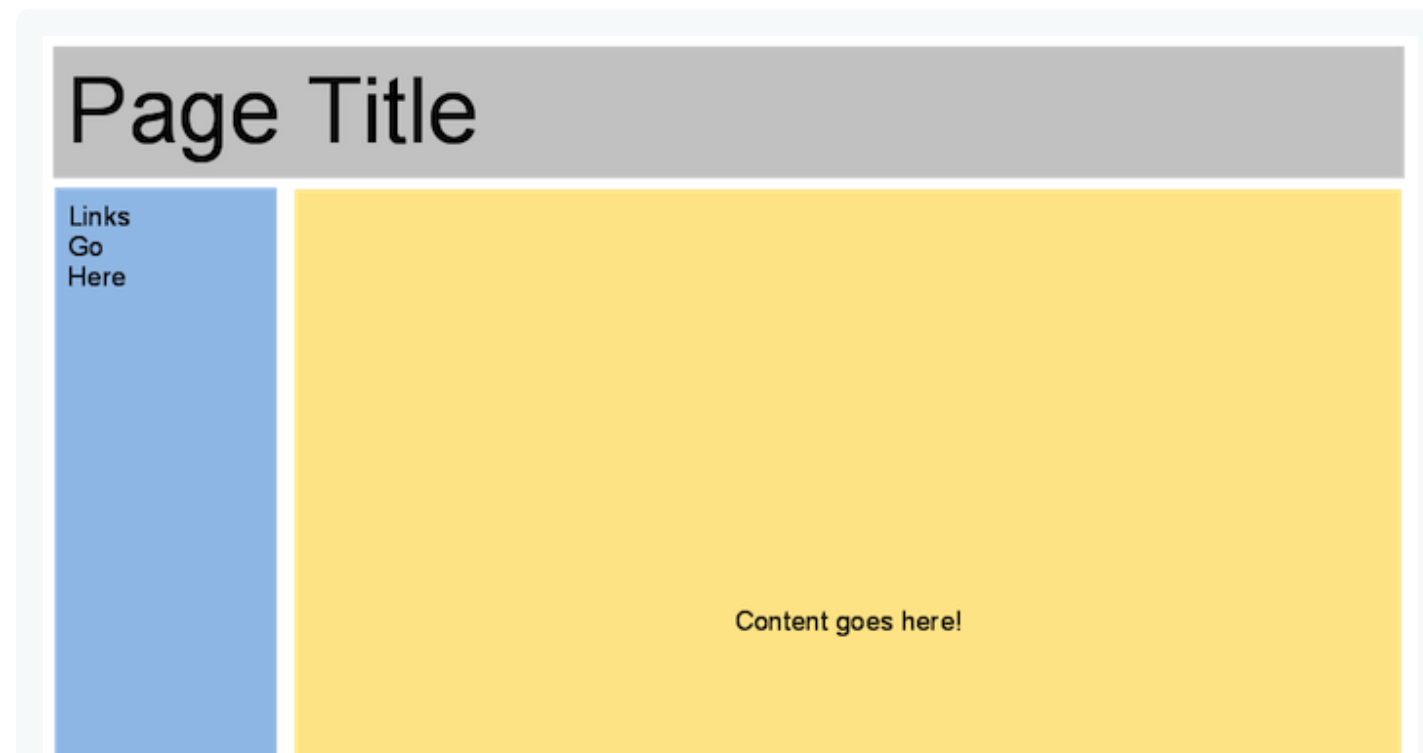
```

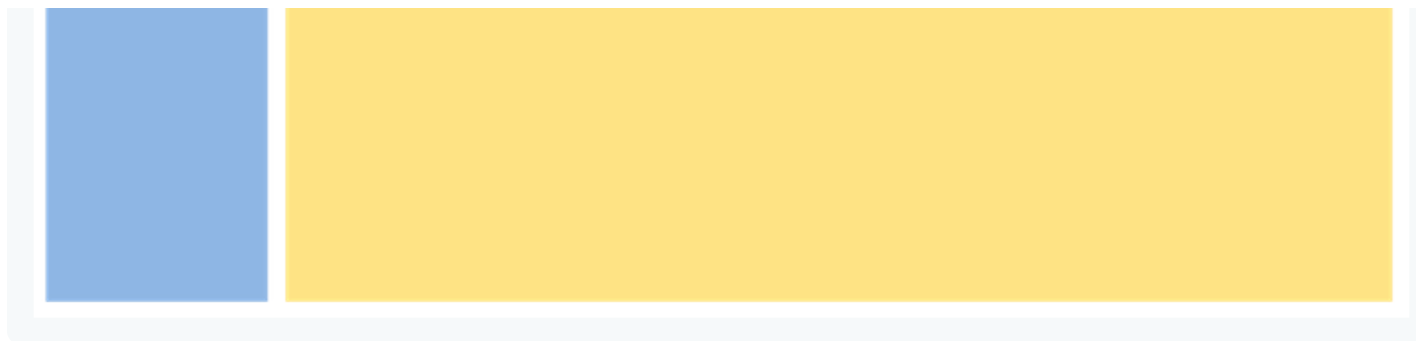
Updating More Than Just The Template Area

A lot of dynamic applications tend to have a layout that generally stays the same regardless of the template you're displaying, and

then a content area that shows the template that you're looking at. Maybe there's a sidebar with some navigation in it that generally stays the same, but each time you update the main content area you also need to update the title on the page. It's pretty easy to cheat a bit with Handlebars to make that work. Handlebars helpers are allowed to not return anything, in which case they won't add anything to the template output. You can use a helper that doesn't return anything to update other parts of the page!

Imagine a site that has a structure something like this:





And maybe the markup looks like this:

```
1 <body>
2   <h1>Title</h1>
3   <div id="nav-sidebar">
4     <ul>
5       <li>Links</li>
6       <li>Go</li>
7       <li>Here</li>
8     </ul>
9   </div>
10  <div id="content">
11    Content goes here!
12  </div>
13 </body>
```

Here's a quick helper that will let you update the `h1` for the title each time you apply content to the main content area:

```
1 Handlebars.registerHelper("title", function(fn) {
2   $('h1').html(fn(this));
3
4   return "";
```

```
5 });
```

And finally, a template that uses the helper:

```
1  {{#title}}Details for {{name}}{{/title}}
2
3  <div class="user-details">
4    ...user details template stuff goes here...
5  </div>
```

When the template above is run, it will call the title helper, but since the title helper just returns an empty string, it won't add anything to the template output. It will, though, use jQuery to update the `h1` element for the page's title.

Wrapping Up

I think this about wraps up the series on Handlebars.js. It's been fun to share Handlebars.js with everyone, and to hear about the apps you're writing with Handlebars.js. Please let me know in the comments if there's anything else relating to Handlebars.js that you'd be interested in hearing about, or if you have any questions.

[code](#)[html and xhtml](#)[javascript](#)[jquery](#)

13 Responses to “Handlebars.js Part 3: Tips and Tricks”

Matt Lewis on [August 18, 2015 at 3:54 pm](#) said:

Thanks for this tip! It helped me a ton debugging my Handlebars templates and rendering order.

Reply

Mike Griffin on [October 5, 2013 at 10:16 pm](#) said:

When I first read the title of your post I thought the `{{debug}}` tag was

going to do a JavaScript “debugger;” statement and I thought, how brilliant, I might create one and see if it helps debugging. I’ve created a Desktop application that allows you to run full handlebar templates on your desktop and was looking at debugging options, that’s how I found your article. You can see our reveal video here http://www.tiraggo.com/movies/M2G_Desktop_Reveal.html

Reply

Jure Mav on **October 2, 2013 at 2:49 pm** said:

Handlebars updated meanwhile. Helpers now pass object with hash parameter if options is not defined. Upper debug helper always show value data in console although options parameter is not passed.

To bypass this error, replace line #6 with:
if (!optionalValue.hash && optionalValue) {

Reply

Anonymous on **May 4, 2011 at 11:11 pm** said:

This is great tips and tricks.

Reply

Dating on **April 30, 2011 at 4:49 pm** said:

The documents, dating from 2002 to 2008, detail accounts of prisoners at the infamous US ...

Reply

Jamesenchaca Jamesenchaca on **April 25, 2011 at 6:20 am** said:

it provide the code of software , but many times is can not undestnddable.

<http://www.articlespeak.com/lavalife-free-trial-log-in-to-find-love-on-a-dating-site/>

Reply

Joost Schuur on **April 24, 2011 at 7:06 pm** said:

In your jQuery plugin example, what are you accomplishing by using the 'compiled' object in 'compiled[template] =

Handlebars.compile(template);'? Why not just use a simple variable?

I thought you might be attempting to cache the compiled template, but I don't see a check against an existing cached item there.

Reply

Renaestbrooks **Renaestbrooks** on **April 21, 2011 at 6:17 am** said:

i want study any any programming language but i can say in this time demand is .NET .

<http://www.articlespeak.com/wall-street-journal-subscription-discount-best-deals/>

Reply

redidas on **April 15, 2011 at 5:14 am** said:

Great timing with these articles! I just started toying with displaying Tumblr JSON via handlebars, but have run into troubles.

Expressions with a dash in them don't seem to work, like {{regular-title}} for example. Is this just a limitation with the handlebar/mustache expression?

Reply

Anonymous on **April 15, 2011 at 7:58 pm** said:

Great question. The issue there is that Handlebars expects the expression to use valid JavaScript variable names. So anything with a dash wouldn't be valid.

Reply

Anonymous on [April 13, 2011 at 6:03 pm](#) said:

Wow, pretty impressive stuff dude. Well done.

<http://www.total-privacy.it.tc>

Reply

Joey Lomanto on [April 13, 2011 at 5:24 pm](#) said:

Alan these articles have been awesome. Thanks for taking the time to put them together 😊

Reply

Anonymous on [April 13, 2011 at 5:32 pm](#) said:

Thanks so much, Joey. Feel free to let me know if there's anything else you'd like me to write about:

alan+tv@carsonified.com

Reply

Leave a Reply

Name *

Email Address (will not be published) *

Website

Submit Comment



Want to learn more about Javascript?

Learn how to use JavaScript to
add interactivity to websites.

[Learn more](#)

Stay current

Sign up for our newsletter, and we'll send you
news and tutorials on web design, coding,
business, and more!

Email Address

Subscribe



©2016 Treehouse Island, Inc.

[About](#) • [Careers](#) • [Blog](#) • [Affiliate Program](#) • [Terms](#) • [Privacy](#) • [Press Kit](#) • [Contact](#)



