

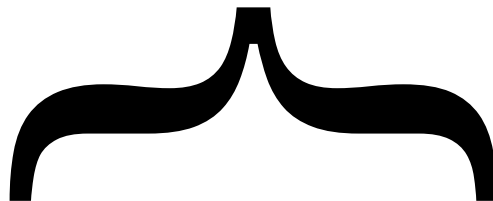
# Christophe Coenraets

The Web Platform and Mobile Application Development

Navigation 

## Tutorial: HTML Templates with Mustache.js

by **Christophe Coenraets** on December 20, 2011 in **HTML 5, JavaScript, JQuery, Mustache.js, Tutorial**



When developing modern HTML applications, you often write a lot of HTML fragments programmatically. You concatenate HTML tags and dynamic data, and insert the resulting UI markup into the DOM. Here is a random code example of this approach:

```
1 $.each(messages.reverse(), function(index, message) {  
2     $('#messageList').append(  
3         '<li><span class="list-title">' +  
4         message.userName + '</span>' +  
5         '<abbr class="list-timestamp" title="' +  
6         message.datePosted + '></abbr>' +  
7         '<p class="list-text">' + message.messageText + '</p></li>');  
8     }  
9 });
```

The proliferation of this kind of code throughout your application comes with some downsides. The tight coupling of UI and data logic doesn't promote separation of concerns and reuse. It makes your application harder to write and harder to maintain.

HTML templates address this issue by decoupling the UI definition (HTML markup) from the data. There are a number of HTML template solutions out there: jQuery Templates, Underscore.js, and [Mustache.js](#) to name a few. Mustache.js is a popular choice because of its powerful syntax and fast rendering.

[Mustache](#) is a “logic-less” template syntax. “Logic-less” means that it doesn't rely on procedural statements (if, else, for, etc.): Mustache templates are entirely defined with tags. Mustache is implemented in different languages: Ruby, JavaScript, Python, PHP, Perl, Objective-C, Java, .NET, Android, C++, Go, Lua, Scala, etc. [Mustache.js](#) is the JavaScript implementation.

In this article, we take a quick tour of some of the capabilities of Mustache.js.

To start using Mustache.js, simply add a script tag to your html file pointing to mustache.js which is available [here](#).

You can run all the examples below [here](#).

## Sample 1: Basic Template

This is a self-explanatory example. Note that:

- Instead of being defined in a variable, the data often comes from a service call (see sample 2)
- Instead of being defined in a variable, the template is often read from a file (see sample 3)

```
1 var person = {
2   firstName: "Christophe",
3   lastName: "Coenraets",
4   blogURL: "http://coenraets.org"
5 };
6 var template = "<h1>{{firstName}} {{lastName}}</h1>Blog: {{blogURL}}";
7 var html = Mustache.to_html(template, person);
8 $('#sampleArea').html(html);
```

Result:

# Christophe Coenraets

Blog: <http://coenraets.org>

## Sample 2: Basic Template using Ajax data

Same as sample 1, except that we get the data from an Ajax service call.

```
1 $.getJSON('json/data.json', function(data) {
2   var template = "<h1>{{firstName}} {{lastName}}</h1>Blog: {{blogURL}}";
3   var html = Mustache.to_html(template, data);
4   $('#sampleArea').html(html);
5 });
```

Result:

# John Smith

Blog: <http://johnsmith.com>

## Sample 3: Externalized Template

Same as sample 2, except that we read the template from the main HTML file.

```
1 $.getJSON('json/data2.json', function(data) {
2   var template = $('#personTpl').html();
3   var html = Mustache.to_html(template, data);
4   $('#sampleArea').html(html);
5 });
```

The template is defined as follows in index.html:

```
1 <script id="personTpl" type="text/template">
2 <h1>{{firstName}} {{lastName}}</h1>
```

```
3 | <p>Blog URL: <a href="{{blogURL}}">{{blogURL}}</a></p>
4 | </script>
```

Result:

# Lisa Jones

Blog URL: <http://lisajones.com>

**NOTE:** Sample 3 represents the way templates are being used in many dynamic Web applications:

- You get data from an Ajax service
- You read the template from an external file

In the remaining of this article, we declare the data and the template in variables to keep the examples self-contained. Remember to refer to sample 3 for a traditional setup when using templates in a dynamic Web application.

## Sample 4: Enumerable Section

```
1 | var data = {name: "John Smith", skills: ['JavaScript', 'PHP', 'Java']};
2 | var tpl = "{{name}} skills:<ul>{{#skills}}<li>{{.}}</li>{{/skills}}</ul>";
3 | var html = Mustache.to_html(tpl, data);
4 | $('#sampleArea').html(html);
```

Result:

John Smith skills:

- JavaScript
- PHP
- Java

## Sample 5: Enumerable Section with Objects

```
1 | var data = {
2 |   employees: [
3 |     { firstName: "Christophe",
4 |       lastName: "Coenraets"},
5 |     { firstName: "John",
6 |       lastName: "Smith"}
7 |   ]};
8 | var template = "Employees:<ul>{{#employees}}<li>{{firstName}} {{lastName}}</li>{{/employees}}</ul>";
9 |
10 |
11 | var html = Mustache.to_html(template, data);
12 | $('#sampleArea').html(html);
```

Result:

Employees:

- Christophe Coenraets
- John Smith

## Sample 6: Nested Objects

You can use the dot notation to access object properties.

```
1  var person = {
2    firstName: "Christophe",
3    lastName: "Coenraets",
4    blogURL: "http://coenraets.org",
5    manager : {
6      firstName: "John",
7      lastName: "Smith"
8    }
9  };
10 var template = "<h1>{{firstName}} {{lastName}}</h1><p>{{blogURL}}</p>" +
11               "Manager: {{manager.firstName}} {{manager.lastName}}";
12 var html = Mustache.to_html(template, person);
13 $('#sampleArea').html(html);
```

Result:

# Christophe Coenraets

http://coenraets.org

Manager: John Smith

## Sample 7: Dereferencing

Same as sample 6, except that we “dereference” the manager object to make it easier to access its properties (without having to use the dot notation).

```
1  var person = {
2    firstName: "John",
3    lastName: "Smith",
4    blogURL: "http://johnsmith.com",
5    manager : {
6      firstName: "Lisa",
7      lastName: "Jones"
8    }
9  };
10 var tpl = "<h1>{{firstName}} {{lastName}}</h1><p>{{blogURL}}</p>" +
11          "{{#manager}}Manager: {{firstName}} {{lastName}}{{/manager}}";
12 var html = Mustache.to_html(tpl, person);
13 $('#sampleArea').html(html);
```

Result:

# John Smith

http://johnsmith.com

Manager: Lisa Jones

## Sample 8: Function

Templates can reference functions like `totalPrice` in this example.

```
1  var product = {
2    name: "FooBar",
3    price: 100,
4    salesTax: 0.05,
5    totalPrice: function() {
6      return this.price + this.price * this.salesTax;
7    }
8  };
9  var template = "<p>Product Name: {{name}}</p>Price: {{totalPrice}}";
10 var html = Mustache.to_html(template, product);
11 $('#sampleArea').html(html);
```

Result:

Product Name: FooBar

Price: 105

## Sample 9: Condition

Templates can include conditional sections. Conditional sections only render if the condition evaluates to true. A conditional section begins with `{{#condition}}` and ends with `{{/condition}}`. "condition" can be a boolean value or a function returning a boolean.

```
1  var data = {
2    employees: [
3      { firstName: "Christophe",
4        lastName: "Coenraets",
5        fullTime: true,
6        phone: "617-123-4567"
7      },
8      { firstName: "John",
9        lastName: "Smith",
10       fullTime: false,
11       phone: "617-987-6543"
12     },
13     { firstName: "Lisa",
14       lastName: "Jones",
15       fullTime: true,
16       phone: "617-111-2323"
17     },
18   ];
19  var tpl = "Employees:<ul>{{#employees}}<li>{{firstName}} {{lastName}} " +
20    "{{#fullTime}} {{phone}}{{/fullTime}}</li>{{/employees}}</ul>";
21  var html = Mustache.to_html(tpl, data);
22  $('#sampleArea').html(html);
```

Result:

Employees:

- Christophe Coenraets 617-123-4567

- John Smith
- Lisa Jones 617-111-2323

## Sample 10: Partial

```
1  var data = {
2    firstName: "Christophe",
3    lastName: "Coenraets",
4    address: "1 Main street",
5    city: "Boston",
6    state: "MA",
7    zip: "02106"
8  };
9
10 var template = "<h1>{{firstName}} {{lastName}}</h1>{{>address}}";
11 var partials = {address: "<p>{{address}}</p>{{city}}, {{state}} {{zip}}"};
12 var html = Mustache.to_html(template, data, partials);
13 $('#sampleArea').html(html);
```

Result:

# Christophe Coenraets

1 Main street

Boston, MA 02106

## Sample 11: Partial in Enumerable Section

```
1  var data = { depts: [
2    { name: "Engineering",
3      employees: [
4        {firstName: "Christophe", lastName: "Coenraets"},
5        {firstName: "John", lastName: "Smith"}]
6    },
7    { name: "Sales",
8      employees: [
9        {firstName: "Paula", lastName: "Taylor"},
10       {firstName: "Lisa", lastName: "Jones"}]
11   }]
12 };
13
14 var tpl = "{{#depts}}<h1>{{name}}</h1> " +
15           "<ul>{{#employees}}<li>{{employee}}</li>{{/employees}}</ul>{{/depts}}";
16 var partials = {employee: "<li>{{firstName}} {{lastName}}</li>"};
17 var html = Mustache.to_html(tpl, data, partials);
18 $('#sampleArea').html(html);
```

Result:

# Engineering

- Christophe Coenraets
- John Smith

# Sales

- Paula Taylor
- Lisa Jones

You can run all the examples [here](#).

Follow @ccoenraets

4,752 followers

Tweet

102

46

Share

Like

38

## Subscribe

### Related Posts:

[PhoneGap and PhoneGap Build in 5 Minutes](#)

[New Tutorial: Developing and Architecting a PhoneGap Application](#)

[Creating a Dropbox Integration Extension for Brackets](#)

[Exploring and Contributing to Web Platform Docs](#)

[NodeCellar: Sample Application with Backbone.js, Twitter Bootstrap, Node.js, Express, and MongoDB](#)

◀ [Fact Check Michele Bachmann with Politifact and Flex](#)   [Setting Up WordPress on Amazon EC2 in 5 minutes](#) ▶

## 34 Responses to *Tutorial: HTML Templates with Mustache.js*



**Chris** January 6, 2012 at 8:37 am #

Thanks for this post. It's an excellent cheat sheet for mustache! Much appreciated.

REPLY



**Omar Abdallah** January 12, 2012 at 2:38 pm #

Pretty nice engine, was looking for something like that, thanks alot

REPLY



**Larry** January 23, 2012 at 4:47 am #

Do you have an example of doing recursion? I have an object like this:

```
{ "name": "bob",  
  "friend": [{ "name": "luke"},  
    { "name": "steve",  
      "friend": [{ "name": "Pat"},  
        { "name": "Dave"}  
      ]  
    }  
  ]  
}
```

I want to see them in a list like this:

```
bob  
luke  
steve  
Pat  
Dave
```

REPLY



**Larry** January 23, 2012 at 4:49 am #

That didn't show up very good I want to see the output like this:

```
bob  
--luke  
--steve  
---Pat  
---Dave
```

REPLY



**Victor** February 21, 2012 at 9:51 pm #

Nice work! Help me a lot! thanks

REPLY





**Jan** March 9, 2012 at 6:41 pm <#>

Is putting templates in tags as mentioned in sample3 supported in vanilla Mustache? I'm asking, because I was under the impression that this you would need something like ICanHaz.js for this.

It's great if Mustache supports this out-of-the-box, since ICanHaz is significantly slower. At least for what I have been using it for.

REPLY



**Bill Heaton** March 10, 2012 at 11:47 pm <#>

I keep coming back to this article and sending other to so the same when learning Mustache. The examples are source code are great, thanks.

REPLY



**Chintan** March 29, 2012 at 9:36 pm <#>

Is there a way in Mustache to get the index of element in array?

REPLY



**alex** April 4, 2012 at 10:36 am <#>

Here's an example how to separate the templates from the rest of the code:

```
{{title}}
```

```
{{content}}
```

```
var template = $('#template1').html();
```

```
...
```

Cheers!

REPLY



**miss Red** April 10, 2012 at 8:51 pm <#>

very usefull. i m trying to figure out whats abbout jvs

REPLY



**Joe** May 1, 2012 at 4:45 pm #

What is the difference between `mustache.render` and `mustache.to_html`? Can't seem to find this info anywhere.

REPLY



**Yasin Coşkun** May 24, 2012 at 11:24 am #

nice tutorial thanx

REPLY



**Landon** June 11, 2012 at 9:03 pm #

I am working on a project that uses `brunch.io` and `mustache` for templating. I want to add `mustache` partials to the main index (which serves as a 'shell') to load in my content. I create the `mustache` partials in their own `.mustache` file and then add the tag on the index page (ie: `{{>somePartial}}`) but what I seem to be missing is how to get the partial to render... That's what I need help with.

REPLY



**madhuka** June 19, 2012 at 9:45 am #

Thanks, good post

REPLY



**raj** June 21, 2012 at 6:03 am #

Hi how can we handle xml value say my json value is

```
var person = {  
  key : (value is xml )  
  raj  
  dugar  
}  
};
```

how can i get only value of tag i.e dugar from this json response using template

any help will be appreciated

REPLY

**Nitesh** July 28, 2012 at 6:07 pm #

You explained all cases... really helped

REPLY

**mauro** September 3, 2012 at 12:58 pm #

Hey thanks for the tutorial! I have a question. How can I load and append in the body tag a Mustache template which is another folder of my website using jQuery?

REPLY

**David** January 14, 2013 at 1:18 pm #

```
$.get(url, function(data) {  
  var template = data;  
  var html = Mustache.to_html(template, person);  
  $('#sampleArea').html(html);  
});
```

REPLY

**Kumaresan** September 20, 2012 at 11:45 am #

This post was very useful. We need one more use case. I want display the html content or render the HTML content in the document.

for Example:

Here is the sample obj.

```
var obj = {  
  fName: "Kumaresan",  
  lName: "",  
  Ph: "1234",  
  Photo: ""  
}
```

I want to display the output is given below:

1. First Output :

— — — — Kumaresan,

----- 1234,

## 2. Second Output:

First Name : "Kumaresan"

Last Name : ""

Phone : "1234"

Photo Url :

How to write a template for above output?

Thanks,

Kumaresan.

REPLY



**Kumaresan** September 20, 2012 at 11:48 am #

The actual value for "Photo Url" : <img src=""<Phot\_url"> />

REPLY



**narun** December 8, 2012 at 5:50 am #

test

REPLY



**Manikandan Ganesan** October 5, 2012 at 1:46 pm #

Quick kick starter for using Mustache and thanks for sharing such a useful tips!

REPLY



**Partha Das** October 17, 2012 at 6:16 pm #

Is it possible to implement paging and sorting with mustache.js??

REPLY



**Tejas Nityanand** November 1, 2012 at 1:45 pm #

Hey,

Very useful tutorials. I have been looking for the implementation of Mustache, and your blog explains it well. I have one problem though, I tried the 'Sample 2: Basic Template using Ajax data' but in my case when I view the page I am getting an error in the Firefox Error Console saying: 'not well-formed' pointing to the JSON file.

I have used the exact same code as shown in the sample.

[REPLY](#)

**Bas Velthuisen** November 14, 2012 at 7:48 pm #

Hello,

Strangely enough, the code in your tutorial is rendered wrong in both firefox and chrome. All html-tags of the beautified code are visible and html-encoded. I don't think that is meant to be, because they are completely unreadable.

Is this a setting in my browsers, or has something changed lately in the libraries that are used?

Cheers,  
Bas

[REPLY](#)

**Minerva** December 29, 2012 at 12:06 pm #

Hello! Someone in my Facebook group shared this site with us so I came to take a look. I'm definitely enjoying the information. I'm bookmarking and will be tweeting this to my followers! Superb blog and superb design.

[REPLY](#)

**social media for small business infographic** January 12, 2013 at 2:24 am #

You must give your marketing methods to be able to work.

In this field, the Australian Institute of Internet Marketing helps people a lot. This really is also advantageous too simply because if someone breaks down inside your area, but aren't from your area, you've the opportunity to beat out your competition for auto repairs on their vehicle, which can bring in several thousand dollars a year alone.

[REPLY](#)

## Trackbacks/Pingbacks

[Javascript Resources\[updated\]](#) « [Kooljoy.com Blog](#) - December 29, 2011

[...] [HTML Templates with Mustache.js](#) [...]

[Howto create Partial – Mustache.js – Marco Senkpiel](#) - May 10, 2012

[...] um die Notation zu finden wenn man das ohne ICanHaz.js etc. machen möchte. Deshalb mal hier ein Codesnippet aus dem Blog von Christophe [...]

[What is the difference between Mustache.render\(\) and Mustache.to\\_html\(\)?](#) - June 3, 2012

[...] documentation makes no mention of Mustache.to\_html(), but every tutorial for Mustache.js online uses Mustache.to\_html(). Therefore I am surely missing some [...]

[Tutorial: HTML Templates with Mustache.js | TechnoVeille](#) - September 3, 2012

[...] original : ici < WysihTML5 Tagués avec : Framework, Html, Javascript Publié dans [...]

[CouchDB Journal – Part 1 Establishing a Dev Environment » JetPgm](#) - October 5, 2012

[...] [Tutorial: HTML Templates with Mustache.js](#) [...]

[หัดใช้ Mustache กันดีกว่า | KHASATHAN.IN.TH](#) - October 30, 2012

[...] [Tutorial: HTML Templates with Mustache.js](#) [...]

[JSON rendering : The Webtech Blog](#) - December 12, 2012

[...] is also MoustacheJS (where do they dream up these names?) – but from what I can see this seems to require JS [...]

## Leave a Reply

Name (required)

Email (will not be published) (required)

Website

Submit Comment

Search...



## About Me



I'm a Technical Evangelist for Adobe where I focus on the Web Platform, Mobile Applications and Enterprise Integration.

[Read More](#)

## Social Networks

POPULAR LATEST COMMENTS TAGS

### Sample App with Backbone.js and Twitter Bootstrap

FEBRUARY 13, 2012

### NodeCellar: Sample Application with Backbone.js, Twitter Bootstrap, Node.js, Express, and MongoDB

OCTOBER 4, 2012

### Creating a REST API using Node.js, Express, and MongoDB

OCTOBER 2, 2012

### Using Backbone.js with jQuery Mobile

MARCH 5, 2012

### Simple Offline Data Synchronization for Mobile Web and PhoneGap Applications

MAY 16, 2012

## Recent Tweets

.@tbranyen @bobchesley NodeCellar is now hosted here:

<http://t.co/cTYASGGy> Source code here: <https://t.co/ojzK6WIt> about 22 hours ago

RT @filmaj: cordova cli tools 2.3.6 available! Support for platform-specific <preference> tags has landed. npm install -g cordova ... 9 days ago

Join #AdobeTechLive Thu 1pm ET. Future of CSS + new HTML tools:

#EdgeAnimate, #Reflow, #EdgeInspect, and #PhoneGap

<http://t.co/sd8h4ohD> 40 days ago

Blogged screencast: PhoneGap and PhoneGap Build in 5 Minutes

<http://t.co/Eo3ketVJ> 41 days ago

Looking forward to presenting with @agreenblatt at #createtheweb Montreal tonight <http://t.co/gVwYkoHB> 52 days ago

Follow @ccoenraets on Twitter

© 2013 Christophe Coenraets. All Rights Reserved.

Powered by [WordPress](#). Designed by  **WOOTHEMES**