Telerik Academy

HTML Templates with Handlebars.js

Making JavaScript code by template!

Telerik Software Academy

Learning & Development Team

http://academy.telerik.com

Telerik Academy

Table of Contents

- HTML Templates
- Handlebars.js Template Engine
  - Setup and usage
- Creating Templates
  - Binding values to HTML
- Handlebars Expressions
  - Block and conditional
- Helpers

Telerik Academy

HTML Templates

Overview

HTML Templates

- HTML templates or template engines:
  - Make the HTML dynamic
    - Allow programming logic inside the otherwise static HTML
  - Provide cleaner and more sustainable way to dynamically create DOM elements
- HTML templates are many:
  - Basic: Handlebars.js, mustache.js, underscore.js, jQuery templates
  - Advanced: KendoUI, AngularJS

Telerik Academy

Handlebars.js

Overview and Setup

Handlebars.js

- Handlebars.js is an open-source template engine
  - Provides both run-time and compiled templates for rendering HTML
  - Allows one-way data-binding of HTML elements to JavaScript objects
  - Produces HTML code based on a template and a given JavaScript object

Setup and Usage

- To use Handlebars follow the steps:
  - Download handlerbars.js
    - From the site at

```
http://handlebarsjs.com/
```

    - Using bower and

```
$ bower install handlebars
```

    - Using NuGet

```
search for "handlebars"
```

  - Src it in HTML file:

```
<script src="path/to/handlebars.js"></script>
```

Setup and Usage

- (cont.) To use Handlebars follow the steps:
  - Create a template
    - In a script tag with invalid type attribute

```
<script id="post-template" type="text/x-handlebars-template">
  <div class='post'>
    <h1 class="post-title">{{title}}</h1>
    <p class="post-content">{{{content}}}</p>
  </div>
</script>
```

  - Render the template

```
var post = {title: '…', content: '…'},
    htmlTemplate = postTemplateNode.innerHTML,
    postTemplate = Handlebars.compile(htmlTemplate),
postNode.innerHTML = postTemplate(post);
```

Setup and Usage

Live Demo

Telerik Academy

Creating HTML Templates with Handlebars.js

Creating HTML Templates

- HTML templates act much like `string.Format()` in C# and Java and `stringWithFormat: @"…"` in Objective-C
  - Put placeholders within a template string, and replace these placeholders with values
- Handlebars.js marks placeholders with double curly brackets `{{value}}`
  - When rendered, the placeholders between the curly brackets are replaced with the corresponding value

Telerik Academy

Creating Templates: Example

- Creating the template

```
<script id="post-template" type="text/x-handlebars-template">
  <div class='post'>
    <h1 class="post-title">{{title}}</h1>
    <p class="post-content">{{content}}</p>
  </div>
</script>
```

- Create the JavaScript object and render the template to produce pure HTML code

```
var post = { title: 'Post Title', content: 'Post content' };
var postTemplateNode = document.getElementById('post-template'),
    postTemplateHtml = templateNode.innerHTML,
    postTemplate = Handlebars.compile(postTemplateHtml);
domNode.innerHTML = postTemplate(post);
```

Creating Templates

Live Demo

HTML Escaping

- Handlebars.js escapes all values before rendering them
- If the value should not be escaped there are two possibilities:
  - Use triple curly brackets in the template string

```
{{{value}}}
```

  - Mark in the value not to be escaped

```
var post = { title: '…',
             content: new Handlebars.SafeString('…')
};
```

    - Mostly used with helpers

Telerik Academy

HTML Escaping

Live Demo

Telerik Academy

Handlebars Expressions

Blocks and Conditionals

Handlebars.js Expressions

- Handlebars.js supports expressions within the templates
  - Block expressions
    - For iterating over a collection of elements
  - Conditional expressions

Block expressions

- Block expressions are used to iterate over a collection of objects (like array)
  - Created using `{{#collection}}` and `{{/collection}}`
    - Everything between will be evaluated for each object in the collection

```
<ul class="categories-list">
  {{#categories}}
    <li class="category-item">
      <a href="#/categories/{{id}}">{{name}}</a>
    </li>
  {{/categories}}
</ul>
```

Block expressions

- Block expressions are used to iterate over a collection of objects (like array)
  - Created using `{{#collection}}` and `{{/collection}}`
    - Everything between will be evaluated for each object in the collection

```
<ul class="categories-list">
  {{#categories}}
    <li class="category-item">
      <a href="#/categories/{{id}}">{{name}}</a>
    </li>
  {{/categories}}
</ul>
```

The id & name of every object

Block Expressions

Live Demo

Telerik Academy

Conditional Expressions

- Render code only if a condition is fulfulled
  - Using `{{#if condition}} {{/if}}` or `{{unless condition}} {{/unless}}`

```
<h1>Posts</h1>
  <ul class="posts">
    {{#posts}}
      <li>
        <strong>
          <a href="#/posts/{{id}}">{{title}}</a>
        </strong>
        {{#if author}}
          <span class="author">by {{author}}</span>
        {{/if}}
      </li>
   {{/posts}}
</ul>
```

Telerik Academy

Conditional Expressions

- Render code only if a condition is fulfulled
  - Using `{{#if condition}} {{/if}}` or `{{unless condition}} {{/unless}}`

```
<h1>Posts</h1>
  <ul class="posts">
    {{#posts}}
      <li>
        <strong>
          <a href="#/posts/{{id}}">{{title}}</a>
        </strong>
        {{#if author}}
          <span class="author">by {{author}}</span>
        {{/if}}
      </li>
    {{/posts}}
</ul>
```

If author is false-like value, this code will not be rendered

Conditional Expressions

Live Demo

Telerik Academy

Handlebars.js Helpers

Handlebars.js Helpers

- Handlebars.js helpers are like functions that do predefined job
- Build-in helpers:
  - `{{#each collection}} {{/each}}`
    - Much like block expressions but with more control like `{{@index}}` and `{{this}}`
  - `{{#with obj}} {{/with}}`
    - Used to minify the path
    - Write `{{prop}}` Instead of `{{obj.prop}}`

Telerik Academy

Using Built-in Handlebars.js Helpers

Live Demo

Telerik Academy

Registering Custom Helpers

- Handlebars.js supports extension of the built-in helpers
  - Using `Handlebars.registerHelper(…)`:

```
Handlebars.registerHelper ('tags', function(tags){
  var html = '<ul class="tags-list">',
      tag;
  for(tag in tags){
    html += '<li class="tag">' + tag + '</li>';
  }
  return new Handlebars.SafeString(html);
});
```

Telerik Academy

Registering Custom Helpers

- Handlebars.js supports extension of the built-in helpers
  - Using `Handlebars.registerHelper(…)`:

```
Handlebars.registerHelper ('tags', function(tags){
  var html = '<ul class="tags-list">',
      tag;
  for(tag in tags){
    html += '<li class="tag">' + tag + '</li>';
  }
  return new Handlebars.SafeString(html);
});
```

SageString is mandatory here

Telerik Academy

Registering Custom Helpers

- Handlebars.js supports extension of the built-in helpers
  - Using `Handlebars.registerHelper(…)`:

```
Handlebars.registerHelper ('tags', function(tags){
  var html = '<ul class="tags-list">',
      tag;
  for(tag in tags){
    html += '<li class="tag">' + tag + '</li>';
  }
  return new Handlebars.SafeString(html);
});
```

SageString is mandatory here

Otherwise the result will be escaped

Telerik Academy

Creating Handlebars.js Helpers

Live Demo

Telerik Academy

HTML Templates with Handlebars.js

Questions?

http://academy.telerik.com

Homework

1. Create the following using Handlebars.js

№	Title	Date #1	Date #2
0	Course Introduction	Wed 18:00, 28-May-2014	Thu 14:00, 29-May-2014
1	Document Object Model	Wed 18:00, 28-May-2014	Thu 14:00, 29-May-2014
2	HTML5 Canvas	Thu 18:00, 29-May-2014	Fri 14:00, 30-May-2014
3	Kinect JS Overview	Thu 18:00, 29-May-2014	Fri 14:00, 30-May-2014
4	SVG and RaphaelJS	Wed 18:00, 04-Jun-2014	Thu 14:00, 05-Jun-2014
5	Animations with Canvas and SVG	Wed 18:00, 04-Jun-2014	Thu 14:00, 05-Jun-2014
6	DOM operations	Thu 18:00, 05-Jun-2014	Fri 14:00, 06-Jun-2014
7	Event model	Thu 18:00, 05-Jun-2014	Fri 14:00, 06-Jun-2014
8	jQuery overview	Wed 18:00, 11-Jun-2014	Thu 14:00, 12-Jun-2014
9	HTML templates	Wed 18:00, 11-Jun-2014	Thu 14:00, 12-Jun-2014
10	Exam preparation	Thu 18:00, 12-Jun-2014	Fri 14:00, 13-Jun-2014
11	Exam	Tue 10:00, 17-Jun-2014	Tue 16:30, 17-Jun-2014
12	Teamwork Defense	Thu 10:00, 19-Jun-2014	Thu 10:00, 19-Jun-2014

Homework (2)

2. Create a dynamic select using Handlebars.js
   - The options in the select should be generated based on a collection of JavaScript objects
   - Example:

```
var items = [{
  value: 1,
  text: 'One'
}, {
  value: 2,
  text: 'Two'
}];

var selectHTML = selectTemplate(items);
```

Homework (3)

3. *Create a jQuery plugin for ListView
   - Apply a template for each item of a collection
   - Using the data-template attribute set the ID of the template to use for the items
   - Must work with different collections and templates
   - *More examples on the next slides

```
<ul id="books-list"
    data-template="book-item-ter
<script id="book-item-template"
  <li class="book-item">
    <a href="/#books/{{id}}">
      <strong>{{title}}</strong>
    </a>
  </li>
</script>
```

- JavaScript: The Good parts
- Secrets of the JavaScript Ninja
- Core HTML5 Canvas
- JavaScript Patterns

```
$('#books-list').listview(books);
```

Homework (4)

3. *(cont.)Create a jQuery plugin for ListView

```
<table>
  <thead>
    <tr><th>#</th><th>Name</th><th>Mark</th><
  </thead>
  <tbody id="students-table"
         data-template="students-row-template">
  </tbody>
</table>

<script id="students-row-template" type="…">
  <tr class="student-row">
    <td>{{number}}</td>
    <td>{{name}}</td>
    <td>{{mark}}</td>
  </tr>
</script>
```

#	Name	Mark
1	Peter Petrov	5.5
2	Stamat Georgiev	4.7
3	Maria Todorova	6
4	Georgi Geshov	3.7

```
$('#students-table').listview(students);
```

Homework (5)

4. *Extend the previous task to set the template inside the DOM element, instead of setting it with data-template

```
<table>
  <thead>
    <tr><th>#</th><th>Name</th><th>Mark</th></tr>
  </thead>
  <tbody id="students-table">
    <tr class="student-row">
      <td>{{number}}</td>
      <td>{{name}}</td>
      <td>{{mark}}</td>
    </tr>
  </tbody>
</table>
```

```
$('#students-table').listview(students);
```

Free Trainings @ Telerik Academy

- "C# Programming @ Telerik Academy
  - csharpfundamentals.telerik.com
- Telerik Software Academy
  - academy.telerik.com
- Telerik Academy @ Facebook
  - facebook.com/TelerikAcademy
- Telerik Software Academy Forums
  - forums.academy.telerik.com