

Warming Up With **ember.js**

Level 1 - Warming Up

Named Templates



Review

We can put HTML inside a Handlebars template

index.html



```
<body>
  <script type='text/x-handlebars'>
    <div class='navbar'>...</div>
    <div class='container'>...</div>
    <footer class='container'>...</footer>
  </script>
</body>
```



What if we wanted to make the content in this template dynamic?

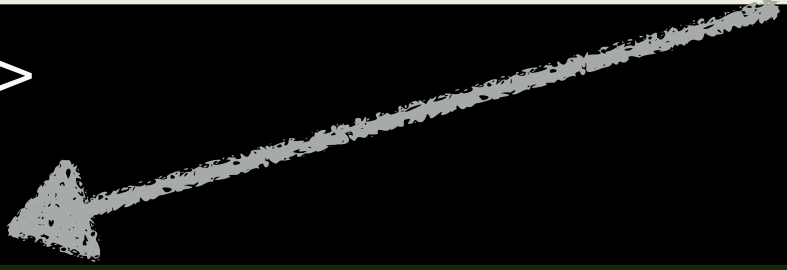


Adding Data to Templates

Called a "handlebars expression"

index.html

```
<script type='text/x-handlebars'>
  <div class='navbar'>...</div>
  <div class='container'>
    <h1>Welcome to {{siteName}}!</h1>
  </div>
  <footer class='container'>...</footer>
</script>
```



{{siteName}}'s value will need to be provided by our Ember app, so it can print The Flint & Flame!

Later on in this course we'll be using templates to add dynamic content.



Our Current Webpage Layout

body



template

div.navbar

div.container

footer.container

We are going to have multiple pages on our website.

- Homepage
- About Page
- Product List



The Different Pages

body



template

div.navbar

div.container

footer.container

About Page

```
<h1>About The Fire Spirits</h1>
```

Home Page

```
<h1>Welcome to The Flint & Flame!</h1>
```

We want this to be the default



We Need to Name our Templates

body



'application' template

div.navbar

div.container

footer.container

'about' template



```
<h1>About The Fire Spirits</h1>
```

'index' template



```
<h1>Welcome to The Flint & Flame!</h1>
```

We want this to be the default




Adding the Template Name

Each template needs a unique name.

index.html

```
<script type='text/x-handlebars' data-template-name='application'>  
  <div class='navbar'>...</div>  
  <div class='container'>...</div>  
  <footer class='container'>...</footer>  
</script>
```



The application template is shown
for every page by default.



Adding the Homepage Template

index.html

```
<script type='text/x-handlebars' data-template-name='application'>
  <div class='navbar'>...</div>
  <div class='container'>...</div>
  <footer class='container'>...</footer>
</script>

<script type='text/x-handlebars' data-template-name='index'>
  <h1>Welcome to The Flint & Flame!</h1>
</script>
```

'index' is what we'll call the homepage template name



Adding our About Template

index.html

```
<script type='text/x-handlebars' data-template-name='application'>
  <div class='navbar'>...</div>
  <div class='container'>...</div>
  <footer class='container'>...</footer>
</script>
```

```
<script type='text/x-handlebars' data-template-name='index'>
  <h1>Welcome to The Flint & Flame!</h1>
</script>
```

```
<script type='text/x-handlebars' data-template-name='about'>
  <h1>About The Fire Spirits</h1>
</script>
```





Only our application template is showing!

No welcome message!



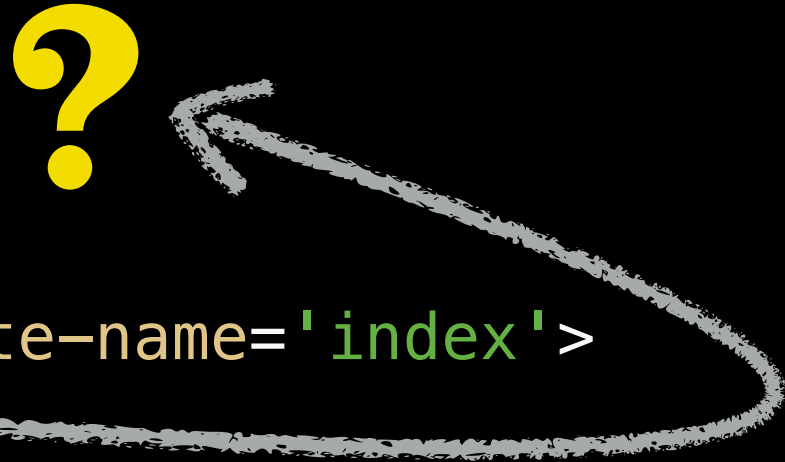
But where do they go?

We need a way of telling our templates where on the page to render.

```
<script type='text/x-handlebars' data-template-name='application'>
  <div class='navbar'>...</div>
  <div class='container'>...</div>
  <footer class='container'>...</footer>
</script>

<script type='text/x-handlebars' data-template-name='index'>
  <h1>Welcome to The Flint & Flame!</h1>
</script>

<script type='text/x-handlebars' data-template-name='about'>
  <h1>About The Fire Spirits</h1>
</script>
```



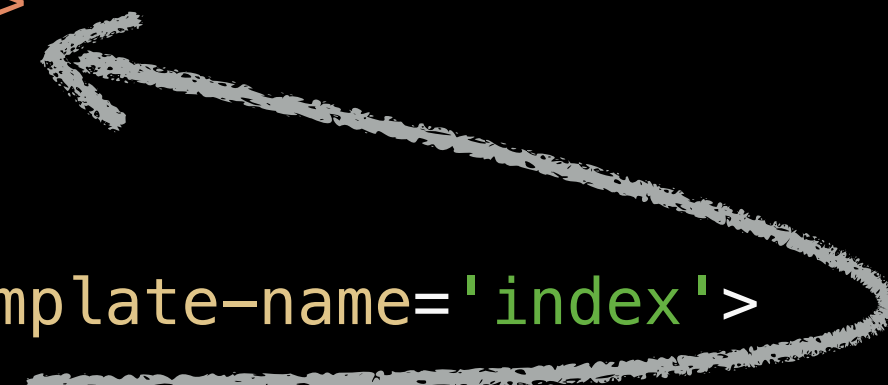
Outlet

Using the handlebars expression “outlet” we’re giving our code a hint as to where templates should be inserted.

```
<script type='text/x-handlebars' data-template-name='application'>
  <div class='navbar'>...</div>
  <div class='container'>{{outlet}}</div>
  <footer class='container'>...</footer>
</script>

<script type='text/x-handlebars' data-template-name='index'>
  <h1>Welcome to The Flint & Flame!</h1>
</script>

<script type='text/x-handlebars' data-template-name='about'>
  <h1>About The Fire Spirits</h1>
</script>
```



If our Ember code reaches an {{outlet}}, by default it will look to find a template named ‘index’ and render that in place of the outlet.





Welcome to The Flint & Flame!

© 2013 The Flint & Flame

Credits



Our index template is rendered within our application template

