

Warming Up With **ember.js**

Level 7 - Acting on Instinct

Handling Events in Views



Review

index.html

```
<script type='text/x-handlebars' data-template-name='product'>
  {{#if isNotReviewed}}
    <div class='new-review'>
      <h3>Review {{title}}</h3>
      {{#if review.text}}
        <p class='text-muted'>{{review.text}}</p>
      {{/if}}
      {{textarea valueBinding='review.text'}}
      <button {{action 'createReview'}} class='btn-primary'>Review</button>
    </div>
  {{/if}}
</script>
```



We have a form to create reviews

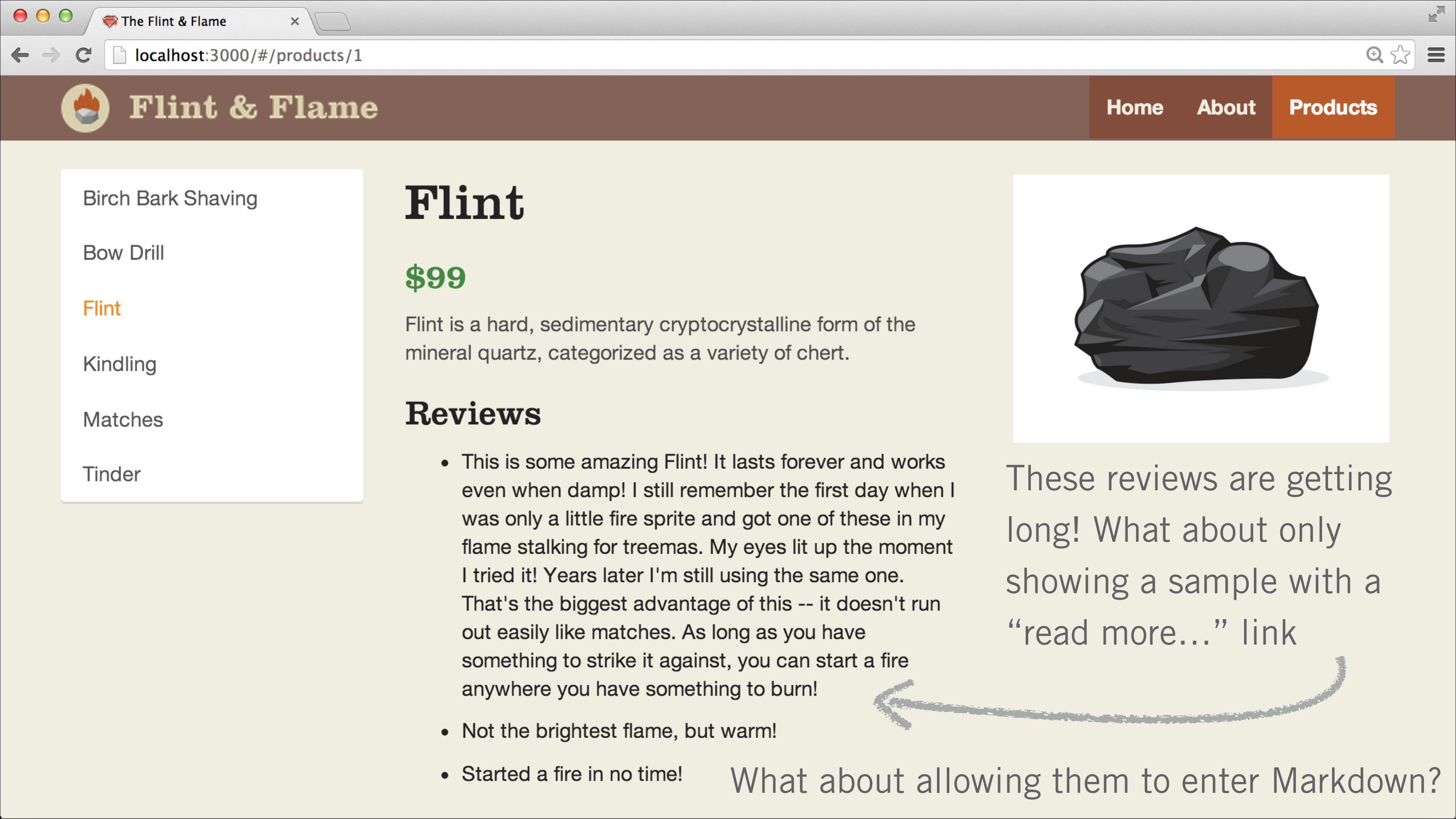


Review

app.js

```
App.ProductController = Ember.ObjectController.extend({
  isNotReviewed: Ember.computed.alias('review.isNew'),
  review: function() {
    return this.store.createRecord('review', {
      product: this.get('model')
    });
  }.property('model'),
  actions: {
    createReview: function() {
      var controller = this;
      this.get('review').set('reviewedAt', new Date());
      this.get('review').save().then(function(review) {
        controller.get('model.reviews').addObject(review);
      });
    }
  }
});
```





Birch Bark Shaving

Bow Drill

Flint

Kindling

Matches

Tinder

Flint

\$99

Flint is a hard, sedimentary cryptocrystalline form of the mineral quartz, categorized as a variety of chert.

Reviews

- This is some amazing Flint! It lasts forever and works even when damp! I still remember the first day when I was only a little fire sprite and got one of these in my flame stalking for treemas. My eyes lit up the moment I tried it! Years later I'm still using the same one. That's the biggest advantage of this -- it doesn't run out easily like matches. As long as you have something to strike it against, you can start a fire anywhere you have something to burn!
- Not the brightest flame, but warm!
- Started a fire in no time!




These reviews are getting long! What about only showing a sample with a “read more...” link

What about allowing them to enter Markdown?

Viewing Our Reviews

index.html

```
<script type='text/x-handlebars' data-template-name='reviews'>
  <h3>Reviews</h3>
  <ul>
    {{#each}}
      <li><p>{{text}}</p></li>
    {{else}}
      <li>
        <p class='text-muted'>
          <em>No reviews yet. Be the first to write one!</em>
        </p>
      </li>
    {{/each}}
  </ul>
</script>
```



We can abstract out a “review” into a View



Content Within a View

index.html

```
<script type='text/x-handlebars' data-template-name='reviews'>
  <h3>Reviews</h3>
  <ul>
    {{#each}}
      {{#view tag='li'}}
        <div class='content'>{{text}}</div>
      {{/view}}
    {{else}}
      ...
    {{/each}}
  </ul>
</script>
```

Creates a generic view with a `li` tag rather than a `div`



Using a View Class

index.html

```
<script type='text/x-handlebars' data-template-name='reviews'>
  <h3>Reviews</h3>
  <ul>
    {{#each}}
      {{#view 'App.ReviewView' tag='li'}}
        <div class='content'>{{text}}</div>
      {{/view}}
    {{else}}
      ...
    {{/each}}
  </ul>
</script>
```



Will use the `App.ReviewView` object as the view, let's create it!



Creating a View Class

app.js

```
App.ReviewView = Ember.View.extend({  
  ???  
});
```



TODO


1. We need a class 'isExpanded' that gets added to the 'li' when the text is expanded.
2. Need a click event, which toggle's this class.



Add the Proper Class

app.js

```
App.ReviewView = Ember.View.extend({  
  isExpanded: false,  
  classNameBindings: ['isExpanded']  
});
```



Will NOT have a class of `is-expanded` initially.

How can we add it when the review is clicked?



Listening for Click Events

When the review `li` element is clicked, change the expanded property

app.js

```
App.ReviewView = Ember.View.extend({
  isExpanded: false,
  classNameBindings: ['isExpanded'],
  click: function() {
    this.toggleProperty('isExpanded');
  }
});
```



Ember View Events

See the Ember API Docs for all events at <http://emberjs.com/>

click

mouseDown

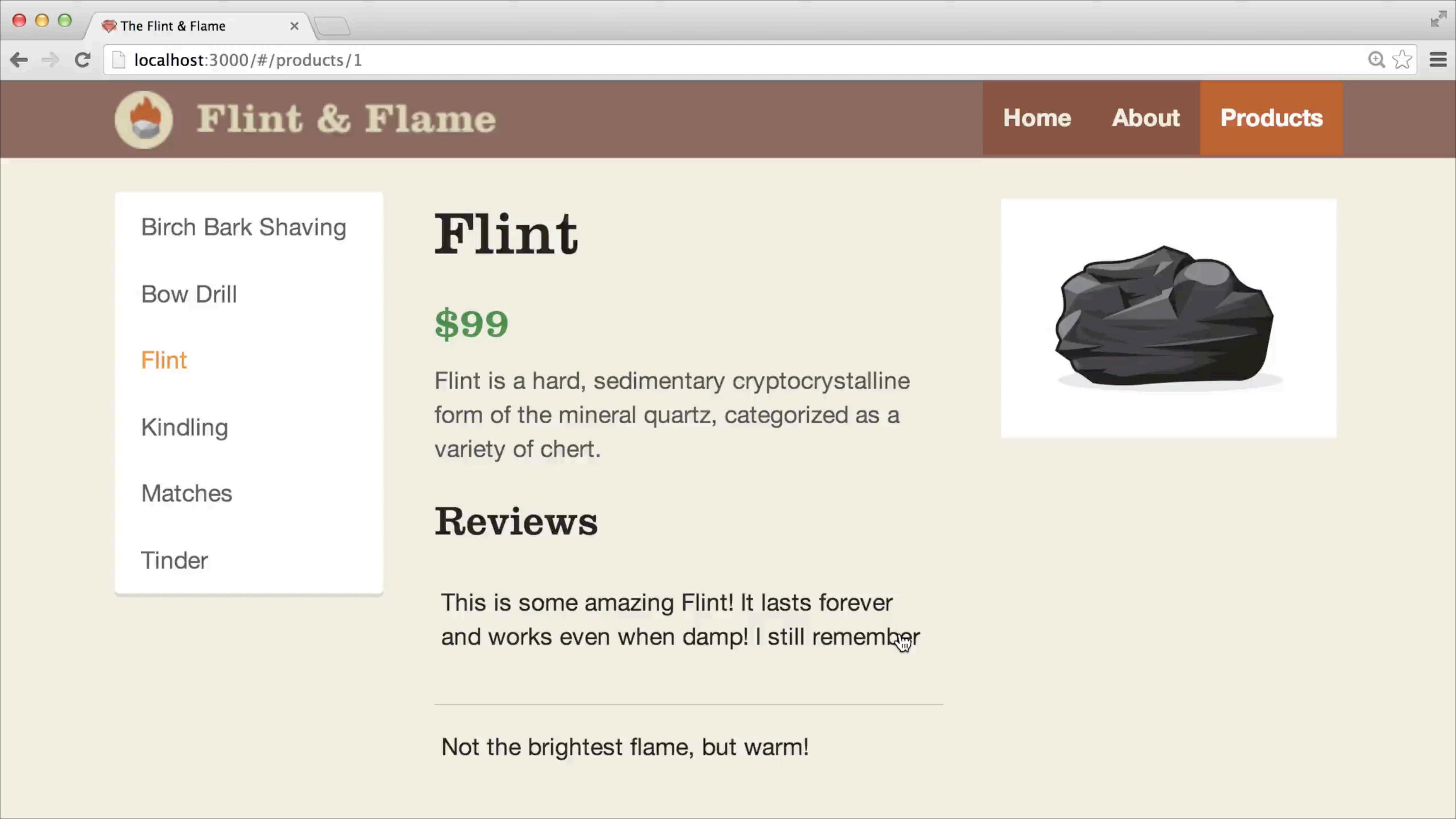
submit

change

keyDown

keyUp



[Birch Bark Shaving](#)[Bow Drill](#)[Flint](#)[Kindling](#)[Matches](#)[Tinder](#)

Flint

\$99

Flint is a hard, sedimentary cryptocrystalline form of the mineral quartz, categorized as a variety of chert.

Reviews

This is some amazing Flint! It lasts forever and works even when damp! I still remember

Not the brightest flame, but warm!



Our Problem

Need to add a class 'readMore' to the reviews that have more than 140 characters, so we can add "Read More".

Our Solution

1. Send our 'text.length' value into the View.
2. Create a function called 'readMore' which will check if string is > 140 characters.
3. Add the readMore className.
4. Add a "Read More" / "Read Less" link



Pass in the length

index.html

```
<script type='text/x-handlebars' data-template-name='reviews'>
  <h3>Reviews</h3>
  <ul>
    {{#each}}
      {{#view 'App.ReviewView' tag='li' length=text.length}}
        <div class='content'>{{text}}</div>
      {{/view}}
    {{else}}
      ...
    {{/each}}
  </ul>
</script>
```



Sets the length property on the ReviewView



Add the readMore Function

app.js

```
App.ReviewView = Ember.View.extend({
  isExpanded: false,
  classNameBindings: ['isExpanded'],
  click: function() {
    this.toggleProperty('isExpanded');
  },
  readMore: function() {
    return this.get('length') > 140;
  }.property('length')
});
```




Add the ClassName

app.js

Add a class if this is a long review

```
App.ReviewView = Ember.View.extend({
  isExpanded: false,
  classNameBindings: ['isExpanded', 'readMore'],
  click: function() {
    this.toggleProperty('isExpanded');
  },
  readMore: function() {
    return this.get('length') > 140;
  }.property('length')
});
```



index.html

```
{{#view 'App.ReviewView' tag='li' length=text.length}}
  <div class='content'>{{text}}</div>
  <span class='expand text-success'>
    Read {{#if view.isExpanded}}Less{{else}}More{{/if}}
  </span>
{{/view}}
```



Ember Computed

app.js

```
App.ReviewView = Ember.View.extend({  
  readMore: function() {  
    return this.get('length') > 140;  
  }.property('length')  
});
```



Functionally the same!

```
App.ReviewView = Ember.View.extend({  
  readMore: Ember.computed.gt('length', 140)  
});
```



Ember.computed

alias

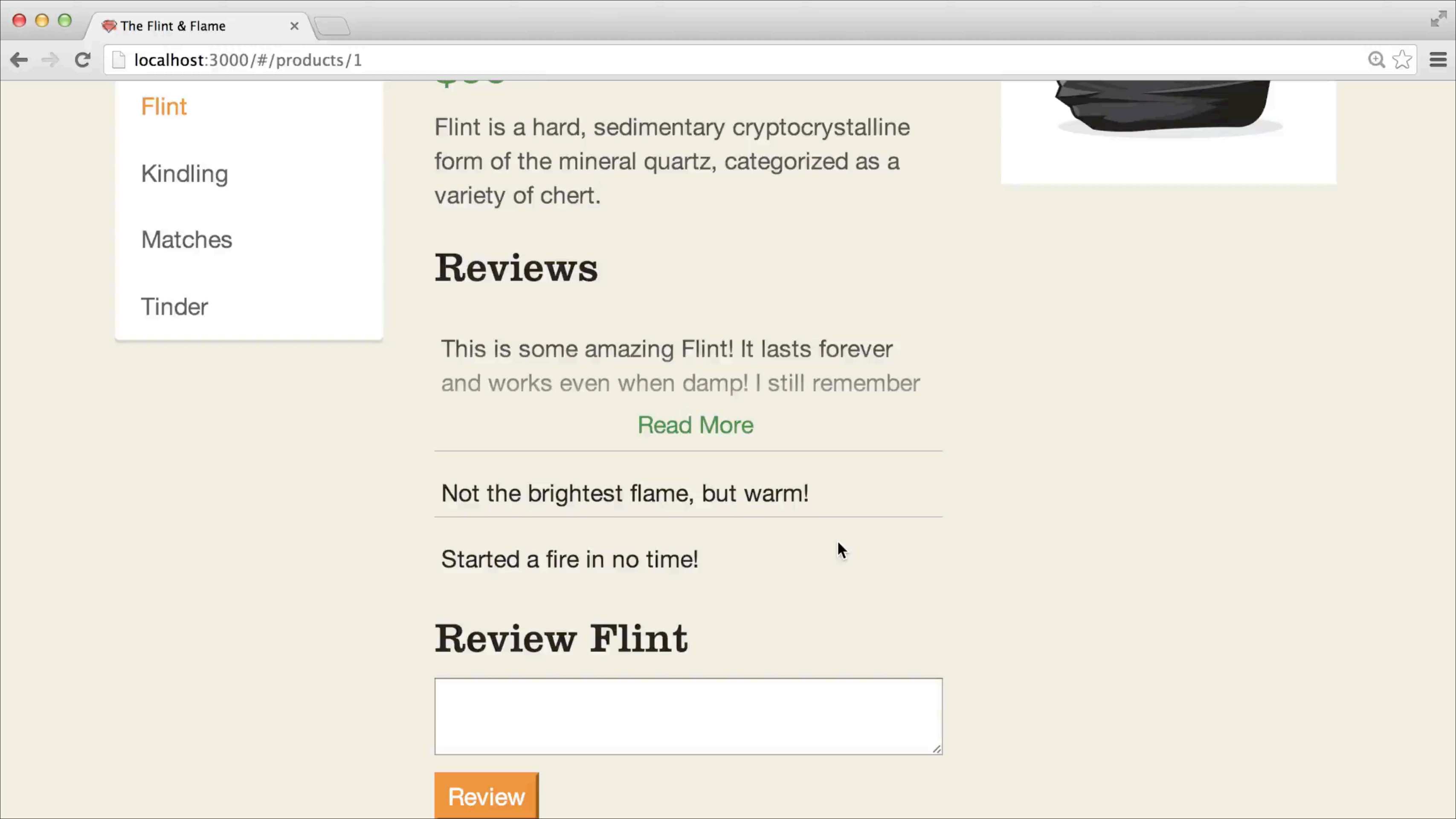
gt gte

lt lte

map any filter

max min





Flint

Kindling

Matches

Tinder

Flint is a hard, sedimentary cryptocrystalline form of the mineral quartz, categorized as a variety of chert.

Reviews

This is some amazing Flint! It lasts forever and works even when damp! I still remember

[Read More](#)

Not the brightest flame, but warm!

Started a fire in no time!

Review Flint

[Review](#)

Defining Handlebars Helpers

app.js


```
Ember.Handlebars.registerBoundHelper('markdown', function(text) {  
  return text;  
});
```



Defines a handlebars helper named “markdown”

index.html

```
<script type='text/x-handlebars' data-template-name='reviews'>  
  {{#view 'App.ReviewView' tag='li' length=text.length}}  
    <div class='content'>{{markdown text}}</div>  
  {{/view}}  
</script>
```



Use our handlebars helper, passing in the review



Using External Libraries

app.js

```
Ember.Handlebars.registerBoundHelper('markdown', function(text) {  
  return markdown.toHTML(text);  
});
```



We're going to use an external Markdown library which provides a `markdown.toHTML()` method

index.html


```
<script type='text/x-handlebars' data-template-name='reviews'>  
  {{#view 'App.ReviewView' tag='li' length=text.length}}  
    <div class='content'>{{markdown text}}</div>  
  {{/view}}  
</script>
```



Turning Off Escaping

app.js

```
Ember.Handlebars.registerBoundHelper('markdown', function(text) {  
  return new Handlebars.SafeString(markdown.toHTML(text));  
});
```

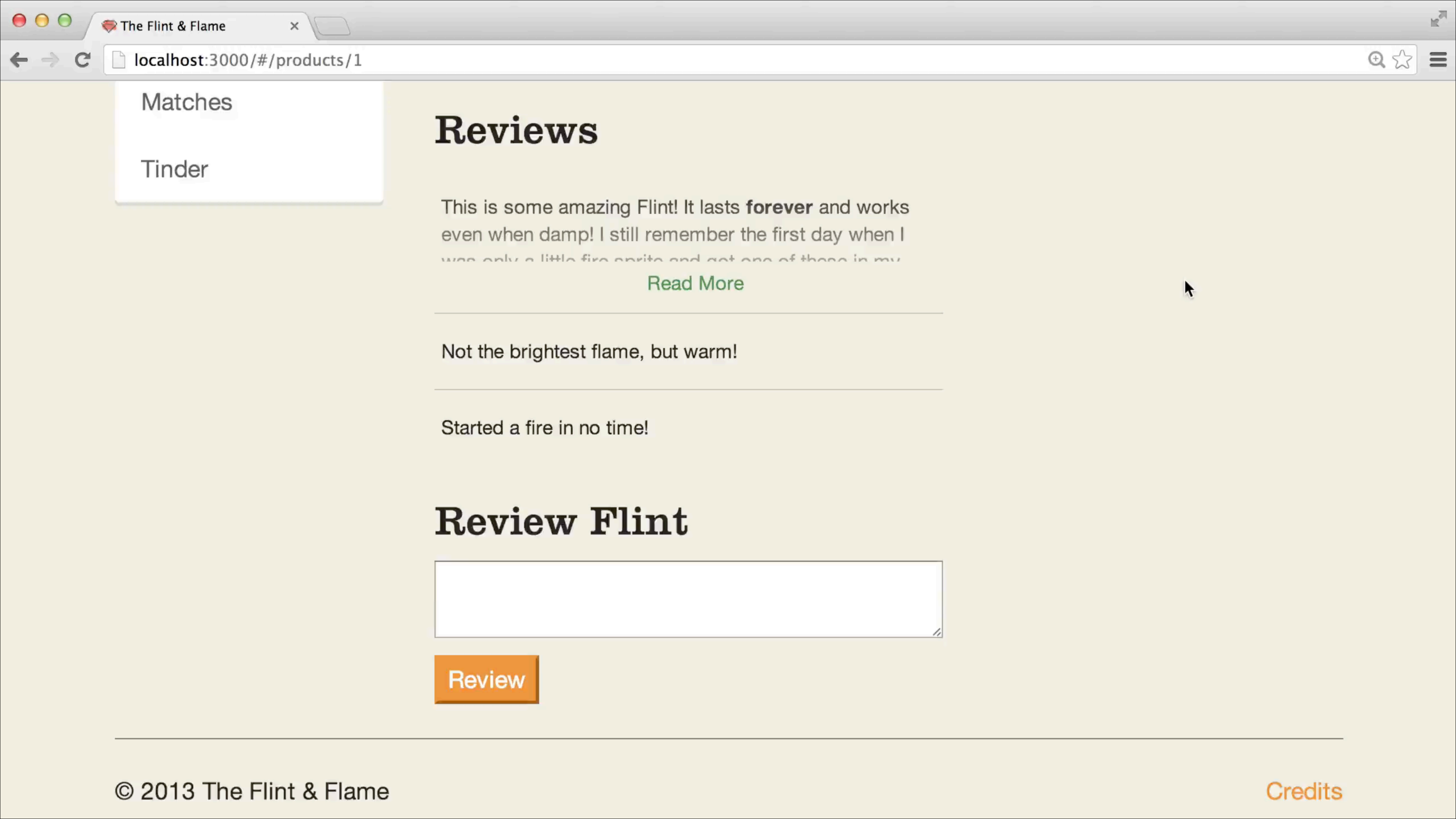


Handlebars will escape text by default, unless its a SafeString

index.html

```
<script type='text/x-handlebars' data-template-name='reviews'>  
  {{#view 'App.ReviewView' tag='li' length=text.length}}  
    <div class='content'>{{markdown text}}</div>  
  {{/view}}  
</script>
```





Matches

Tinder

Reviews

This is some amazing Flint! It lasts **forever** and works even when damp! I still remember the first day when I was only a little fire sprite and got one of these in my

[Read More](#)

Not the brightest flame, but warm!

Started a fire in no time!

Review Flint

Review

Challenges

- Create a handlebars helper to change the currency formatting using the accounting.js library (i think that's it?)

