

# Warming Up With **ember.js**

## Level 7 - Acting on Instinct

Controller Actions



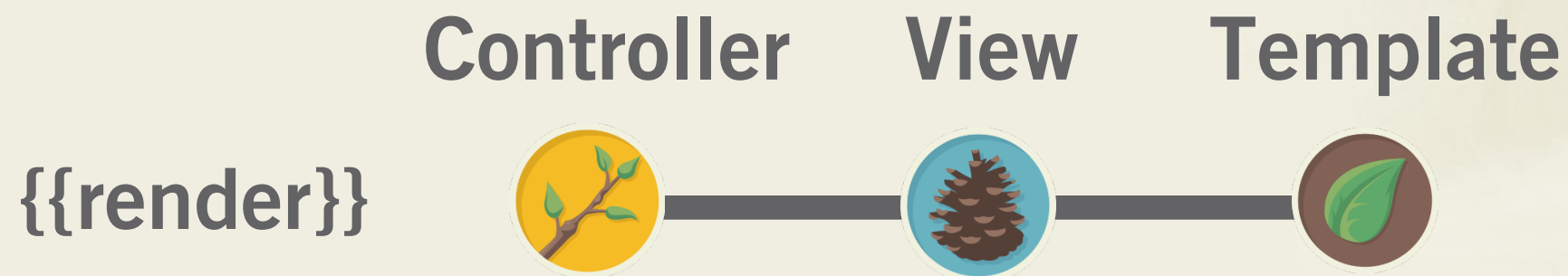
# Review

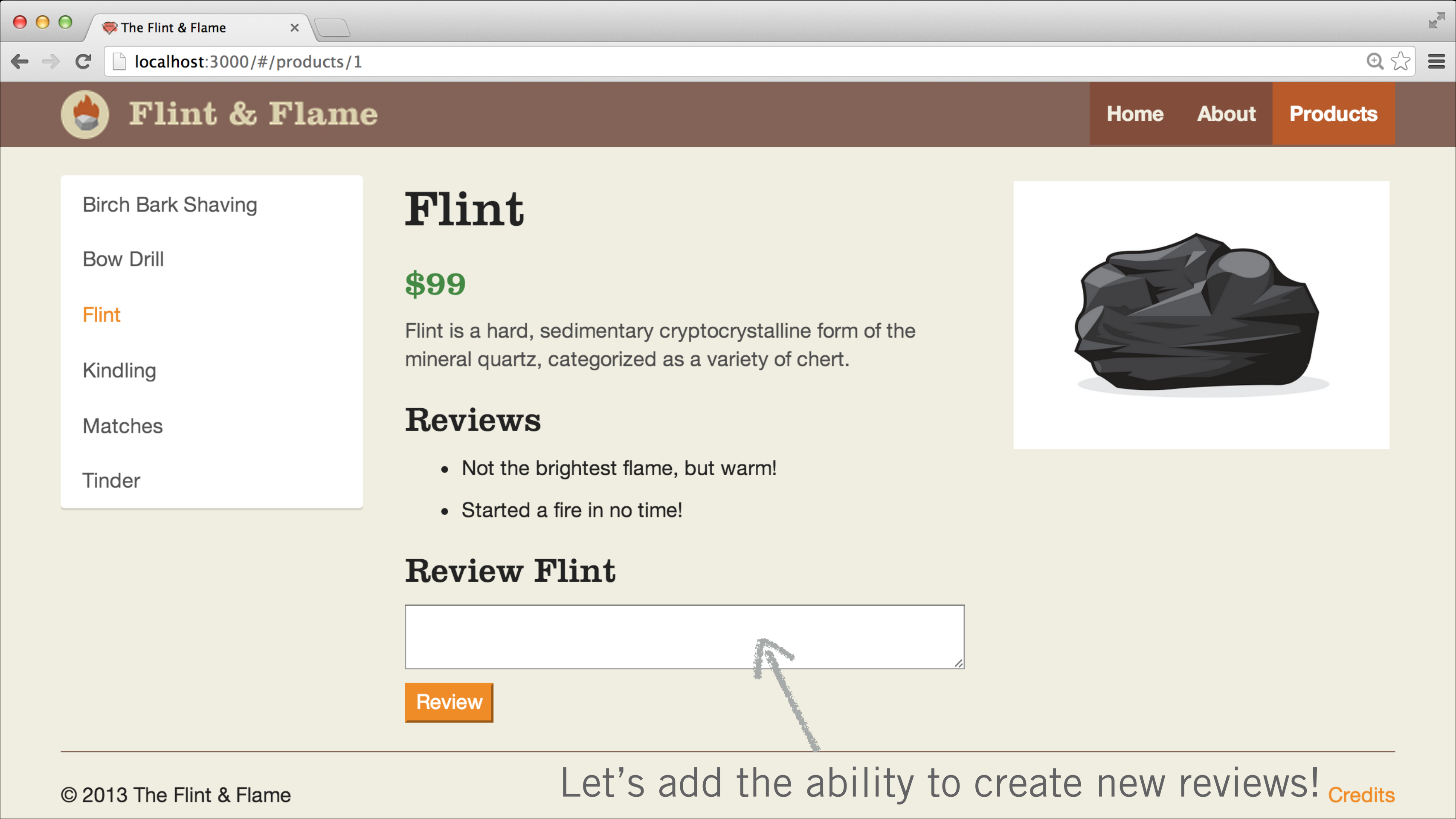
index.html

```
<script type='text/x-handlebars' data-template-name='product'>
  <h2>{{title}}</h2>
  {{render 'reviews' reviews}}
</script>
```

← The object passed in will become the model

```
<script type='text/x-handlebars' data-template-name='reviews'>
  <h3>Reviews</h3>
  <ul>
    {{#each}}...{{/each}}
  </ul>
</script>
```





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

- Not the brightest flame, but warm!
- Started a fire in no time!

### Review Flint

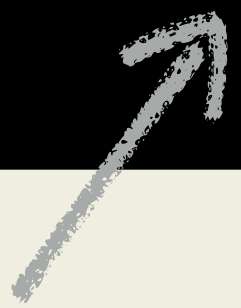
**Review**



# valueBinding for the textarea

index.html

```
<script type='text/x-handlebars' data-template-name='product'>
  ...
  <div class='new-review'>
    <h3>Review {{title}}</h3>
    {{textarea valueBinding='text'}}
  </div>
</script>
```



Binds the value of the text area to the text property on the controller

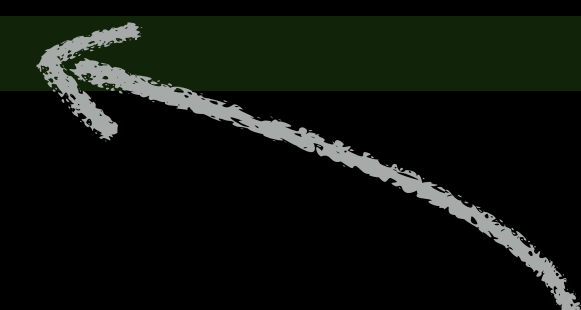




# Outputting a Property

index.html

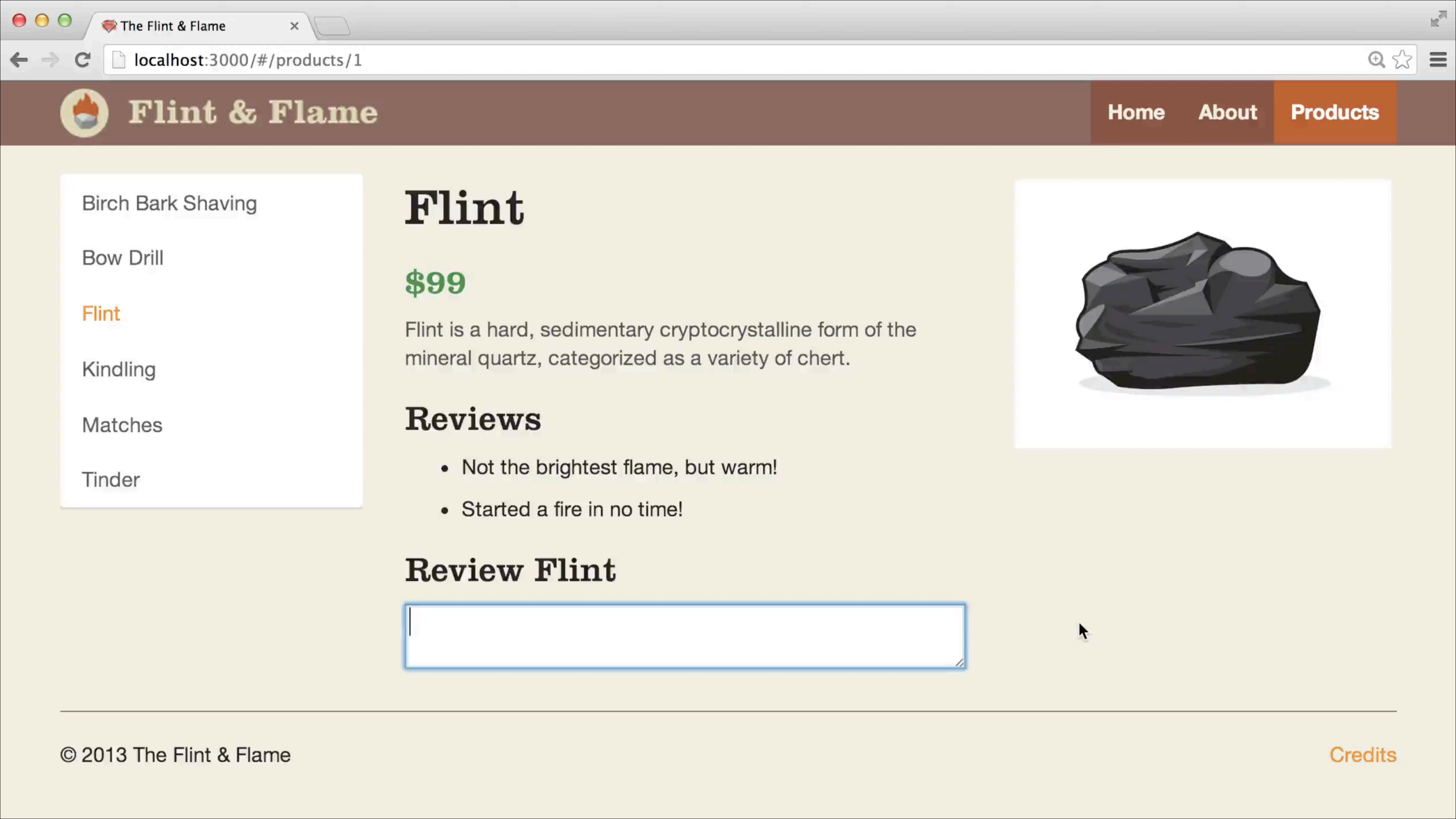
```
<script type='text/x-handlebars' data-template-name='product'>
  <div class='new-review'>
    <h3>Review {{title}}</h3>
    <p class='text-muted'>{{text}}</p>
    {{textarea valueBinding='text'}}
  </div>
</script>
```



Lets print this property out to the screen

*Just like any other property*





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

- Not the brightest flame, but warm!
- Started a fire in no time!

## Review Flint

# Next Steps

1. Only show “review” paragraph if {{text}} has a value.
2. Create the submit button.
3. Write the code which receives the review on submit.

index.html

```
<script type='text/x-handlebars' data-template-name='product'>
  <div class='new-review'>
    <h3>Review {{title}}</h3>
    <p class='text-muted'>{{text}}</p>
    {{textarea valueBinding='text'}}
  </div>
</script>
```



# First Two Steps

Only show the text if they've started typing

index.html

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

Call the createReview function  
on the controller when clicked.





# Controller Action

```
<script type='text/x-handlebars' data-template-name='product'>
  {{textarea valueBinding='text'}}
  <button {{action 'createReview'}} class='btn-primary'>Review</button>
</script>
```



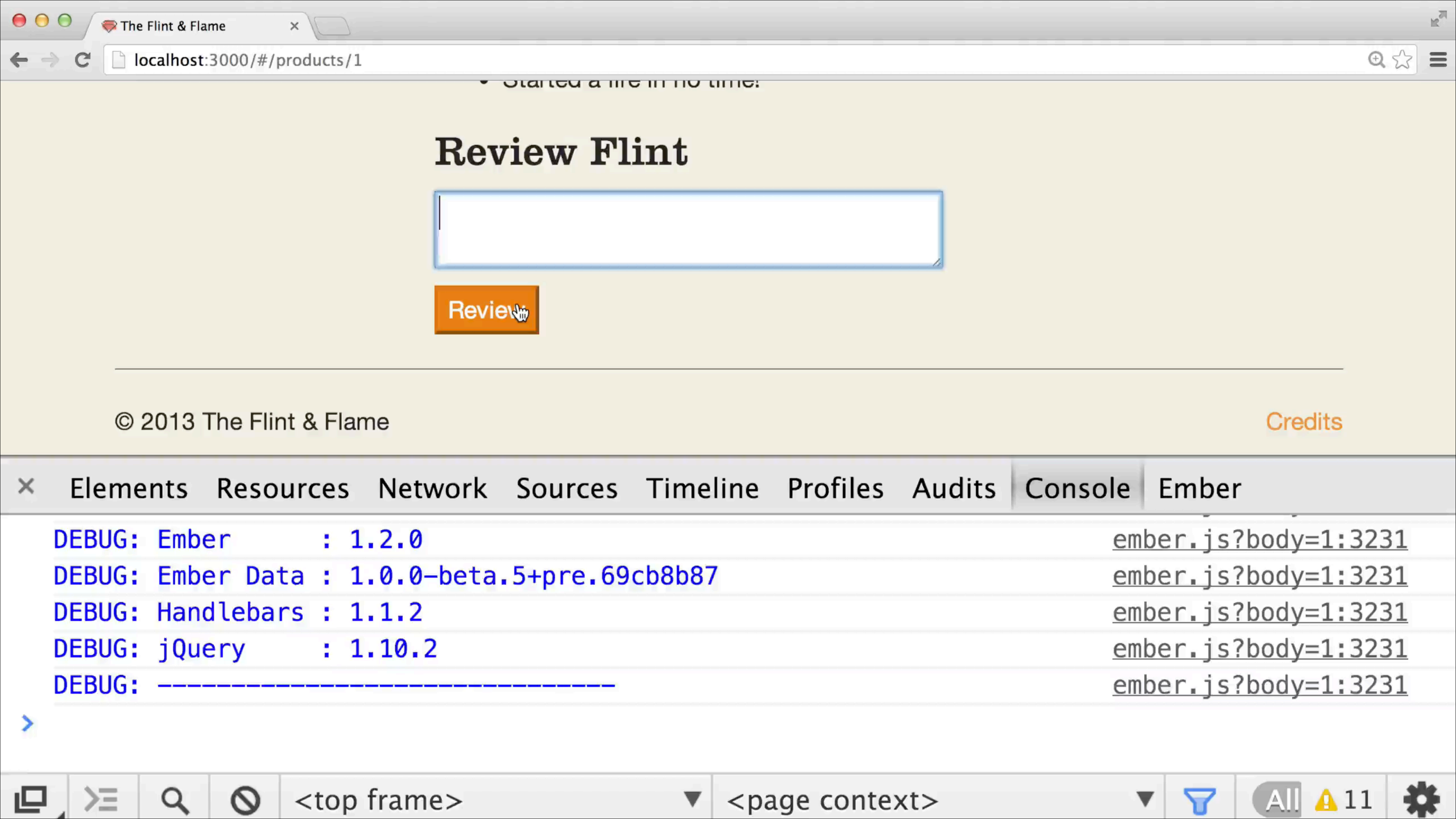
app.js

```
App.ProductController = Ember.ObjectController.extend({
  text: '',
  actions: {
    createReview: function() {
      console.log('createReview Called');
    }
  }
});
```

If left off, the property would be set on the model!

Clicking our button triggers the createReview function to be run





# Saving with Ember Data

app.js

```
App.ProductController = Ember.ObjectController.extend({  
  text: '',  
  actions: {  
    createReview: function() {  
      console.log('createReview Called');  
  
      // Step 1: Build a new Review object  
      // Step 2: Save the Review  
      // Step 3: Clear out the text variable  
    }  
  }  
});
```



We'll need to complete these steps in order to save this review

# Building a new Review Object

app.js

```
App.ProductController = Ember.ObjectController.extend({
  text: '',
  actions: {
    createReview: function() {
      // Step 1: Build a new Review object
      var review = this.store.createRecord('review', {
        text: this.get('text'),
        product: this.get('model'),
        reviewedAt: new Date()
      });

      // Step 2: Save the Review
      // Step 3: Clear out the text variable
    }
  }
});
```

←  
Sets up the new review object  
locally, but hasn't touched the server





# Saving a Review

app.js

```
App.ProductController = Ember.ObjectController.extend({
  text: '',
  actions: {
    createReview: function() {
      // Step 1: Build a new Review object
      var review = this.store.createRecord('review', {
        text: this.get('text'),
        product: this.get('model'),
        reviewedAt: new Date()
      });

      // Step 2: Save the Review
      review.save()

      // Step 3: Clear out the text variable
    }
  }
});
```

Will add it to our local fixtures. Would do a POST to a server if using the REST adapter



# The save() Promise

app.js

```
App.ProductController = Ember.ObjectController.extend({
  text: '',
  actions: {
    createReview: function() {
      var review = this.store.createRecord('review', {
        text: this.get('text'),
        product: this.get('model'),
        reviewedAt: new Date()
      });
      var controller = this;

      review.save().then(function(review) {
        // Will be called when the save call finishes
      });
    }
  }
});
```

Need to be able to reference the controller in the save callback



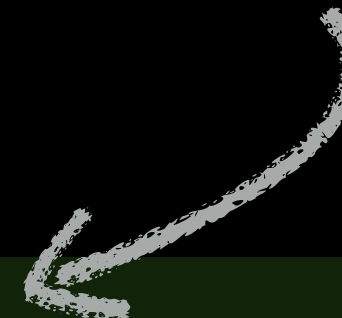
# Clear out the reviewText Variable

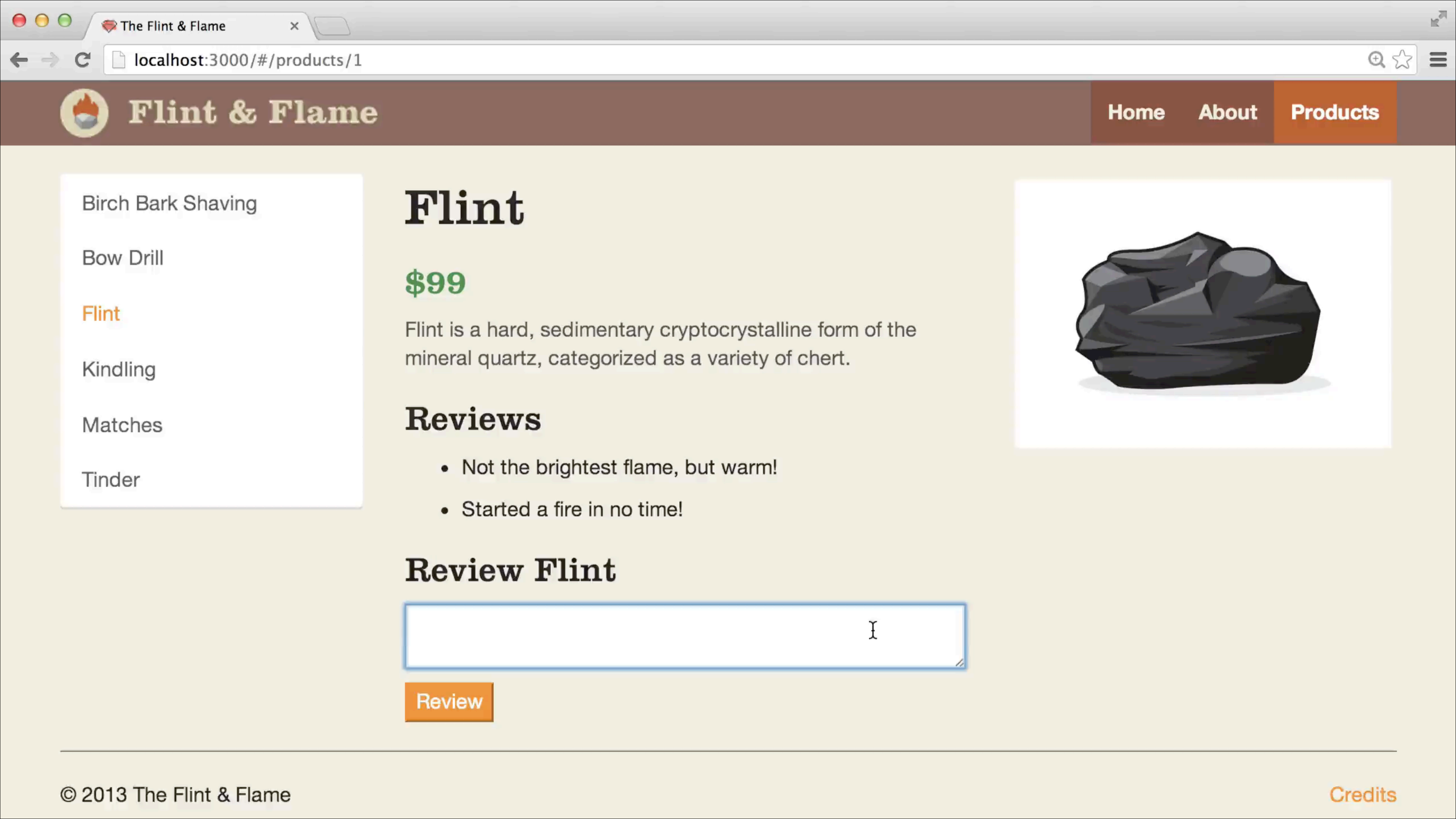
app.js

```
App.ProductController = Ember.ObjectController.extend({
  text: '',
  actions: {
    createReview: function() {
      var review = this.store.createRecord('review', {
        text: this.get('text'),
        product: this.get('model'),
        reviewedAt: new Date()
      });
      var controller = this;

      review.save().then(function(review) {
        controller.set('text', '');
        controller.get('model.reviews').addObject(review);
      });
    }
  }
});
```

Clear out reviewText and add the  
review to products.reviews





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

- Not the brightest flame, but warm!
- Started a fire in no time!

### Review Flint

**Review**

