JavaScript Functions

Object vs Function

```
var vacation = {
  init: function() {
    // init vacation
  }
};

$(document).ready(function() {
  vacation.init();
});
```

Limited to 1 confirmation on the page

Notice it's capitalized!

```
function Vacation(destination) {
  // init vacation to destination
}

$(document).ready(function() {
  var paris = new Vacation('Paris');
  var london = new Vacation('London');
});
```

Can have multiple vacations

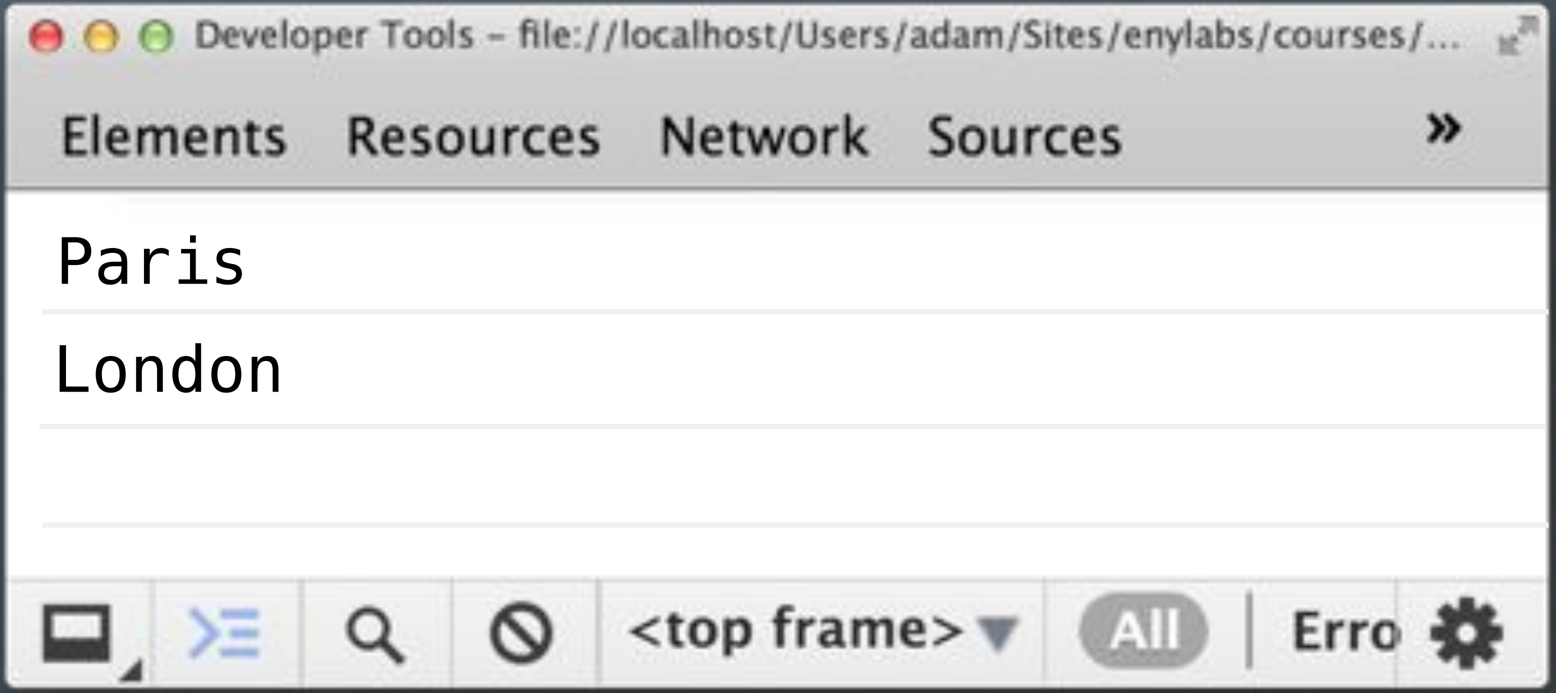
Functions as Objects Example

```
function Vacation(destination) {
  this.details = function() {
    console.log(destination);
  }
}

$(document).ready(function() {
  var paris = new Vacation('Paris');
  paris.details();

  var london = new Vacation('London');
  london.details();
});
```

paris and london both have their own scoped variables

Browser Logging

```
console.log(message);
```


Developer Tools – file://localhost/Users/adam/Sites/enylabs/courses/...
Elements
Resources
Network
Sources
»
<top frame>
All
Erro

```
Paris
London
```

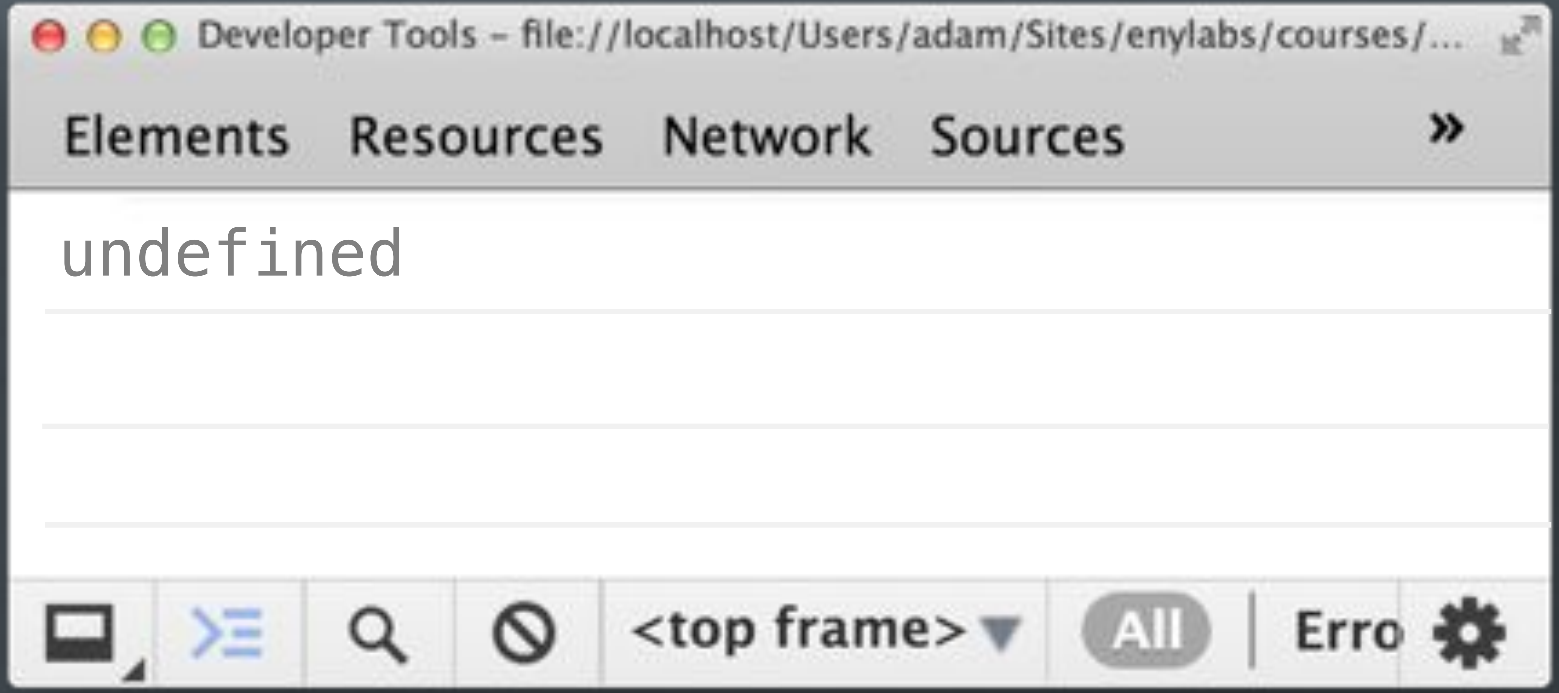
Functions as Objects Example

```
function Vacation(destination) {
  this.details = function() {
    console.log(destination);
  }
}

$(document).ready(function() {
  var paris = new Vacation('Paris');
  console.log(paris.destination);
});
```

Can't access locally scoped variables

Browser Logging

```
console.log(message);
```



Developer Tools – file://localhost/Users/adam/Sites/enylabs/courses/...
Elements
Resources
Network
Sources
»
<top frame>
All
Erro

```
undefined
```

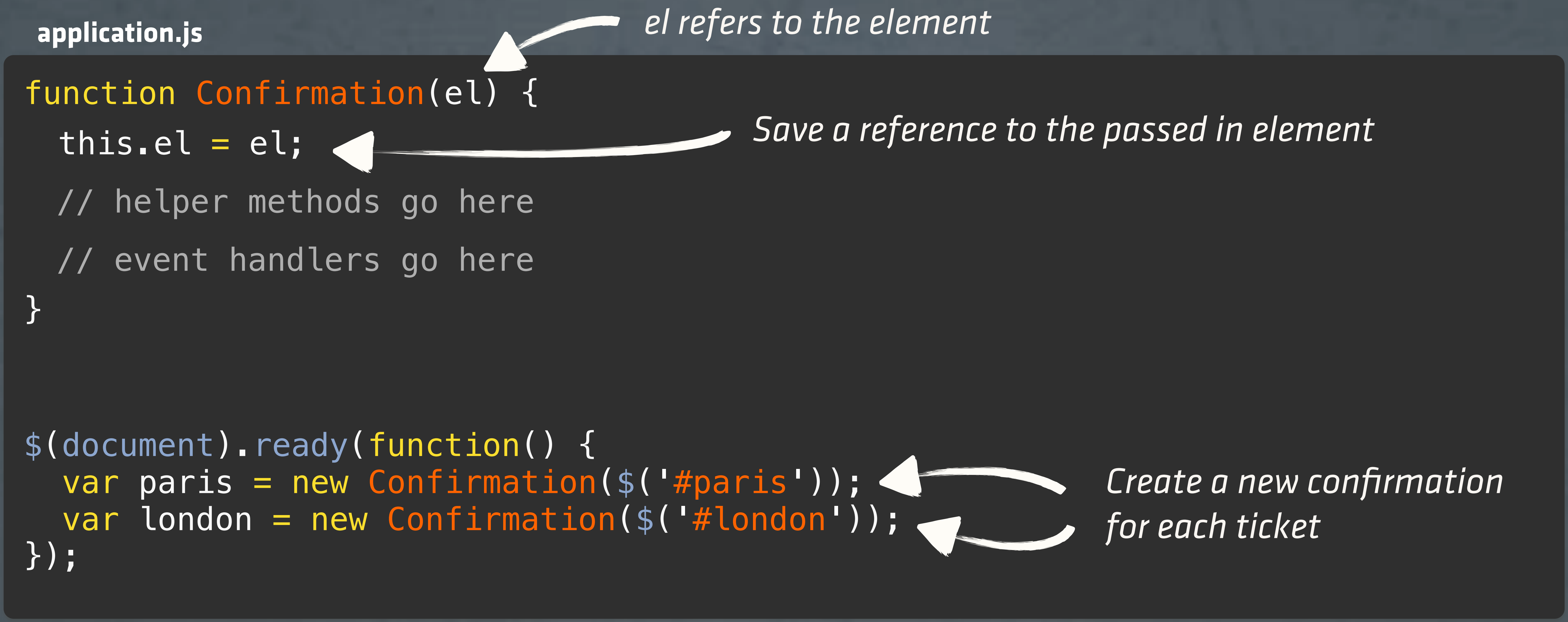
Lets Move this Object into a Reusable Function

application.js

```
var confirmation = {
  init: function() {
    $('.confirmation').on('click', 'button', this.loadConfirmation);
    $('.confirmation').on('click', '.view-boarding-pass', this.showBoardingPass);
  },
  loadConfirmation: function() { ... }
  showBoardingPass: function(event) { ... }
};

$(document).ready(function() {
  confirmation.init();
});
```

Refactoring to Functions

application.js



```
function Confirmation(el) {
  this.el = el;
  // helper methods go here
  // event handlers go here
}



$(document).ready(function() {
  var paris = new Confirmation($('#paris'));
  var london = new Confirmation($('#london'));
});
```

el refers to the element

Save a reference to the passed in element

Create a new confirmation for each ticket

Adding the methods and event listeners

application.js

```
function Confirmation(el) {
  this.el = el;

  this.loadConfirmation = function() { ... }
  this.showBoardingPass = function(event) { ... }

  this.el.on('click', 'button', this.loadConfirmation);
  this.el.on('click', '.view-boarding-pass', this.showBoardingPass);
}

$(document).ready(function() {
  var paris = new Confirmation($('#paris'));
  var london = new Confirmation($('#london'));
});
```

Fixing the target element

application.js

```
function Confirmation(el) {
  this.el = el;

  this.loadConfirmation = function() {
    $.ajax('confirmation.html', {
      timeout: 3000,
      success: function(response) {
        $('.ticket').html(response).slideDown();
      }
    });
  }

  this.el.on('click', 'button', this.loadConfirmation);
  this.el.on('click', '.view-boarding-pass', this.showBoardingPass);
}
```

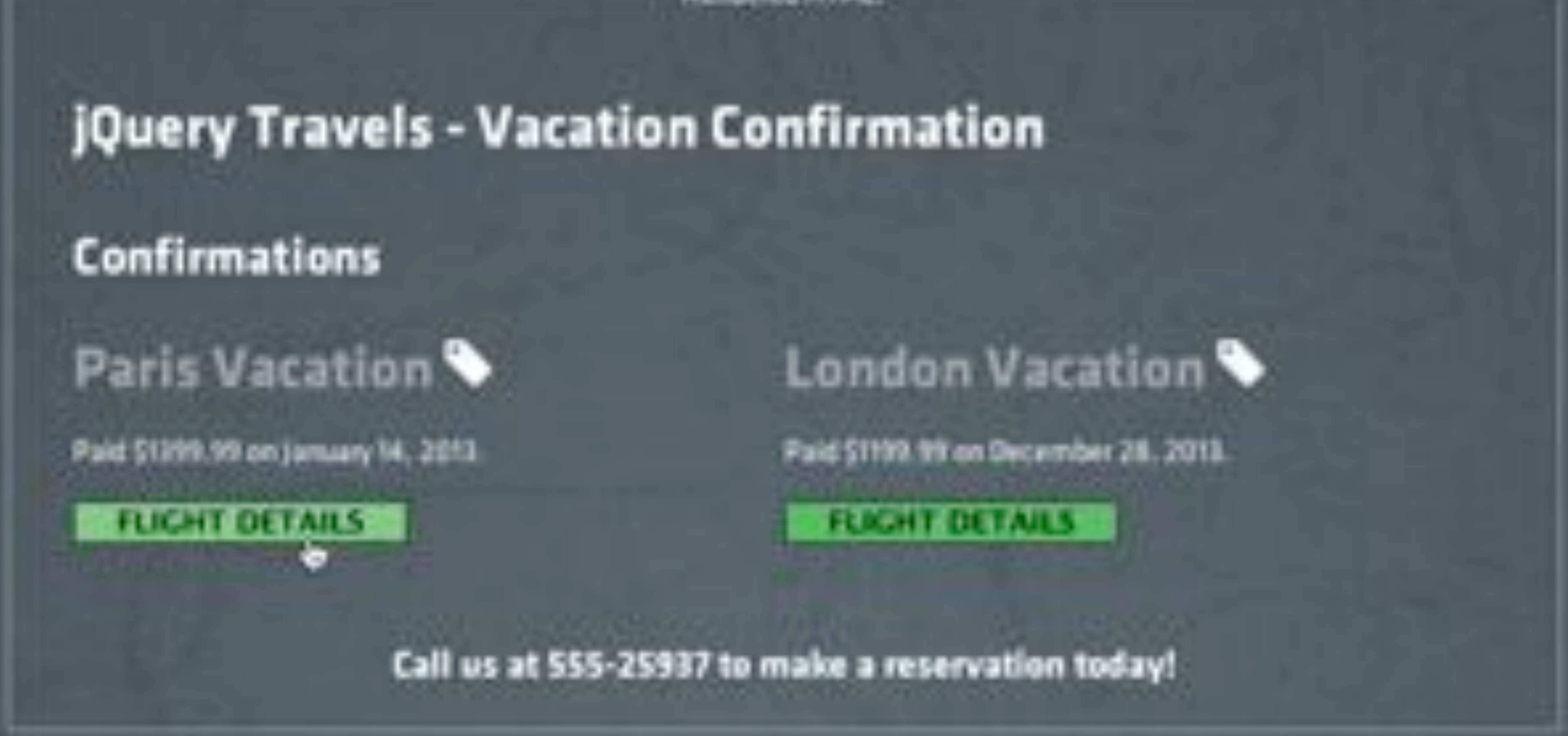
Will target all tickets, not just the one that was clicked

Fixing the target element

application.js

```
function Confirmation(el) {
  this.el = el;
  this.ticket = this.el.find('.ticket');

  this.loadConfirmation = function() {
    $.ajax('confirmation.html', {
      timeout: 3000,
      success: function(response) {
        this.ticket.html(response).slideDown();
      }
    });
  }

  this.el.on('click', 'button', this.loadConfirmation);
  this.el.on('click', '.view-boarding-pass', this.showBoardingPass);
}
```

Rendered HTML:



jQuery Travels - Vacation Confirmation

Confirmations

Paris Vacation 🏷

Paid $1399.99 on January 14, 2013.

London Vacation 🏷

Paid $1199.99 on December 28, 2013.

Call us at 555-25937 to make a reservation today!

We’re referencing the wrong ‘this’

application.js

```
function Confirmation(el) {
  this.el = el;
  this.ticket = this.el.find('.ticket');

  this.loadConfirmation = function() {
    $.ajax('confirmation.html', {
      timeout: 3000,
      success: function(response) {
        this.ticket.html(response).slideDown();
      }
    });
  }
}
```

This refers to ajax settings and not our function’s el

Sometimes jQuery changes the value of ‘this’

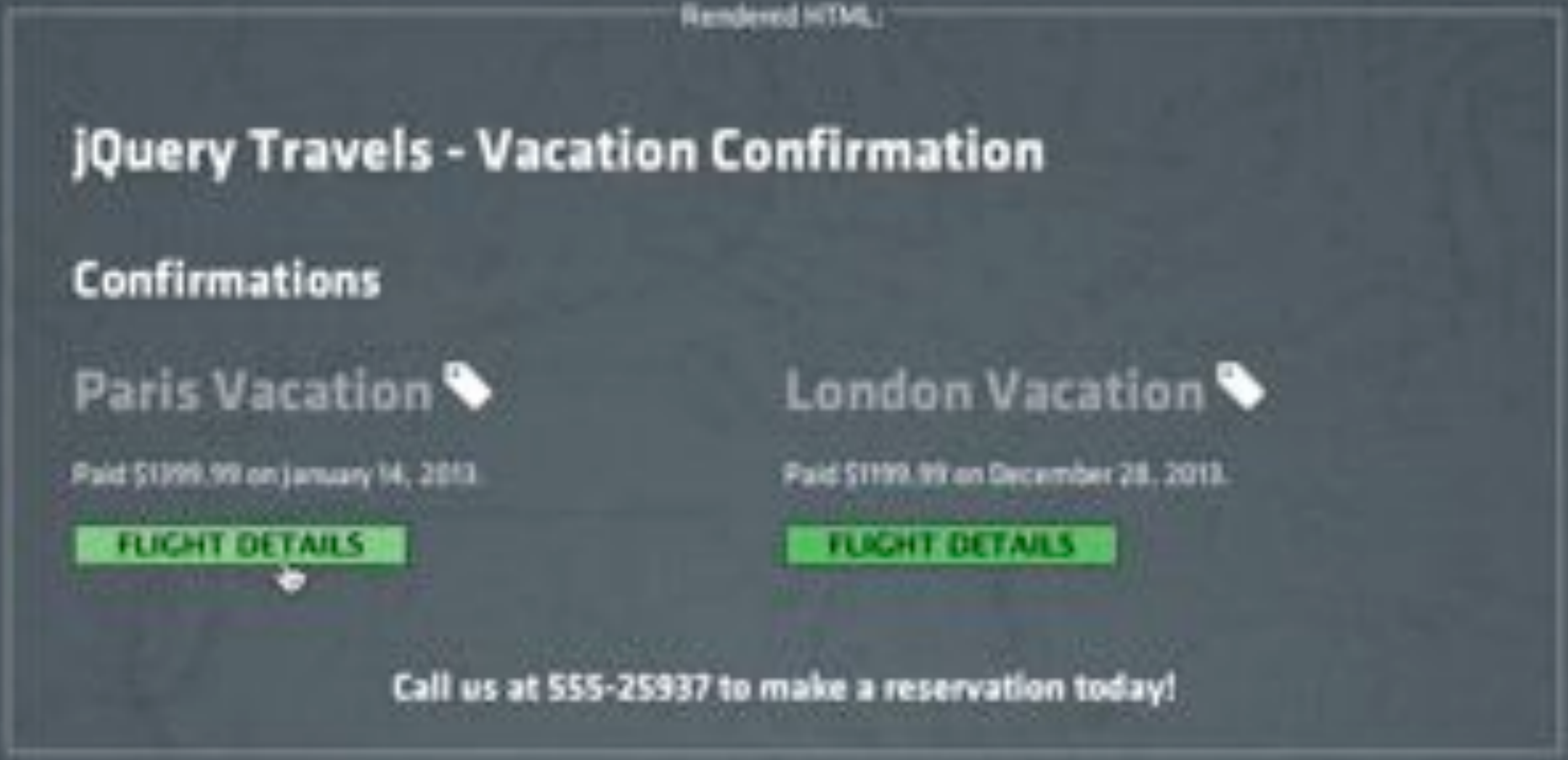
Inside AJAX callbacks, this is set to the AJAX settings.

Setting the proper ‘this’

application.js

```
function Confirmation(el) {
  this.el = el;
  this.ticket = this.el.find('.ticket');
  var confirmation = this;

  this.loadConfirmation = function() {
    $.ajax('confirmation.html', {
      timeout: 3000,
      context: confirmation,
      success: function(response) {
        this.ticket.html(response).slideDown();
      }
    });
  }
}
```

Allows us to set the value of “this” inside our callbacks.



Rendered HTML:

jQuery Travels - Vacation Confirmation

Confirmations

Paris Vacation 🏷

Paid $1399.99 on January 14, 2013.

London Vacation 🏷

Paid $1199.99 on December 28, 2013.

Call us at 555-25937 to make a reservation today!

Refactoring hardcoded DOM Elements

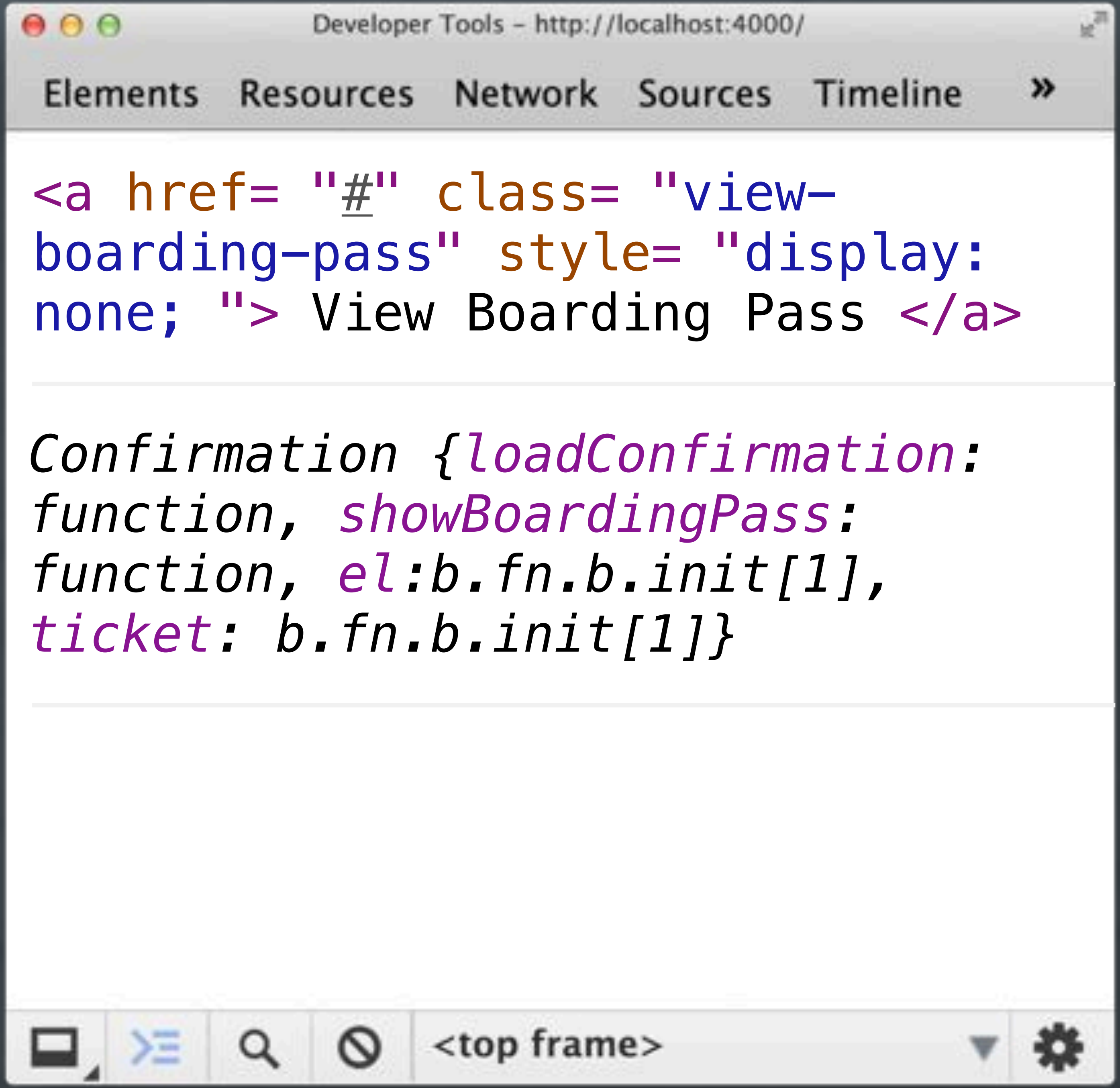
application.js

```
this.showBoardingPass = function(event) {
  event.preventDefault();
  $('.view-boarding-pass').hide();
  $('.boarding-pass').show();

  console.log(this);
  console.log(confirmation);
}
```

Run when the link is clicked

In the context of an event callback 'this' is set to the dom object

References the confirmation object in the parent scope



```
<a href= "#" class= "view-boarding-pass" style= "display: none; "> View Boarding Pass </a>

Confirmation {loadConfirmation: function, showBoardingPass: function, el:b.fn.b.init[1], ticket: b.fn.b.init[1]}
```

Refactoring to Functions

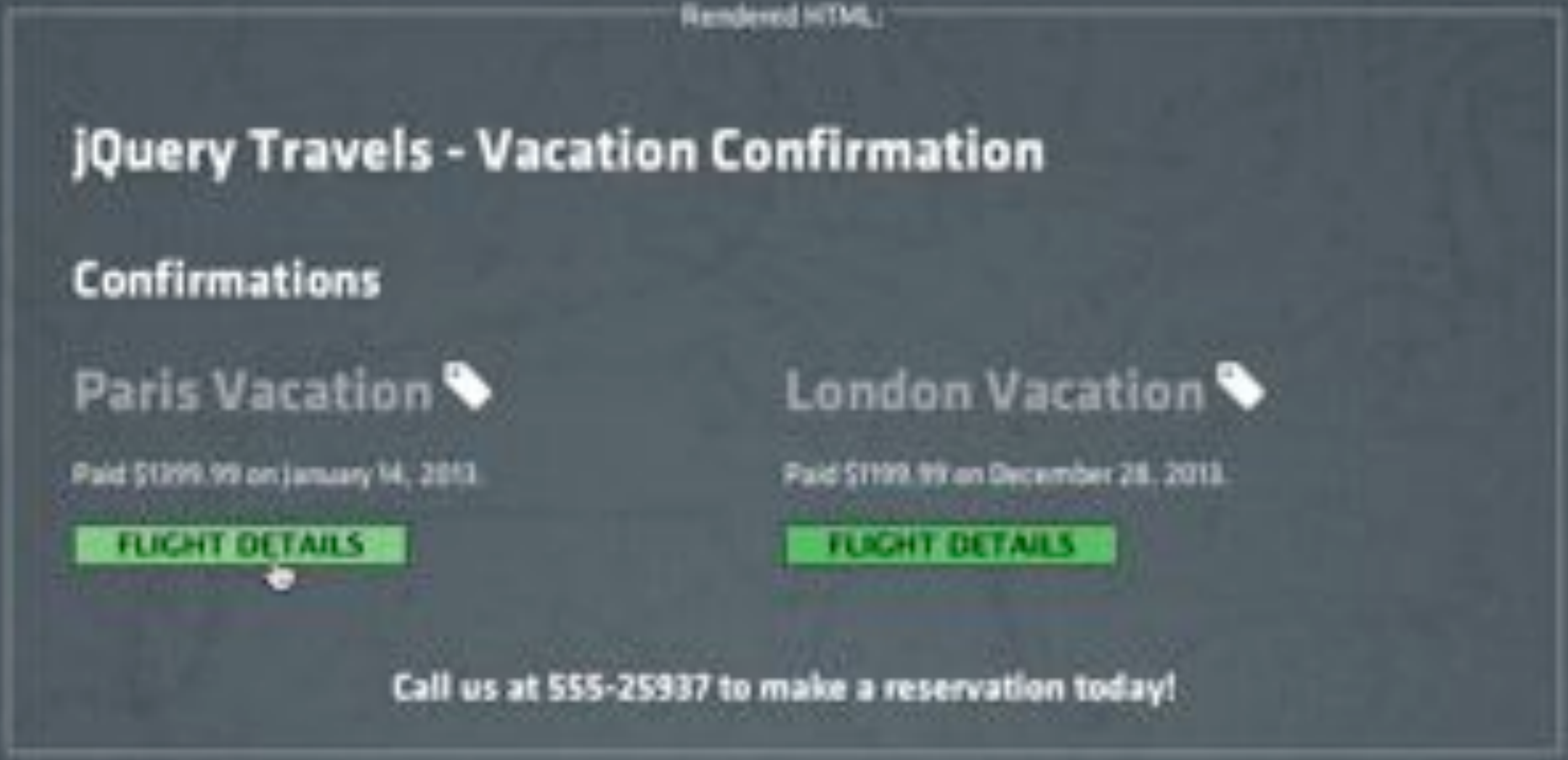
Remember, this will be the link that was clicked

application.js

```
this.showBoardingPass = function(event) {
  event.preventDefault();
  $(this).hide();
  confirmation.el.find('.boarding-pass').show();
}
```

Able to use variables from the parent scope

Rendered HTML:



jQuery Travels - Vacation Confirmation

Confirmations

Paris Vacation

Paid $1399.99 on January 14, 2013.

London Vacation

Paid $1199.99 on December 28, 2013.

Call us at 555-25937 to make a reservation today!