

HTML5 and CSS3 for Mobile Applications

Prof. Paul Krause, University of Surrey
Laying the Foundations for a Makeover

Objectives for today

- The new HTML5 elements
- How to make IE < 9 recognise them
- How to reset all browsers to the same foundation
- Starting off the assembly of your very own Boilerplate

```
<!DOCTYPE html>
<html>
<head>
  <title><%= title %></title>
  <%= stylesheet_link_tag "application", :media => "all" %>
  <%= javascript_include_tag "application" %>
  <%= csrf_meta_tags %>
</head>
<body>

  <%= render 'layouts/header' %>

  <p class="notice"><%= notice %></p>
  <p class="alert"><%= alert %></p>

  <%= render 'layouts/navigation' %>

  <section class="sidebar" id="right">
    <% if user_signed_in? %>
      Signed in as <%= current_user.email %>:
      <%= link_to "Sign out", destroy_user_session_path, :method => :delete %>
    <% else %>
      <%= link_to "Sign up", new_user_registration_path %> or <%= link_to "sign in",
        new_user_session_path %>
    <% end %>
  </section>

  <%= yield %>

  <%= render 'layouts/footer' %>

</body>
</html>
```

Ashtead Allotments Association

Healthy Eating and Exercise preserve Life's Balance

- [Home](#)
- [Events](#)
- [Discussion](#)
- [Shop](#)

[Sign up](#) or [sign in](#)

AaaCore#index

Find me in app/views/aaa_core/index.html.erb

© 2012 Ashtead Allotment's Association

HTML5 - away with 'divitis'

- abbr
- article
- aside
- audio
- canvas
- datalist
- details
- figcaption
- figure
- footer
- header
- hgroup
- mark
- meter
- nav
- output
- progress
- section
- summary
- time
- video

ARIA roles

- Accessible Rich Internet Applications
 - part of the Web Accessibility Initiative (WAI)
- Designed to improve the accessibility and interoperability of web content and applications
- ARIA roles can be added to tags to provide more meaning
 - role = “{value}”
- See: <http://www.w3.org/TR/wai-aria/>

Setting Up a Boilerplate

Boilerplate = Standardised View Code

- Rails has started us off with “views/layouts/application.rb”
- But it is worth adding to this, and reusing it/refining it across your application portfolio
- Helps ensure consistency across browsers and devices

Character Encoding

- Rule No. 1
 - “You should always specify a character encoding on every HTML page you serve”
 - <http://blog.whatwg.org/the-road-to-html-5-character-encoding>
- There are many choices but UTF-8 makes internationalisation easier to deal with and is recommended in the HTML5 spec

Character Encoding

- Rule No. 1
 - “You should always specify a character encoding on every HTML page you serve”
 - <http://blog.whatwg.org/the-road-to-html-5-character-encoding>
- There are many choices but UTF-8 makes internationalisation easier to deal with and is recommended in the HTML5 spec

```
<head>
  <meta charset="utf-8">
  <title><%= title %></title>
  <%= stylesheet_link_tag      "application", :media => "all" %>
  <%= javascript_include_tag  "application" %>
  <%= csrf_meta_tags %>
</head>
```

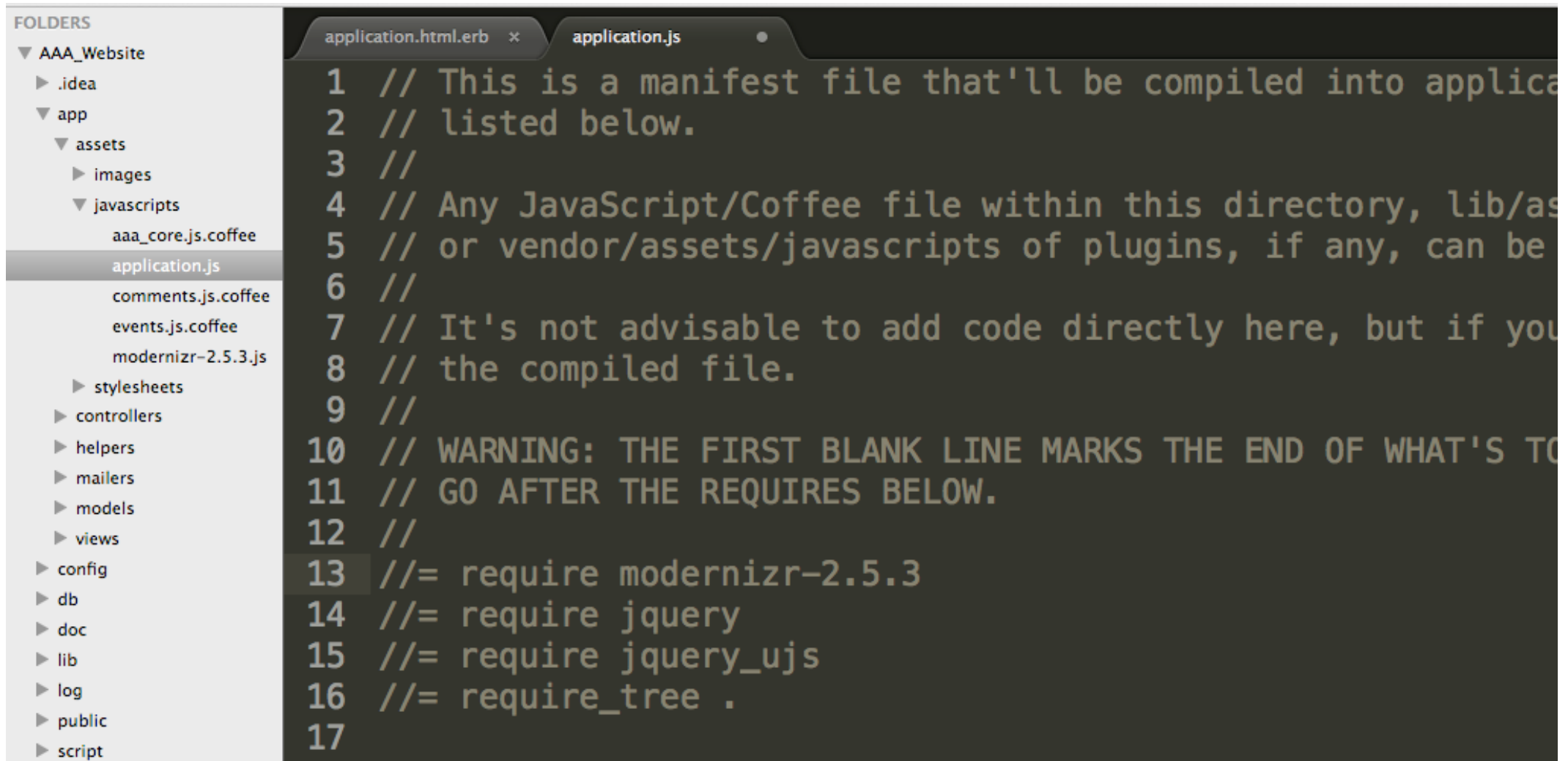
Joel on Software

- “The Absolute Minimum” to read
 - <http://www.joelonsoftware.com/articles/Unicode.html>

Bit of a Snag :(

- The new HTML5 elements are not supported by IE < 9
- Not too difficult to fix, by using JavaScript to create the new elements
 - The “HTML5 shiv”
- We will use Modernizr to do this
 - and a shed load of other things

Updating: app/assets/javascripts/application.js



The image shows a code editor interface. On the left is a 'FOLDERS' sidebar showing a project structure. The 'app/assets/javascripts' folder is expanded, and 'application.js' is selected. The main editor area shows the content of 'application.js' with line numbers 1 through 17. The code is a manifest file for RequireJS, containing comments and require statements for 'modernizr-2.5.3', 'jquery', 'jquery_ujs', and 'require_tree'.

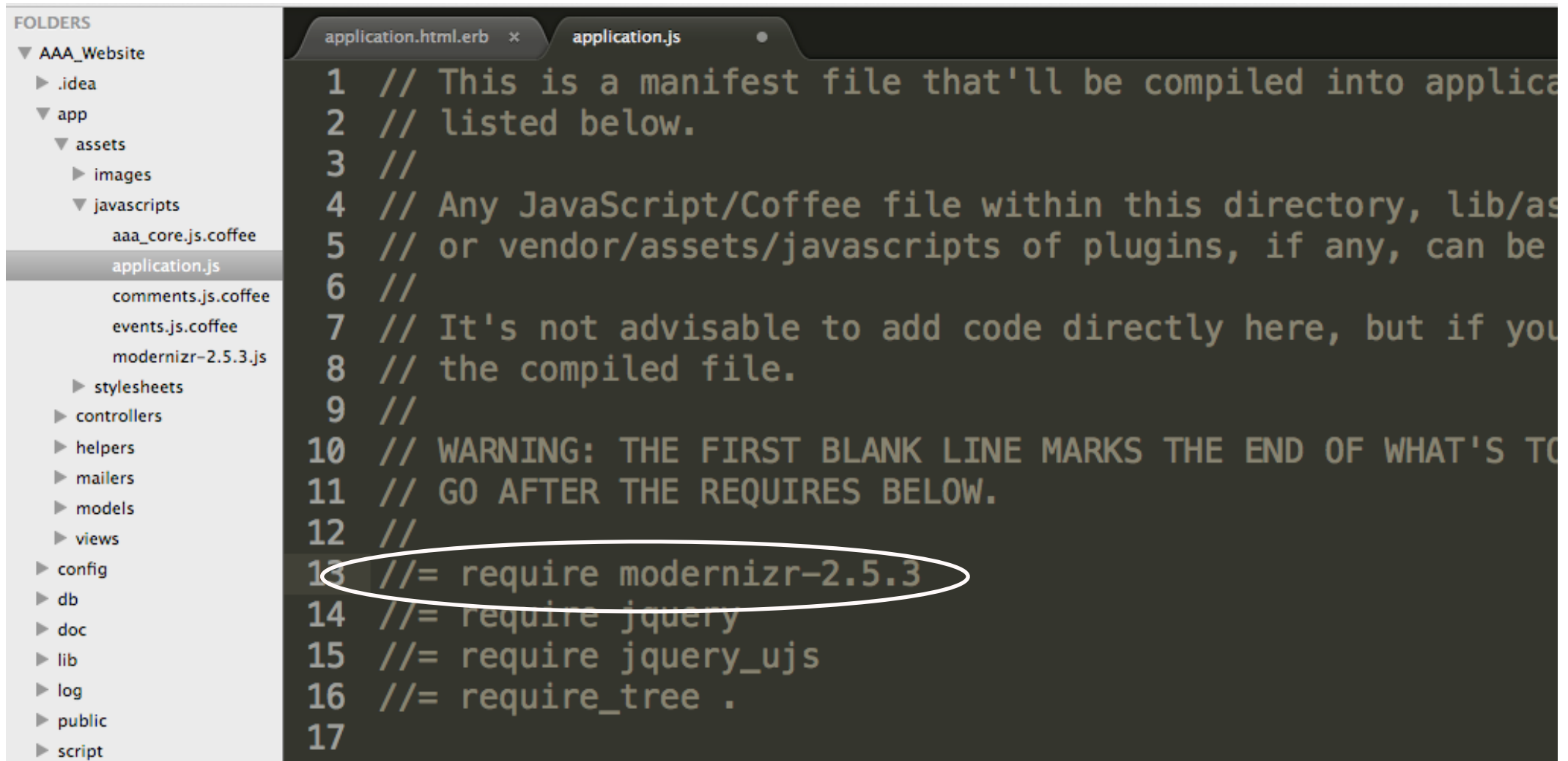
FOLDERS

- ▼ AAA_Website
 - ▶ .idea
 - ▼ app
 - ▼ assets
 - ▶ images
 - ▼ javascripts
 - aaa_core.js.coffee
 - application.js**
 - comments.js.coffee
 - events.js.coffee
 - modernizr-2.5.3.js
 - ▶ stylesheets
 - ▶ controllers
 - ▶ helpers
 - ▶ mailers
 - ▶ models
 - ▶ views
 - ▶ config
 - ▶ db
 - ▶ doc
 - ▶ lib
 - ▶ log
 - ▶ public
 - ▶ script

application.html.erb × **application.js** ●

```
1 // This is a manifest file that'll be compiled into applica
2 // listed below.
3 //
4 // Any JavaScript/Coffee file within this directory, lib/as
5 // or vendor/assets/javascripts of plugins, if any, can be
6 //
7 // It's not advisable to add code directly here, but if you
8 // the compiled file.
9 //
10 // WARNING: THE FIRST BLANK LINE MARKS THE END OF WHAT'S TO
11 // GO AFTER THE REQUIRES BELOW.
12 //
13 //= require modernizr-2.5.3
14 //= require jquery
15 //= require jquery_ujs
16 //= require_tree .
17
```

Updating: app/assets/javascripts/application.js



```
FOLDERS
▼ AAA_Website
  ► .idea
  ▼ app
    ▼ assets
      ► images
      ▼ javascripts
        aaa_core.js.coffee
        application.js
        comments.js.coffee
        events.js.coffee
        modernizr-2.5.3.js
      ► stylesheets
    ► controllers
    ► helpers
    ► mailers
    ► models
    ► views
  ► config
  ► db
  ► doc
  ► lib
  ► log
  ► public
  ► script

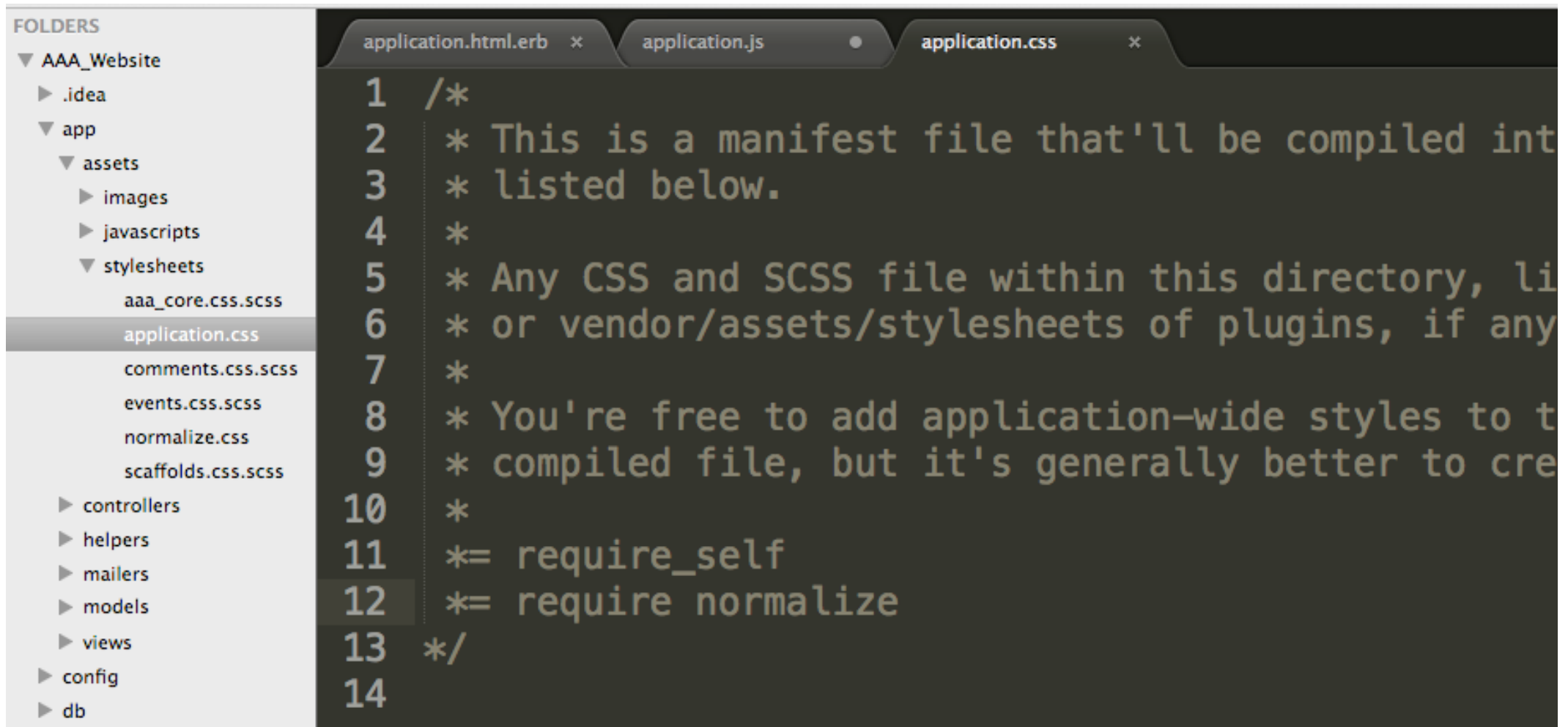
1 // This is a manifest file that'll be compiled into applica
2 // listed below.
3 //
4 // Any JavaScript/Coffee file within this directory, lib/as
5 // or vendor/assets/javascripts of plugins, if any, can be
6 //
7 // It's not advisable to add code directly here, but if you
8 // the compiled file.
9 //
10 // WARNING: THE FIRST BLANK LINE MARKS THE END OF WHAT'S TO
11 // GO AFTER THE REQUIRES BELOW.
12 //
13 //= require modernizr-2.5.3
14 //= require jquery
15 //= require jquery_ujs
16 //= require_tree .
17
```

Normalizing Browsers

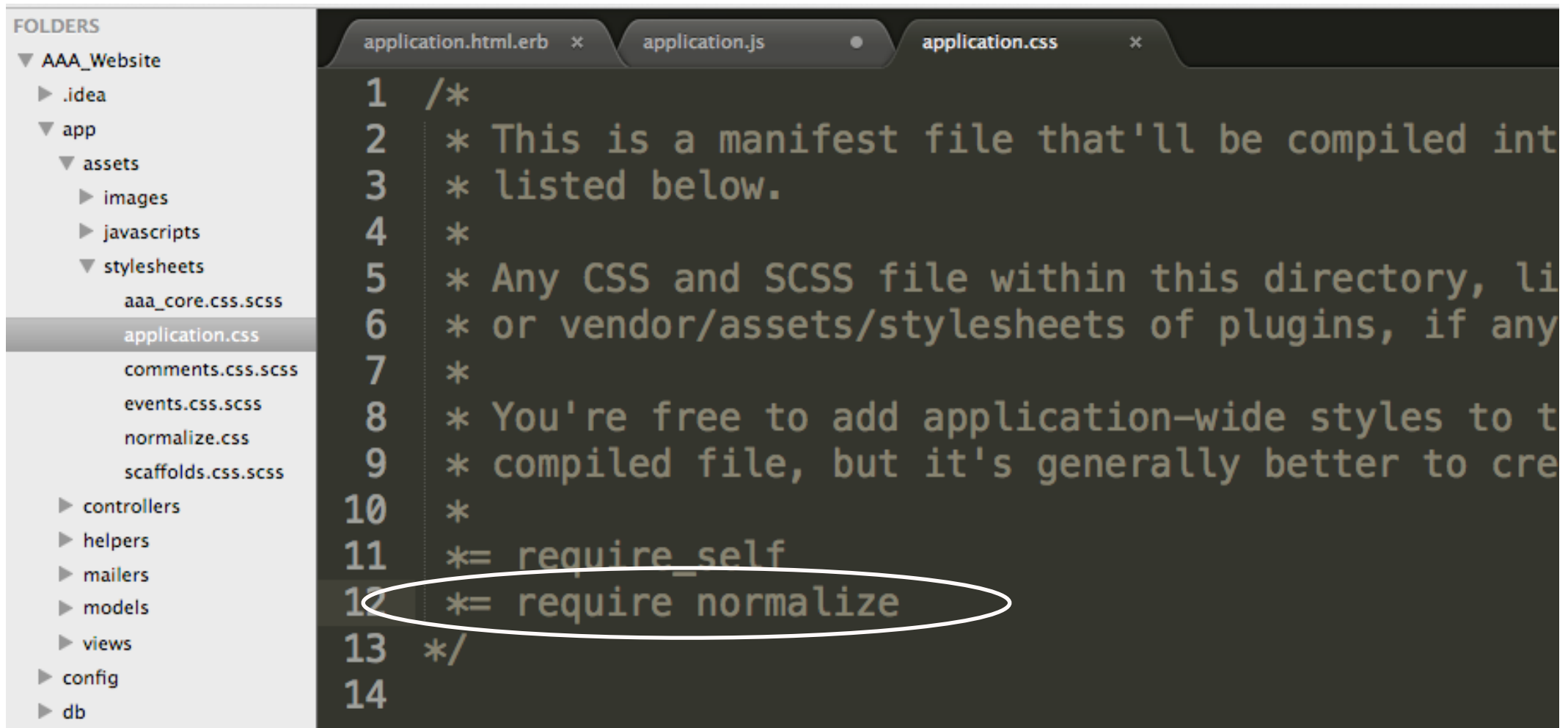
Why Normalize (aka reset)?

- Browsers do not agree on the basic presentation defaults
- Eric Meyer's original reset even removed **bold** from `` and *italics* from ``
 - There is no practical need to go that far
- We will use Normalize.css
 - “preserves useful defaults; normalizes styles for a wide range of elements; corrects bugs and common browser inconsistencies; ...”
 - <http://necolas.github.com/normalize.css/>

app/assets/stylesheets/application.css



app/assets/stylesheets/application.css



Rolling your own Boilerplate

Borrowing from HTML5 Boilerplate

- We have fixed some of the issues, but there are many more
- Using a Boilerplate without fully understanding what it all does is not recommended
- So,
- We are going to pull out of HTML5 Boilerplate what we need, in an as needed basis
 - <http://html5boilerplate.com/>

Ashtead Allotments Association

Healthy Eating and Exercise preserve Life's Balance

[Home](#)
[Events](#)
[Discussion](#)
[Shop](#)

[Sign up](#) or [sign in](#)

AaaCore#index

Find me in `app/views/aaa_core/index.html.erb`

© 2012 Ashtead Allotment's Association

What We Have Done

- Foundation is now in place
 - Modulo pasting over a few more cracks as we go
- Now we are ready to trowel on the styling