



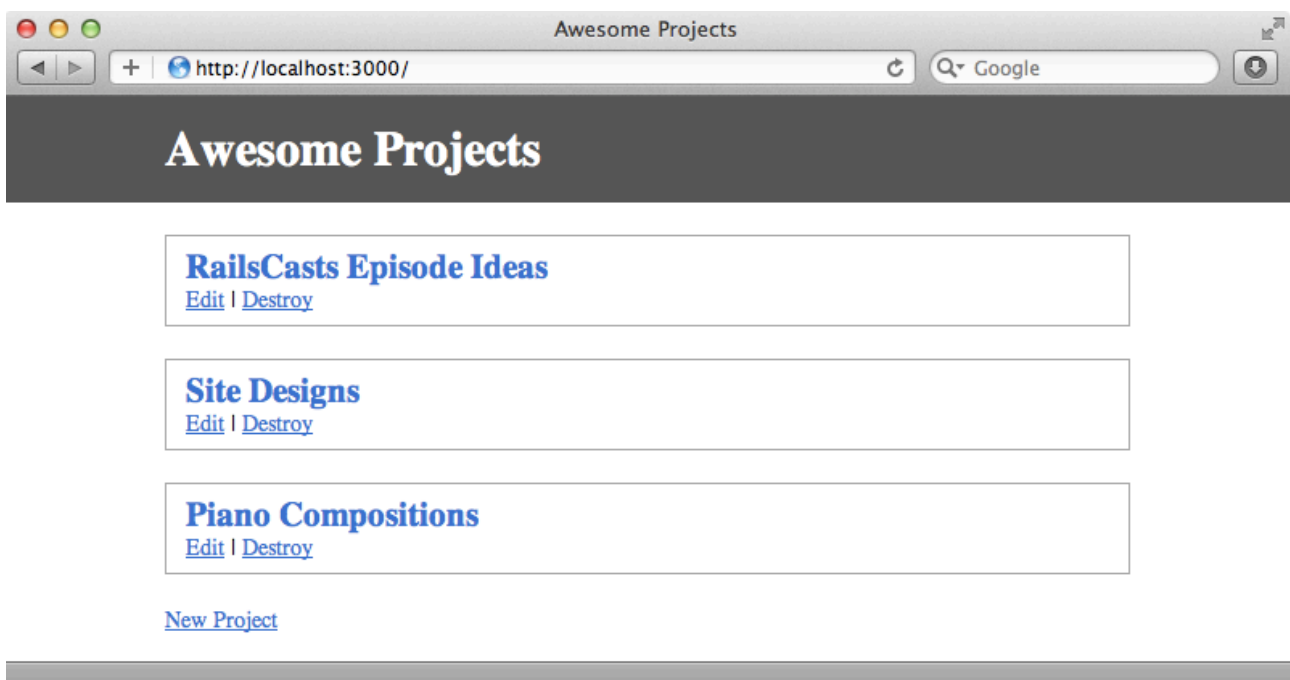
Episode 330

Better SASS  
With Bourbon

If you're tired of typing vendor-specific prefixes for CSS properties you should take a look at the Bourbon library<sup>1</sup> which includes several SASS mixins and functions to make working with CSS more convenient. In this episode we'll show you how to use it in a Rails application.

## Adding Bourbon to a Rails Application

The application we'll be working with is shown below. Its design needs some work so we'll add Bourbon to the app and use some of its features to improve the way the page looks.



Installing Bourbon is easy. All we need to do is add it to the gemfile and run bundle.

```
/Gemfile
```

---

<sup>1</sup> <http://thoughtbot.com/bourbon/>

```
source 'https://rubygems.org'

gem 'rails', '3.2.2'

# Bundle edge Rails instead:
# gem 'rails', :git => 'git://github.com/rails/rails.git'

gem 'sqlite3'

# Gems used only for assets and not required
# in production environments by default.
group :assets do
  gem 'sass-rails', '~> 3.2.3'
  gem 'coffee-rails', '~> 3.2.1'

  # See https://github.com/sstephenson/execjs#readme for more
  # supported runtimes
  # gem 'therubyracer'

  gem 'uglifyer', '>= 1.0.3'
end

gem 'jquery-rails'
gem 'bourbon'
```

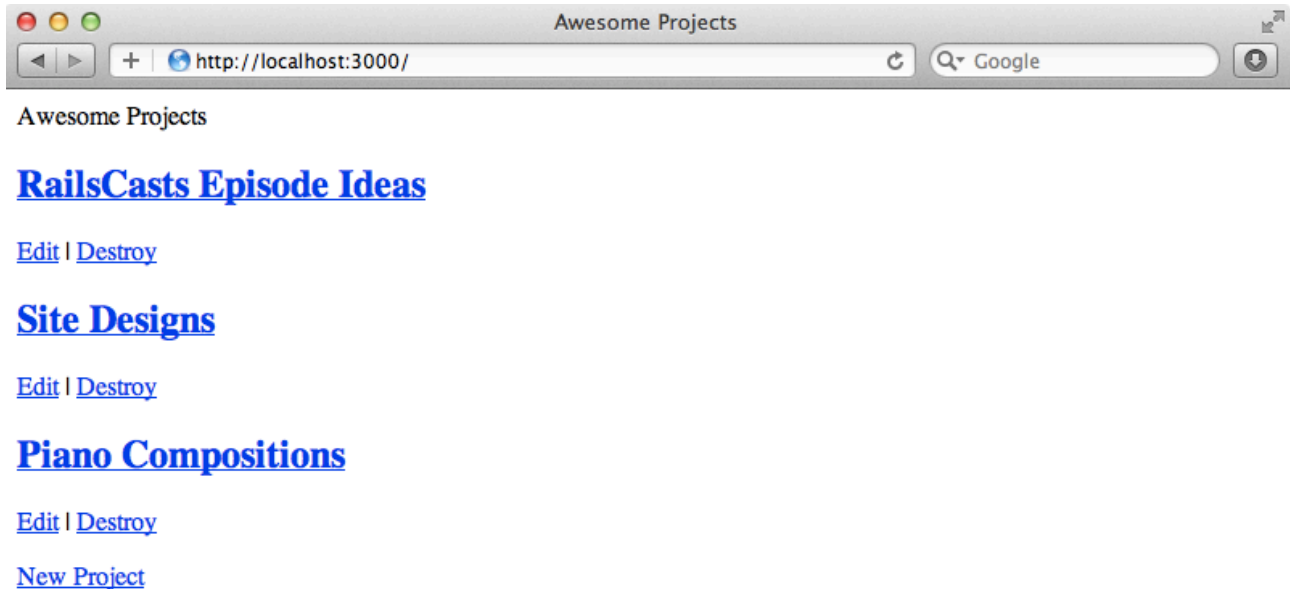
To use Bourbon with Rails' asset pipeline we'll need to change the way that the default `application.css` file works. By default this file will use a Sprockets manifest to load each of the other stylesheet assets. The problem with this is that Sprockets compiles each SASS file into CSS individually which makes it difficult to share the Bourbon mixins across the SASS files. To fix this we can use SASS to load all the `scss` files instead of Sprockets.

We'll need to rename our the `application.css` file to `application.css.scss`, then we can then remove the Sprockets manifest and use SASS's `@import` command to pull in the files that we want to include. For now we'll just include the bourbon file provided by the gem.

```
/app/assets/stylesheets/application.css.scss
```

```
@import "bourbon";
```

We'll need to restart the Rails server for these changes to be picked up. After we've done that when we reload the page all the styling has gone.



If we look at the `/assets/application.css` file in the browser we'll see that it's blank. Bourbon doesn't add any CSS to our application directly, it just makes it more convenient for us to add our own CSS through SASS. It's still up to us to write all the CSS we need for our application so we'll put the layout and project files back in `application.css.scss`. We set up these CSS files earlier but now we'll be able to use Bourbon in them.

```
/app/assets/stylesheets/application.css.scss
```

```
@import "bourbon";  
@import "layout";  
@import "projects";
```

When we reload the page now it will bring us back to the design we had earlier.

## Modifying The Header

The first improvement we'll make to our design is to change the font. The page currently uses the browser's default font, which is usually Times New Roman.

Bourbon includes a font-family add-on<sup>2</sup> which provides several variables for setting the type of font. These include fallback fonts for when the first font required isn't available. We'll use one of these to set the body's font.

/app/assets/stylesheets/layout.css.scss

```
body {
  margin: 0;
  padding: 0;
  background-color: #FFF;
  font-size: 14px;
  font-family: $verdana;
}
```

Next we'll add a gradient to the header. Bourbon provides a linear-gradient module<sup>3</sup> and we can use this to add a gradient and set its colours. We'll add a gradient that fades from light grey to a darker grey. Note that we haven't set a direction for the gradient; this means that the default value of top will be used.

/app/assets/stylesheets/layout.css.scss

```
#header {
  color: #FFF;
  padding: 15px 100px;
  font-size: 30px;
  font-weight: bold;
  @include linear-gradient(#777, #444);
}
```

A default background colour is included in linear-gradient so we've removed the background-color property from the header. We also want a drop shadow under the header and Bourbon provides a mixin<sup>4</sup> for adding these. We can use this in a similar way to the box-shadow CSS3 property but the mixin will generate all the vendor-specific prefixes for us. We'll add a black shadow with no offset and with a six pixel shadow with a three pixel spread.

---

<sup>2</sup> <http://thoughtbot.com/bourbon/#font-family>

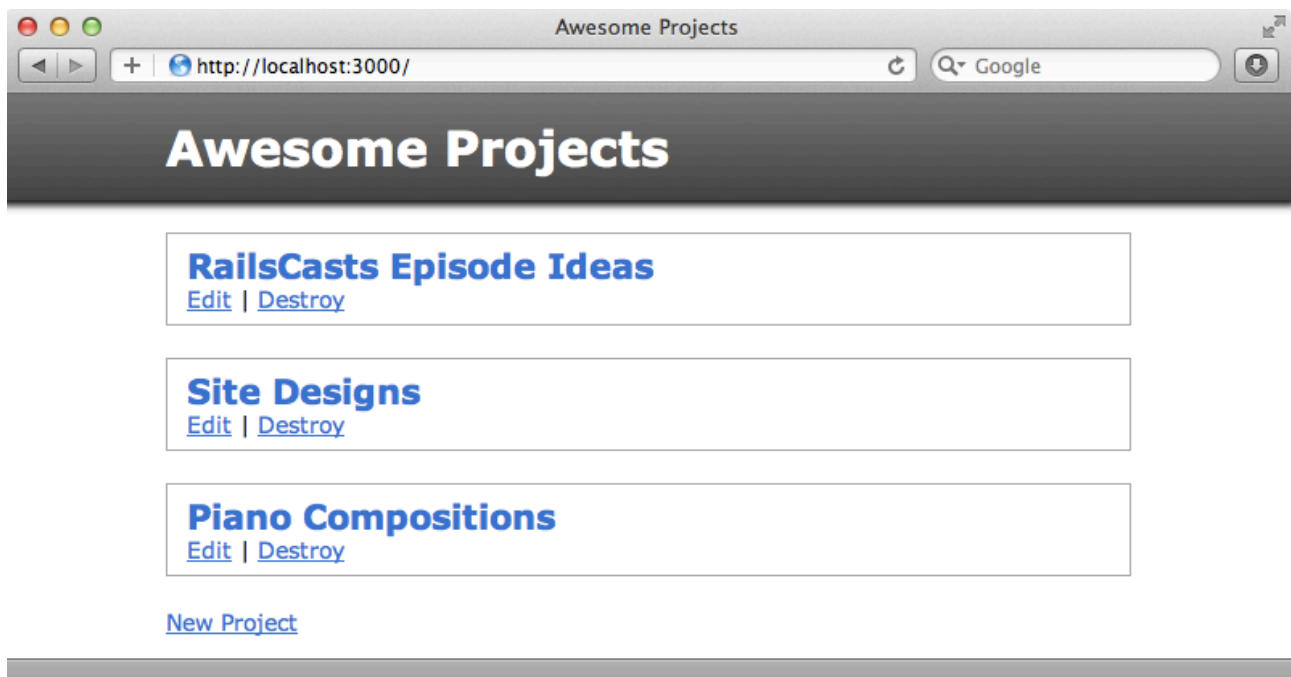
<sup>3</sup> <http://thoughtbot.com/bourbon/#linear-gradient>

<sup>4</sup> <http://thoughtbot.com/bourbon/#box-shadow>

/app/assets/stylesheets/layout.css.scss

```
#header {
  background-color: #555;
  color: #FFF;
  padding: 15px 100px;
  font-size: 30px;
  font-weight: bold;
  @include linear-gradient(#777, #444);
  @include box-shadow(0 0 6px 3px #000);
}
```

When we reload the page now these changes can be seen.



## Modifying The Items

Next we'll make some changes to the list of items. First we'll add rounded corners to the box that surrounds each project. Needless to say Bourbon has a `border-radius`<sup>5</sup> module to do just this. The project-specific styling is in a `projects.css.scss` file so we'll make this change there and add a six pixel radius.

---

<sup>5</sup> <http://thoughtbot.com/bourbon/#border-radius>

/app/assets/stylesheets/projects.css.scss

```
.project {
  border: solid 1px #AAA;
  margin: 20px 0;
  padding: 7px 12px;
  @include border-radius(6px);

  &:hover { background-color: #F8FCCF; }

  h2 {
    margin: 0;
    a { text-decoration: none; }
  }
}
```

Next we'll change the way that the background colour of each item changes when we hover over it. Instead of it changing immediately we'd like it to fade in to that colour. We can use transitions<sup>6</sup> to do this, passing in the attribute we want to change, the time it should take to do the change and the easing effect.

/app/assets/stylesheets/projects.css.scss

```
.project {
  border: solid 1px #AAA;
  margin: 20px 0;
  padding: 7px 12px;
  @include border-radius(6px);
  @include transition(all, 500ms, ease-in-out);

  &:hover { background-color: #F8FCCF; }

  h2 {
    margin: 0;
    a { text-decoration: none; }
  }
}
```

Now the background colour will fade in over half a second when we hover over a project and fade out over the same time when the cursor moves out.

---

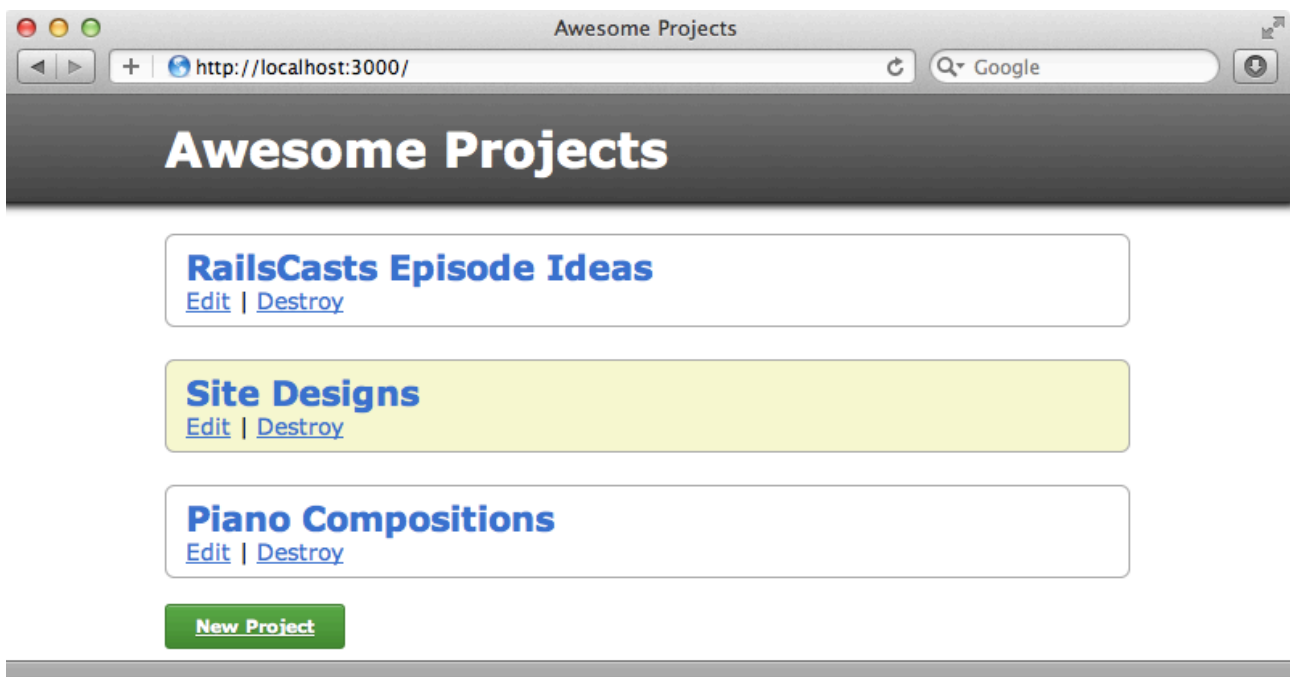
<sup>6</sup> <http://thoughtbot.com/bourbon/#transitions>

The last change we'll make to this page is to improve the "New Project" link. We'd like it to look more like a button instead and Bourbon includes a buttons add-on<sup>7</sup> that we can use to make different styles of button. Our link has a `new_project` class so we'll use that to style it and turn the link into a simple green button.

```
/app/assets/stylesheets/projects.css.scss
```

```
.new_project {  
  @include button(simple, #3FB344);  
}
```

When we reload the page now we'll see the rounded corners and the styled link. (You'll have to take the background colour fade on trust.)



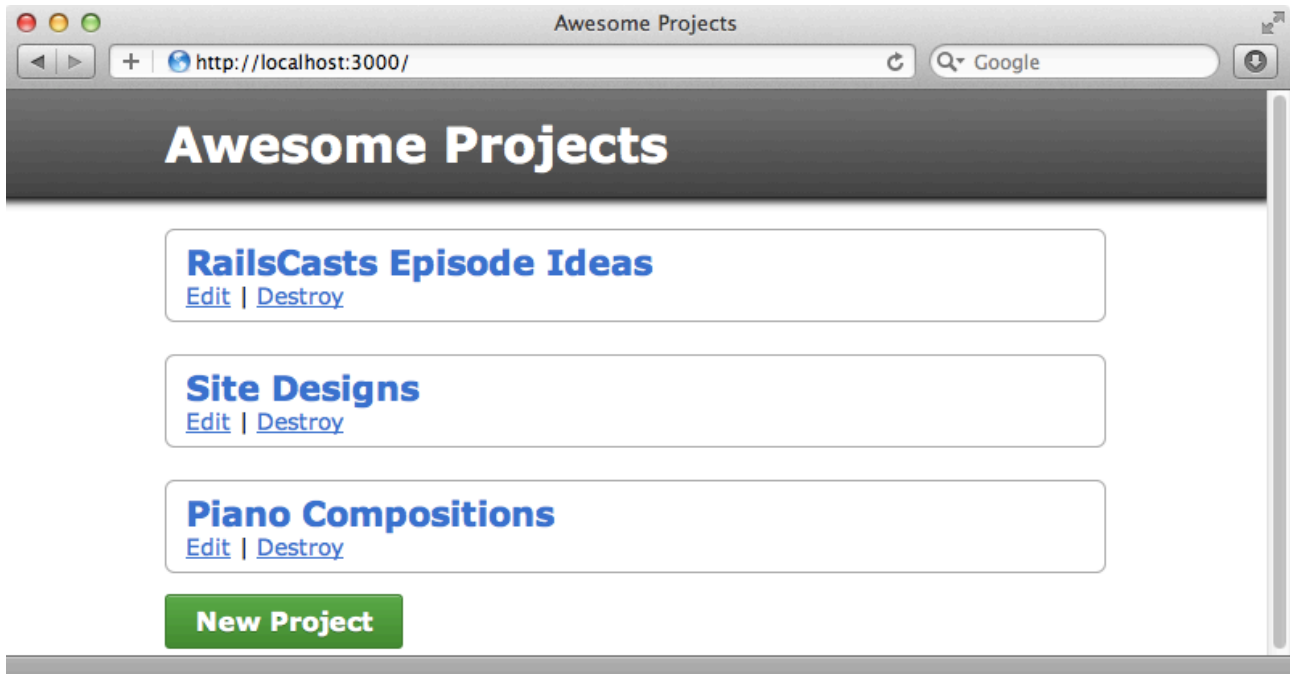
The button has inherited some attributes from the anchor tag such as the underlined text and we want it to be a little larger. We'll add some more CSS to do this.

<sup>7</sup> <http://thoughtbot.com/bourbon/#buttons>

```
/app/assets/stylesheets/projects.css.scss
```

```
.new_project {  
  @include button(simple, #3FB344);  
  text-decoration: none;  
  font-size: 16px;  
}
```

We now have a nice button for adding a new project.



## An Example Of The CSS That Bourbon Generates

Now that the page is looking better let's look at the CSS that Bourbon has generated. We can't show it all here as there's a lot of it but as an example here's the generated CSS for the header and we can see the code for the gradient that we added.

```
#header {
  background-color: #555;
  color: #FFF;
  padding: 15px 100px;
  font-size: 30px;
  font-weight: bold;
  background-color: #777777;
  background-image: -webkit-gradient(linear, left top, left
bottom, color-stop(0%, #777777), color-stop(100%, #444444));
  background-image: -webkit-linear-gradient(top, #777777,
#444444);
  background-image: -moz-linear-gradient(top, #777777, #444444);
  background-image: -ms-linear-gradient(top, #777777, #444444);
  background-image: -o-linear-gradient(top, #777777, #444444);
  background-image: linear-gradient(top, #777777, #444444);
  -webkit-box-shadow: 0 0 6px 3px black;
  -moz-box-shadow: 0 0 6px 3px black;
  box-shadow: 0 0 6px 3px black;
}
```

There's a lot of code here that we've not had to write manually thanks to Bourbon.

SASS also includes several functions that can help out with generating CSS when dealing with colour changes, variations and adjustments. There's more information in episode 268<sup>8</sup> on what can be done with SASS alone.

---

<sup>8</sup> <http://railscasts.com/episodes/268-sass-basics>