

jQuery Mobile

The Theme Framework

Lesson 1, Activity 2: The Theme Framework

Themes in jQuery Mobile

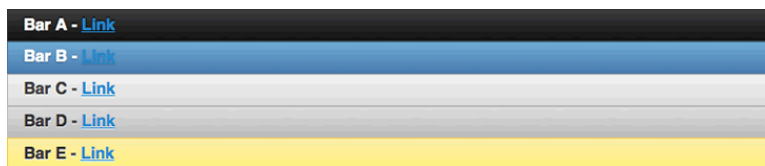
jQuery Mobile themes are set via external CSS stylesheets. Themes consist of settings both for global style attributes - the radius of rounded corners, the color for the on-state of buttons, etc. - and up to 26 "swatches" (lettered from A to Z), each with a unique color scheme. Swatches define the colors, font settings, and textures for elements such as toolbars, content blocks, and buttons. jQuery Mobile recommends building themes with at least three swatches, A to C.

Without any overriding settings, jQuery Mobile uses the stock theme "a" swatch (a dark-colored swatch) for toolbars and the stock theme "c" swatch (a light-colored swatch) for page content. In our work so far, the CSS rules for these swatches have come from the CDN-hosted CSS file we have included in the head of each of our jQuery Mobile. (For performance improvements, we've included the minified CSS file in all of our pages - here's the [complete uncompressed CSS](#).) This default "a" swatch's CSS for toolbars includes, in part, this rule:

```
.ui-bar-a {
border: 1px solid #333 /*{a-bar-border}*/;
background: #111111 /*{a-bar-background-color}*/;
color: #ffffff /*{a-bar-color}*/;
font-weight: bold;
text-shadow: 0 /*{a-bar-shadow-x}*/ -1px /*{a-bar-shadow-y}*/ 1px /*{a-bar-shadow-radius}*/ #000000 /*{a-bar-shadow-color}*/;
background-image: -webkit-gradient(linear, left top, left bottom, from( #3c3c3c /*{a-bar-background-start}*/), to( #111 /*{a-bar-background-end}*/)); /* Saf4+, Chrome */
background-image: -webkit-linear-gradient( #3c3c3c /*{a-bar-background-start}*/, #111 /*{a-bar-background-end}*/); /* Chrome 10+, Saf5.1+ */
background-image: -moz-linear-gradient( #3c3c3c /*{a-bar-background-start}*/, #111 /*{a-bar-background-end}*/); /* FF3.6 */
background-image: -ms-linear-gradient( #3c3c3c /*{a-bar-background-start}*/, #111 /*{a-bar-background-end}*/); /* IE10 */
background-image: -o-linear-gradient( #3c3c3c /*{a-bar-background-start}*/, #111 /*{a-bar-background-end}*/); /* Opera 11.10+ */
background-image: linear-gradient( #3c3c3c /*{a-bar-background-start}*/, #111 /*{a-bar-background-end}*/);
}
```

The background color (background: #111111) is dark and the text color (color: #ffffff) is white; vendor-specific rules (Microsoft's -ms-linear-gradient, Opera's -o-linear-gradient) and general rules cover all the bases for adding a gradient to the background color.

Without adding any new themes or custom CSS, we can use the existing jQuery Mobile stock theme's swatches to style individual pages or elements:



For these ui-bar elements, we add class ui-bar-a (or ui-bar-b, ui-bar-c, etc.) to style the bar with the desired built-in jQuery Mobile theme's swatch style rules:

```
Bar A - Link
Bar B - Link
Bar C - Link
Bar D - Link
Bar E - Link
```

We can also add the attribute data-theme="a" (or "b", or "c", etc.) to elements: you can set pages, bars, listviews, buttons, and form elements to have a specific theme swatch. Anything but header and footer bars inherit the theme swatch of their parent container; headers and footers need to be set explicitly. For example:

- [Page 1](#)
- [Page 2](#)

Lesson 1, Activity 3: Using Default Theme Swatches to Style Pages & Elements

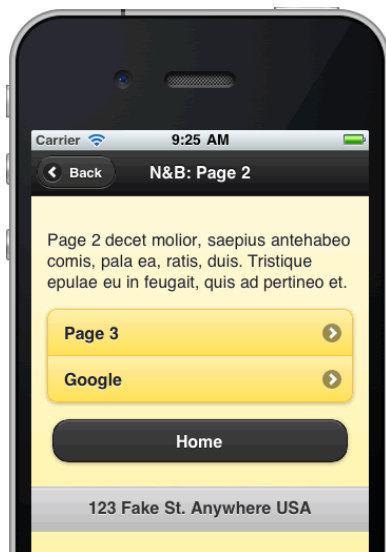
Duration: 15 to 20 minutes.

In this exercise, you will use the default jQuery Mobile theme to style pages and elements.

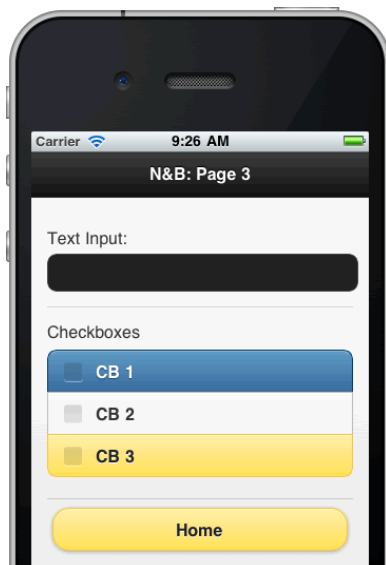
1. Open [ThemeFramework/Exercises/swatches.html](#) in a file editor.
2. Add a `listview` to the `#home` page; style it with swatch "b" as shown:



3. Style `#page2` with swatch "e". Add a button, linking to the `#home` page, at the bottom of `#page2`; style with swatch "a". Style the footer of `#page2` with swatch "d". The result should look like this:



4. Add a `form` to `#page3` with a text input and a set of three checkboxes: style the text input as "a" and the checkboxes as "b", "d", and "e", respectively:

**Solution:**[ThemeFramework/Solutions/swatches.html](#)

---- CODE OMITTED ----

```

<div data-role="page" id="home">
  <div data-role="header">
    <h2>Nan & Bob's</h2>
  </div>
  <div data-role="content">
    <ul data-role="listview" data-inset="true" data-theme="b">
      <li><a href="#page2">Page 2</a></li>
      <li><a href="#page3">Page 3</a></li>
    </ul>
  </div>
  <div data-role="footer">
    <h3>
      123 Fake St. Anywhere USA
    </h3>
  </div>
</div>
<div data-role="page" id="page2" data-add-back-btn="true" data-theme="e">
  <div data-role="header">
    <h2>N&B: Page 2</h2>
  </div>
  <div data-role="content">
    <p>Page 2 decet molior, saepius antehabeo comis, pala ea, ratis, duis. Tristique epulae eu in feugait, quis ad pertineo et.</p>
    <ul data-role="listview" data-inset="true">
      <li><a href="#page3">Page 3</a></li>
      <li><a href="http://www.google.com">Google</a></li>
    </ul>
    <a href="#home" data-role="button" data-theme="a">Home</a>
  </div>
  <div data-role="footer" data-theme="d">
    <h3>
      123 Fake St. Anywhere USA
    </h3>
  </div>
</div>
<div data-role="page" id="page3" data-add-back-btn="true">
  <div data-role="header">
    <h2>N&B: Page 3</h2>
  </div>
  <div data-role="content">
    <form action="#">
      <div data-role="fieldcontain">
        <label for="name">Text Input:</label>
        <input type="text" name="name" id="name" data-theme="a">
      </div>
      <div data-role="fieldcontain">
        <fieldset data-role="controlgroup">
          <legend>Checkboxes</legend>
          <input type="checkbox" name="cb" id="cb-1" data-theme="b">
          <label for="cb-1">CB 1</label>
          <input type="checkbox" name="cb" id="cb-2" data-theme="d">
          <label for="cb-2">CB 2</label>
          <input type="checkbox" name="cb" id="cb-3" data-theme="e">
          <label for="cb-3">CB 3</label>
        </fieldset>
      </div>
    </form>
    <a href="#home" data-role="button" data-theme="e">Home</a>
  </div>
  <div data-role="footer">
    <h3>
      123 Fake St. Anywhere USA
    </h3>
  </div>
</div>
</body>
</html>

```

On the #home page, we add data-theme="b" to the listview.

We add data-theme="e" to #page2; note how the page elements (those not otherwise explicitly styled) inherit the swatch color. We style the footer with data-theme="d".

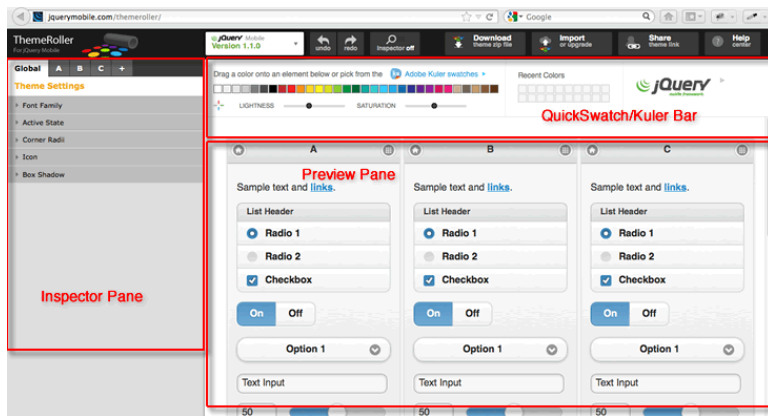
We style the individual form elements on #page3 with various swatch letters to produce the desired visual look.

Lesson 1, Activity 5: ThemeRoller

The default jQuery Mobile theme offers a considerable degree of customization options, but the range of colors and style are, of course, limited. What if we want to add our own colors? And what if we want to change the font or the dropshadow color or roundedness of buttons?

For this we turn to [ThemeRoller](#), a powerful, free, what-you-see-is-what-you-get UI builder available on the jQuery Mobile website. ThemeRoller offers a visual interface from which we can set a variety of color, font, and other options to define our own custom theme; once happy with our design, we can download a single CSS file, link to it in our pages, and thus apply the design choices to our sites.

Open up [ThemeRoller](#) in your browser to try it out:



The "Help" docs (via the link at upper right) offer a helpful overview of the interface:

The ThemeRoller interface has three major zones: the left column contains the inspector panel, along the top is the QuickSwatch/Kuler bar, and below this is the preview window.

*Use the **inspector pane** to set global theme settings on the first tab and tweak the individual style options for each swatch. Above the tabs, there are links to download, share, or import a theme.*

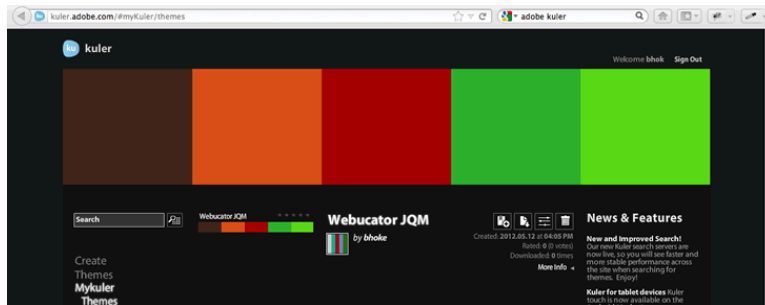
*In the **QuickSwatch bar**, you can turn the inspector feature on to automatically expand the relevant inspector section when you click on an element in the preview pane. Drag and drop a color from the QuickSwatch panel onto an element in the preview pane and the tool will automatically calculate text color and shadow, borders, gradients and even button states. The sliders make it easy to adjust the lightness and saturation of the colors. Click the Adobe Kuler Swatches to load premade color palettes from Adobe's popular color palette sharing site.*

*The **preview pane** shows a sample of common jQuery Mobile widgets that shows a live update each time you make a change to the theme so you can quickly test and tweak the theme.*

When using ThemeRoller, you can either set color options by typing hex codes or by dragging a color onto the element. Similarly, you can type explicit values for the roundedness of corner or drag a slider.

Adobe Kuller

ThemeRoller offers convenient integration with [Adobe Kuller](#), a free online color-theme-generation tool. To use Adobe Kuller, login with a valid Adobe account (register for free, if needed). Once logged in, you can search and browse public themes - filtering by "newest", "highest rated", or other criteria - or create your own. I created a theme ("Webucator JQM") you can [view online](#) - I'll use it in the demo below.

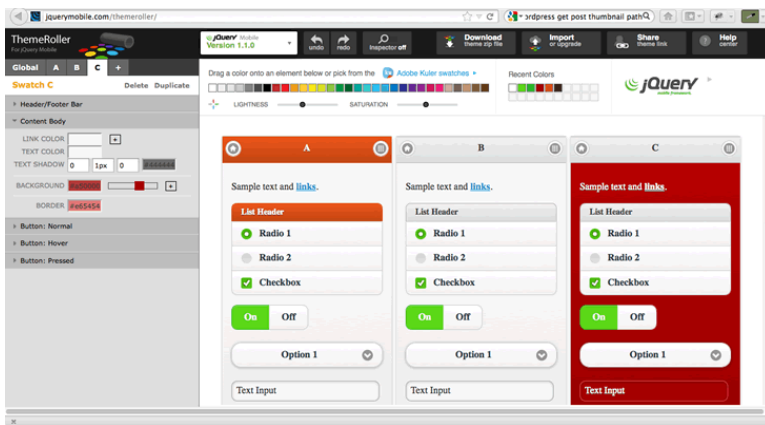


With Adobe Kuller you can create a theme from an uploaded or Flickr-stored image or from a specific base color. You can select rules ("analogous", "triad", etc.) from color theory to generate the scheme, drag axes around a color wheel, or set colors explicitly with HSV, RGB, Hex, or other settings.

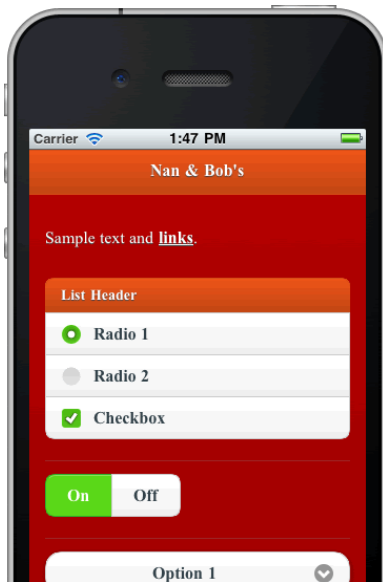
Once you create an Adobe Kuller theme, you can use its colors in ThemeRoller, adding them to toolbars, buttons, or other elements as desired. To use a saved Adobe Kuller theme from within ThemeRoller:

1. Click the "Adobe Kuller swatches" link at the top of ThemeRoller.
2. Choose "Search Title" from the drop-down.
3. Enter (in this case) "webucator" as the search term.
4. Click each color square to add it to ThemeRoller's "Recent Colors" palette, from which you can use the colors in desired elements of the theme.

Let's look at an example. We used the colors from the Adobe Kuller theme we created above to render a custom theme in ThemeRoller. While it might offend some folks' artistic sensibilities, the process works quite nicely:



We click "Download theme zip file" at the top of the ThemeRoller interface, save the .zip file, and unpack the .zip archive. We edit the [index.html](#) file to include our standard Nan & Bob's content - including, in this case, some widget and form elements to highlight the theme's color style rules. Here's the resulting page - easily styled as desired with minimal coding:



Code Sample:

[ThemeFramework/Demos/theme-webucator/index.html](#)

```

---- CODE OMITTED ----

<head>
<meta charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<title>Nan & Bob's</title>
<link rel="stylesheet" href="themes/webucator-adobe-kuller.min.css" />
<link rel="stylesheet" href="http://code.jquery.com/mobile/1.1.0/jquery.mobile.structure-1.1.0.min.css" />
<script src="http://code.jquery.com/jquery-1.7.1.min.js"></script>
<script src="http://code.jquery.com/mobile/1.1.0/jquery.mobile-1.1.0.min.js"></script>
</head>
---- CODE OMITTED ----

```

The only work we need do here to make the custom theme work is to get the linked stylesheets right. We link to the downloaded (and conveniently minified) stylesheet, [themes/webucator-adobe-kuller.min.css](#). Also, instead of linking to the standard jQuery Mobile stylesheet ([jquery.mobile-1.1.0.min.css](#)), we instead link to the (also CDN-hosted) "structure" stylesheet: [jquery.mobile.structure-1.1.0.min.css](#). This ensures that we get the CSS we need to "run" jQuery Mobile, but without the default color/font/etc. theme; we now get these style rules from our custom, downloaded stylesheet.

Lesson 1, Activity 7: **Icons**

As we've seen, jQuery Mobile comes with 18 stock icons out of the box, corresponding to the data-icon="[icon-name]" attribute:

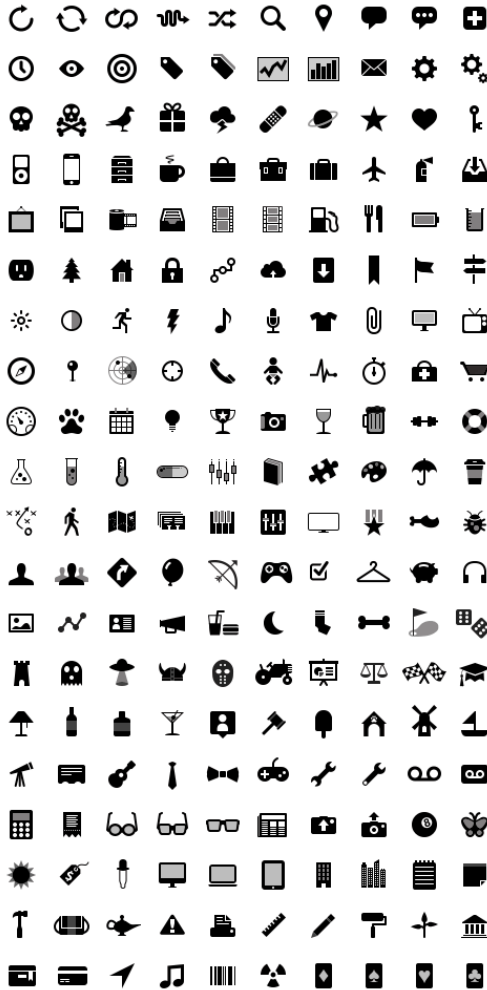
jQuery Mobile Button Icons		
Icon	Code	
Left arrow	data-icon="arrow-l"	◀
Right arrow	data-icon="arrow-r"	▶
Up arrow	data-icon="arrow-u"	▲
Down arrow	data-icon="arrow-d"	▼
Delete	data-icon="delete"	✕
Plus	data-icon="plus"	⊕
Minus	data-icon="minus"	⊖
Check	data-icon="check"	✓
Gear	data-icon="gear"	⚙
Refresh	data-icon="refresh"	↺
Forward	data-icon="forward"	⏭
Back	data-icon="back"	⏮
Grid	data-icon="grid"	⌘
Star	data-icon="star"	★
Alert	data-icon="alert"	⚠
Info	data-icon="info"	ℹ
Home	data-icon="home"	🏠
Search	data-icon="search"	🔍

Icons are presented as a single white icon sprite, positioned with CSS, to save on download time:

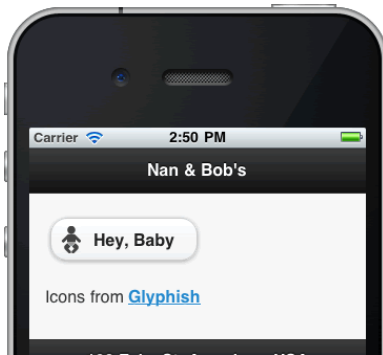


jQuery Mobile automatically adds a semi-transparent black circle behind the icon to enhance contrast for readability, regardless of the current background color.

The stock icons are great - but often we'll wish to use different icons, either because none of the default icons fits our purpose, because we want a different creative look, or both. You can create your own icons - either as sprite images or as single small images - or you can download icons to use with jQuery Mobile. (See Andy Matthews excellent [article on creating your own icons](#) for more information.) [Glyphish](#) offers a popular set of icons, with both free (the license for which require attribution) and paid options. Here's the free set:



Let's use one of the Glyphish icons (my favorite) in an example - open up ThemeFramework/Demos/icons/index.html to check out the code:



Code Sample:

ThemeFramework/Demos/icons/index.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <title>Nan & Bob's</title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.1.0/jquery.mobile-1.1.0.min.css" />
  <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.1.0/jquery.mobile-1.1.0.min.js"></script>
  <style>
    .ui-icon-nb-baby {
      background: url("glyphish-icons/76-baby.png") no-repeat scroll 50% 50% transparent;
      border-radius:0;
      width:18px;
      height:26px;
      right:14px !important;
      top:40% !important;
    }
  </style>
```



```

</head>
<body>
  <div data-role="page" id="home">
    <div data-role="header">
      <h2>Nan & Bob's</h2>
    </div>
    <div data-role="content">
      <a href="#" data-role="button" data-icon="nb-baby" data-inline="true">Hey, Baby</a>
      <p>Icons from <a href="http://glyphish.com/">Glyphish</a></p>
    </div>
    <div data-role="footer">
      <h3>
        123 Fake St. Anywhere USA
      </h3>
    </div>
  </div>
</body>
</html>

```

Here, we use one of the free Glyphish icons ([76-baby.png](#)) as the icon for our button. We add the `data-icon="nb-baby"` to the link - where `nb-baby` is a name we made up to reference our custom icon - in the main body content, along with attributes to make the link a button and to display it in inline style.

We then add some CSS to handle this button. `.ui-icon-nb-baby` is the class that jQuery Mobile automatically adds to a `span` it adds to the markup to contain our button; we add a `background` rule in CSS to show our desired icon. Because the icons from Glyphish are slightly larger than the default 18px by 18px size of the stock jQuery Mobile icons, we add some extra rules to display the icon appropriately, including a `border-radius:0` rule to prevent the edges of the icon from being chopped off.

Custom CSS

As jQuery Mobile is an HTML5 framework, there's nothing to stop us from adding our own custom CSS rules to pages to add any visual style we wish. We can style text, images, widgets, form controls, or other elements by hand, just as we would any desktop site. Of course, working with the framework - default themes and icons and ThemeRoller custom themes - will be easier than hand-crafting CSS for every creative aspect of our sites. But achieving that stunning, unique look for your site probably means a bit of custom CSS. We'll look at some examples in the next exercise.

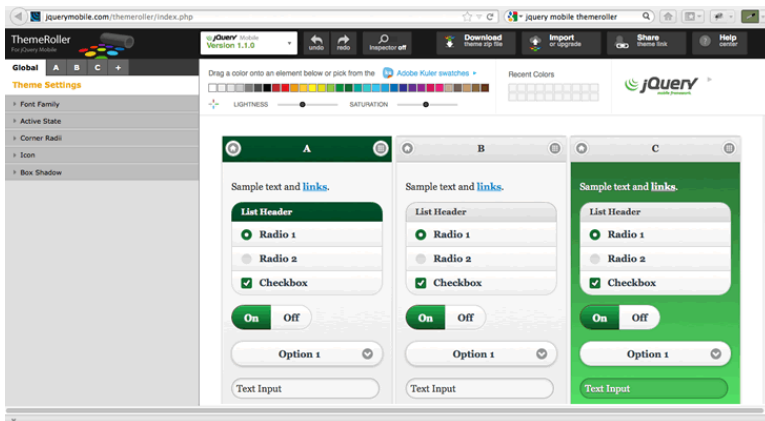
sdf

Lesson 1, Activity 8:

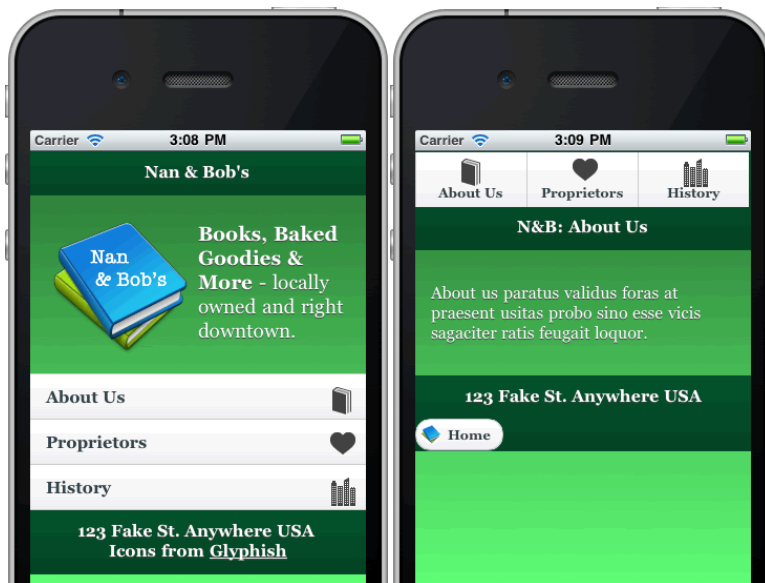
Lesson 1, Activity 9: Styling the Nan & Bob's Site

Duration: 25 to 35 minutes.

Let's give some sample pages from the Nan & Bob's site a facelift using ThemeRoller, custom icons, and some manual updates to the site's style. Nan & Bob want to adopt a green-based look for their site and also to incorporate a new logo into the site. We'll start with a ThemeRoller theme:



We'll use this theme as a starting point, add the Nan & Bob's logo image, and use some custom icons from Glyphish to style the home page (on the left, in the screenshot below) and interior pages (on the right) like this:



In this exercise, you will style the Nan & Bob's site with a custom theme, a large home page logo image, and custom icons.

1. Open [ThemeFramework/Exercises/newdesign/index.html](#) in a file editor.
2. Create a custom theme in [ThemeRoller](#) - you can make it heavy on the green, or choose your own colors.
3. Download and save your custom theme; save the unpacked downloaded files to [ThemeFramework/Exercises/newdesign/themes](#).
4. Link to the downloaded minified stylesheet in the head of [index.html](#) where indicated.
5. Add a two-column grid in the content div of the #home page, with the logo (from the [images](#) directory) on the left and the indicated text on the right. Add a CSS rule to make the right-side text large.
6. Choose a custom icon to represent each of the pages ("about", "proprietors", and "history"). Set each of the home page [listview](#) links to display the custom icons, and add a [navbar](#) element to the header of each of the interior pages, with the appropriate custom icons.
7. Add CSS rules to display each of the custom icons from their [data-icon](#) attributes.
8. Add a button in the footer of each interior page pointing back to the home page, with the small Nan & Bob's logo ([images/nbhomeicon.png](#)) as the icon; add appropriate CSS to display the icon.
9. Test your work in a mobile browser.

Solution:

[ThemeFramework/Solutions/newdesign/index.html](#)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <title>Nan & Bob's</title>
    <link rel="stylesheet" href="themes/webucator5.min.css" />
    <link rel="stylesheet" href="http://code.jquery.com/mobile/1.1.0/jquery.mobile.structure-1.1.0.min.css" />
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js"></script>
    <script src="http://code.jquery.com/mobile/1.1.0/jquery.mobile-1.1.0.min.js"></script>
    <style>
      #homefeatured {
        padding-bottom:30px;
        padding-top:10px;
      }
      #hometagline {
```

```

        font-size:120%;
    }
    .ui-icon-nb-logo {
        background: url("images/nbhomeicon.png") no-repeat scroll 0 0 #fff;
    }
    .ui-icon-nb-about {
        background: url("glyphish-icons/96-book.png") no-repeat scroll 50% 50% transparent;
        border-radius:0;
        width:18px;
        height:26px;
        right:14px !important;
    }
    .ui-icon-nb-proprietors {
        background: url("glyphish-icons/29-heart.png") no-repeat scroll 0 0 transparent;
        border-radius:0;
        width:24px;
        height:21px;
    }
    .ui-icon-nb-history {
        background: url("glyphish-icons/178-city.png") no-repeat scroll 0 0 transparent;
        border-radius:0;
        width:23px;
        height:27px;
    }
}
</style>
</head>
<body>
    <div data-role="page" id="home">
        <div data-role="header">
            <h2>Nan & Bob's</h2>
        </div>
        <div data-role="content">
            <div class="ui-grid-a" id="homefeatured">
                <div class="ui-block-a" style="text-align:right;"></div>
                <div class="ui-block-b" id="hometagline"><strong>Books, Baked Goodies & More</strong> - locally owned and right downtown.</div>
            </div>
            <ul data-role="listview">
                <li data-icon="nb-about"><a href="#about">About Us</a></li>
                <li data-icon="nb-proprietors"><a href="#proprietors">Proprietors</a></li>
                <li data-icon="nb-history"><a href="#history">History</a></li>
            </ul>
        </div>
        <div data-role="footer">
            <h3>
                123 Fake St. Anywhere USA
                <br>Icons from <a href="http://glyphish.com/">Glyphish</a>
            </h3>
        </div>
    </div>
    <div data-role="page" id="about">
        <div data-role="header">
            <div data-role="navbar" data-iconpos="top">
                <ul>
                    <li><a href="#about" class="ui-btn-active" data-icon="nb-about">About Us</a></li>
                    <li><a href="#proprietors" data-icon="nb-proprietors">Proprietors</a></li>
                    <li><a href="#history" data-icon="nb-history">History</a></li>
                </ul>
            </div>
            <h2>N&B: About Us</h2>
        </div>
        <div data-role="content">
            <p>About us paratus validus foras at praesent usitas probo sino esse vicis sagaciter ratis feugait loquor.</p>
        </div>
        <div data-role="footer">
            <h3>
                123 Fake St. Anywhere USA
            </h3>
            <a href="#home" data-icon="nb-logo">Home</a>
        </div>
    </div>
</body>
</html>

```

We add a style link in head of the .html file to add the custom-theme stylesheet we downloaded from ThemeRoller; as before, we also link the CDN-hosted [jquery.mobile.structure-1.1.0.min.css](#) stylesheet to bring in the structural jQuery Mobile CSS.

We add a div with class `ui-grid-a` to display a two-column grid, with the logo on the left and text on the right; some custom CSS displays the content appropriately.

We use the `data-icon` to customize the icons for the home page listview and the interior page navbars, with some custom CSS to display the correct icons.

Last, we add a button to each interior-page footer, with CSS to display the small Nan & Bob's logo. Note that, as this icon is the correct size (unlike the Glyphish icons), we have to do less with CSS to display the custom icon correctly.