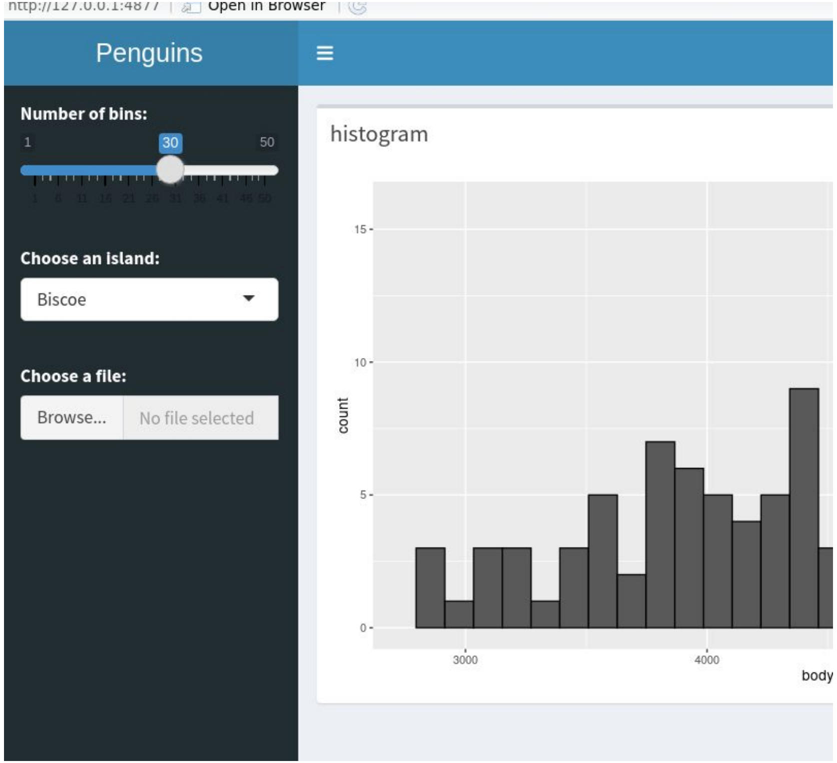
Changing the Appearance of your Shiny App

The defaults on Shiny and *shinydashboard* aren’t bad, but it results in a lot of apps looking sameish.

What if you want to customize the appearance of your app?



Number of bins:

1 30 50

Choose an island:

Biscoe

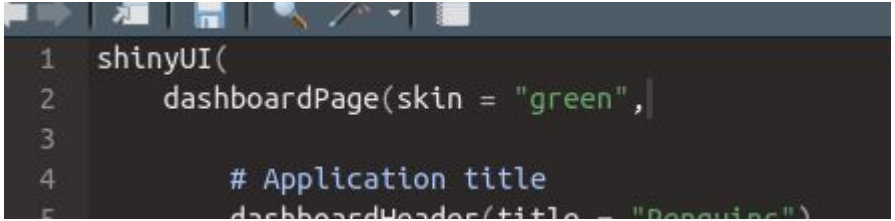
Choose a file:

Browse... No file selected

Option #1 (*shinydashboard* specific): Choose a different skin

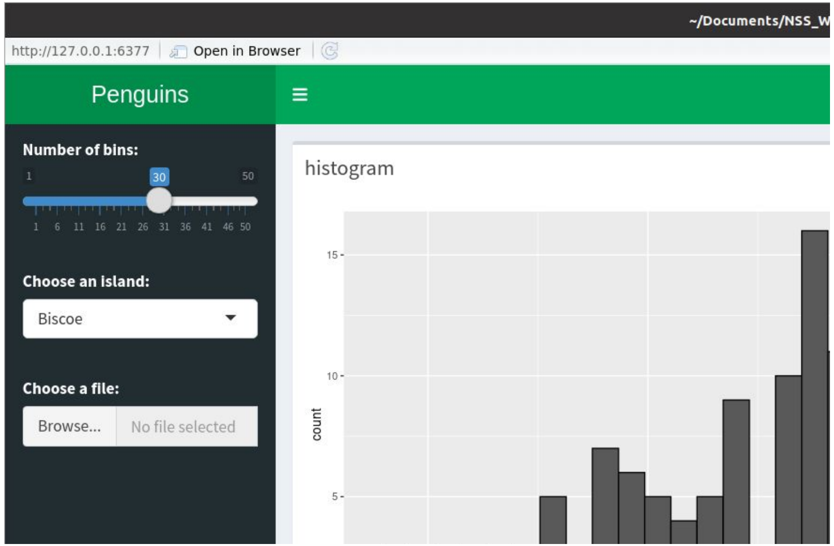
Use the *skin* argument for *dashboardPage*.

See the options here.



```
shinyUI(
    dashboardPage(skin = "green",

        # Application title
        dashboardHeader(title = "Penguins")
```



Number of bins: 30

Choose an island: Biscoe

Choose a file: Browse... No file selected

Option #2: Use a Library that lets you use/build themes

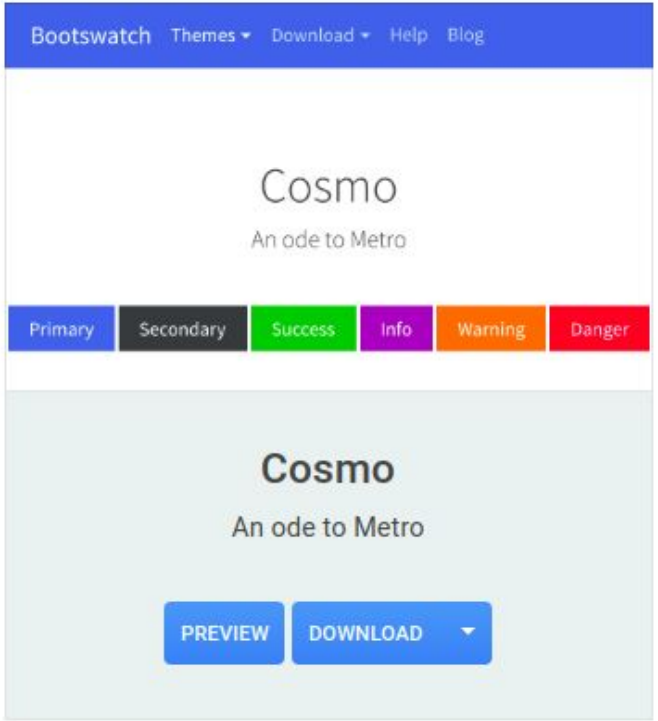
For this, I recommend the bslib library: https://rstudio.github.io/bslib/

shinythemes is another good one, or dashboardthemes if using *shinydashboard*

You can use a Bootswatch theme or modify one as you need.

To use bslib, you must use either a fluidPage, navbarPage, or bootstrapPage layout.

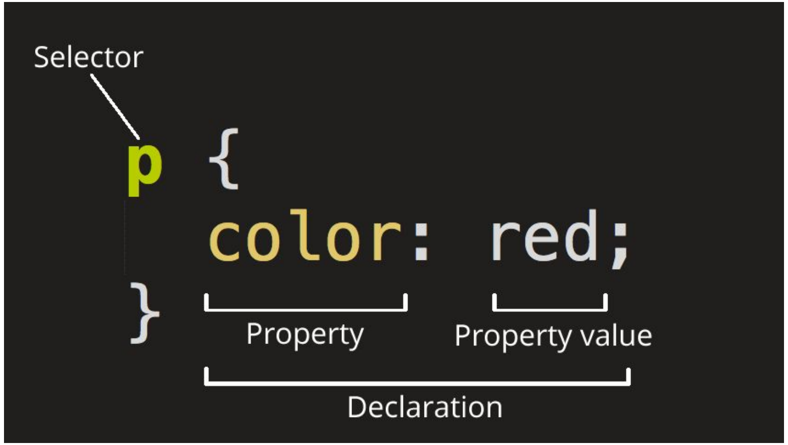
Build your own theme using `bs_theme_preview()`



Option #3: Use CSS

Warning: This is a *lot* of work.

CSS = Cascading Style Sheet



```
p {
  color: red;
}
```

https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/CSS_basics

Option #3: Use CSS

In Shiny, most of the time, you will be selecting a class to modify.

To select a class, you need to do .*classname*. You can also select tags by type with just *tagtype*.

Sometimes you need to specify a series of classes to narrow it down to the element that you want.

Note also that you can change the style when an element is hovered over by tacking on *:hover*

https://www.w3schools.com/cssref/css_selectors.asp

Option #3: Use CSS

Example:

```
h2 {
    color: blue;
}
```

https://www.w3schools.com/cssref/css_selectors.asp

Option #3: Use CSS

This will apply to elements inside h2 tags.

Example:

```
h2 {
    color: blue;
}
```

https://www.w3schools.com/cssref/css_selectors.asp

Option #3: Use CSS

Example:

```
h2 {
  color: blue;
}
```

Change the font color to blue. You can specify a color by name, by hex, or by other methods, like rgba.

https://www.w3schools.com/cssref/css_selectors.asp

Option #3: Use CSS

This will apply to elements inside h2 tags when they are hovered over.

Example:

```
h2:hover {
    color: blue;
}
```

https://www.w3schools.com/cssref/css_selectors.asp

Option #3: Use CSS

Example:

```
.irs-grid-pol.small {
    height: 0px;
}
```

https://www.w3schools.com/cssref/css_selectors.asp

Option #3: Use CSS

This will apply to elements which are of class irs-grid-pol.small.

Example:

```
.irs-grid-pol.small {
    height: 0px;
}
```

https://www.w3schools.com/cssref/css_selectors.asp

Option #3: Use CSS

Example:

```
.irs-grid-pol.small {
  height: 0px;
}
```

Set the height to
0 pixels.

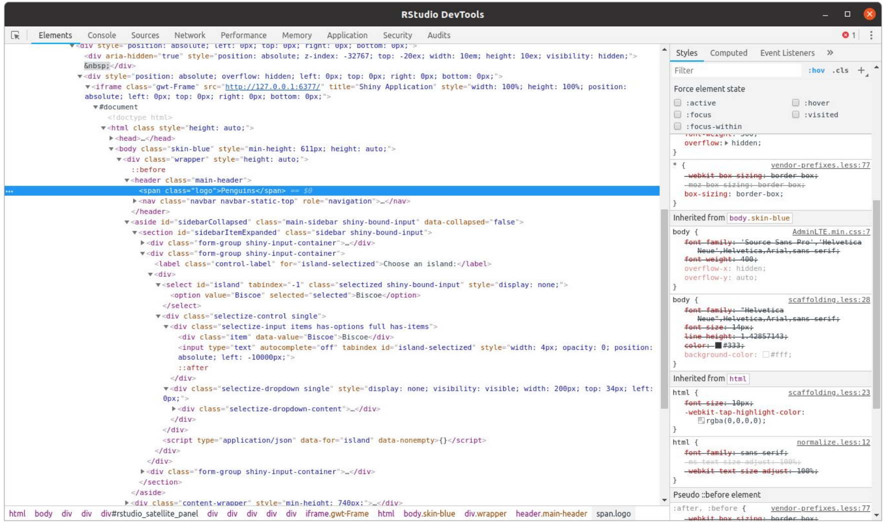
https://www.w3schools.com/cssref/css_selectors.asp

Option #3: Use CSS

The RStudio DevTools are extremely useful when adjusting the look of your Shiny app.

Right-click on an element and choose “Inspect element” to open DevTools.

You can look at the “Styles” tab on the right side to determine what is affecting the appearance of an element and to try out other settings (which can then be added to your .css file).



```
▼<div style="position: absolute; left: 0px; top: 0px; right: 0px; bottom: 0px;">
   <div aria-hidden="true" style="position: absolute; z-index: -32767; top: -20ex; width: 10em; height: 10ex; visibility: hidden;">
   &nbsp;</div>
  ▼<div style="position: absolute; overflow: hidden; left: 0px; top: 0px; right: 0px; bottom: 0px;">
   ▼<iframe class="gwt-Frame" src="http://127.0.0.1:6377/" title="Shiny Application" style="width: 100%; height: 100%; position:
   absolute; left: 0px; top: 0px; right: 0px; bottom: 0px;">
     ▼#document
        <!doctype html>
       ▼<html class style="height: auto;">
         ▶<head>…</head>
         ▼<body class="skin-blue" style="min-height: 611px; height: auto;">
           ▼<div class="wrapper" style="height: auto;">
              ::before
             ▼<header class="main-header">
...             <span class="logo">Penguins</span> == $0
               ▶<nav class="navbar navbar-static-top" role="navigation">…</nav>
              </header>
             ▼<aside id="sidebarCollapsed" class="main-sidebar shiny-bound-input" data-collapsed="false">
               ▼<section id="sidebarItemExpanded" class="sidebar shiny-bound-input">
                 ▶<div class="form-group shiny-input-container">…</div>
                 ▼<div class="form-group shiny-input-container">
                    <label class="control-label" for="island-selectized">Choose an island:</label>
                   ▼<div>
                     ▼<select id="island" tabindex="-1" class="selectized shiny-bound-input" style="display: none;">
                        <option value="Biscoe" selected="selected">Biscoe</option>
                      </select>
                     ▼<div class="selectize-control single">
                       ▼<div class="selectize-input items has-options full has-items">
                          <div class="item" data-value="Biscoe">Biscoe</div>
                          <input type="text" autocomplete="off" tabindex id="island-selectized" style="width: 4px; opacity: 0; position:
                          absolute; left: -10000px;">
                          ::after
                        </div>
                       ▼<div class="selectize-dropdown single" style="display: none; visibility: visible; width: 200px; top: 34px; left:
                       0px;">
                         ▶<div class="selectize-dropdown-content">…</div>
                        </div>
                      </div>
                      <script type="application/json" data-for="island" data-nonempty>{}</script>
                    </div>
                  </div>
                 ▶<div class="form-group shiny-input-container">…</div>
                </section>
              </aside>
             ▶<div class="content-wrapper" style="min-height: 740px;">…</div>
```

```
  font-weight: 300;
  overflow: ▶ hidden;
}
* {                              vendor-prefixes.less:77
  -webkit-box-sizing: border-box;
  -moz-box-sizing: border-box;
  box-sizing: border-box;
}
Inherited from body.skin-blue
body {                                AdminLTE.min.css:7
  font-family: 'Source Sans Pro','Helvetica
     Neue',Helvetica,Arial,sans-serif;
  font-weight: 400;
  overflow-x: hidden;
  overflow-y: auto;
}
body {                               scaffolding.less:28
  font-family: "Helvetica
     Neue",Helvetica,Arial,sans-serif;
  font-size: 14px;
  line-height: 1.42857143;
  color: ■ #333;
  background-color: □ #fff;
}
Inherited from html
html {                               scaffolding.less:23
  font-size: 10px;
  -webkit-tap-highlight-color:
     ▨ rgba(0,0,0,0);
}
html {                                normalize.less:12
  font-family: sans-serif;
  -ms-text-size-adjust: 100%;
  -webkit-text-size-adjust: 100%;
}
Pseudo ::before element
:after, :before {            vendor-prefixes.less:77
  -webkit-box-sizing: border-box;
```

Option #3: Use CSS

CSS can be added directly in your ui.R script using tags$style:

```
tags$style(type = "text/css", ".irs-grid-pol.small
{height: 0px;}"),
```

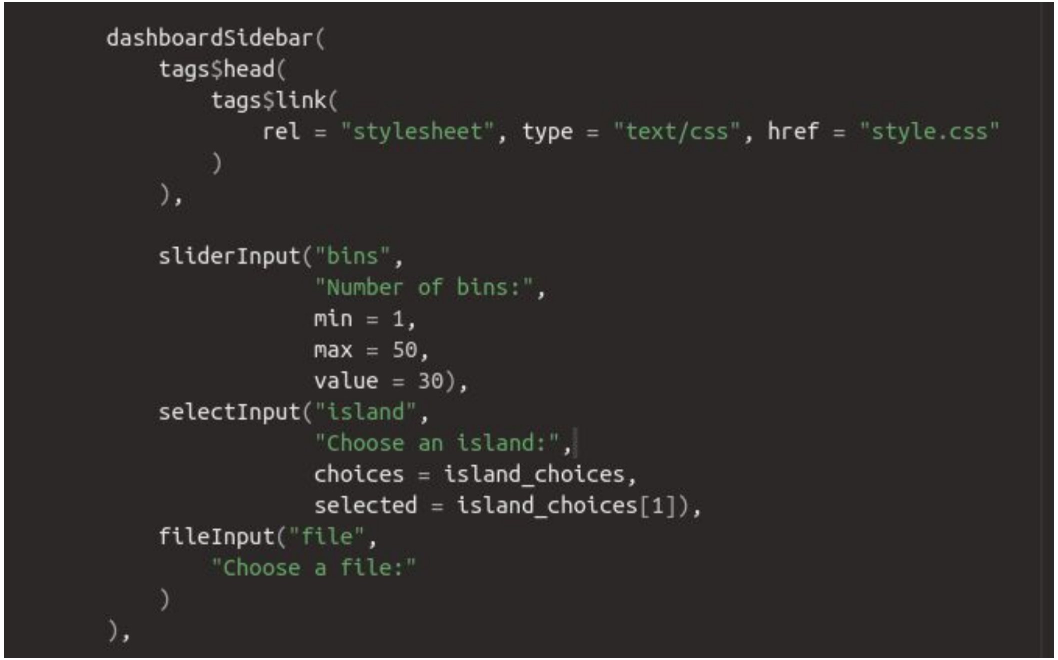
Or by using the style argument:

```
fluidRow(
    style = "padding-top: 20px",
    plotOutput("cylPlot", height = "300px")
)
```

Option #3: Use CSS

Or by linking to a stylesheet contained in a *www* folder.

In general, Shiny will look for images or other static files that you want to include in your app in a folder named *www* in the directory holding your app.



```
dashboardSidebar(
    tags$head(
        tags$link(
            rel = "stylesheet", type = "text/css", href = "style.css"
        )
    ),

    sliderInput("bins",
                "Number of bins:",
                min = 1,
                max = 50,
                value = 30),
    selectInput("island",
                "Choose an island:",
                choices = island_choices,
                selected = island_choices[1]),
    fileInput("file",
        "Choose a file:"
    )
),
```