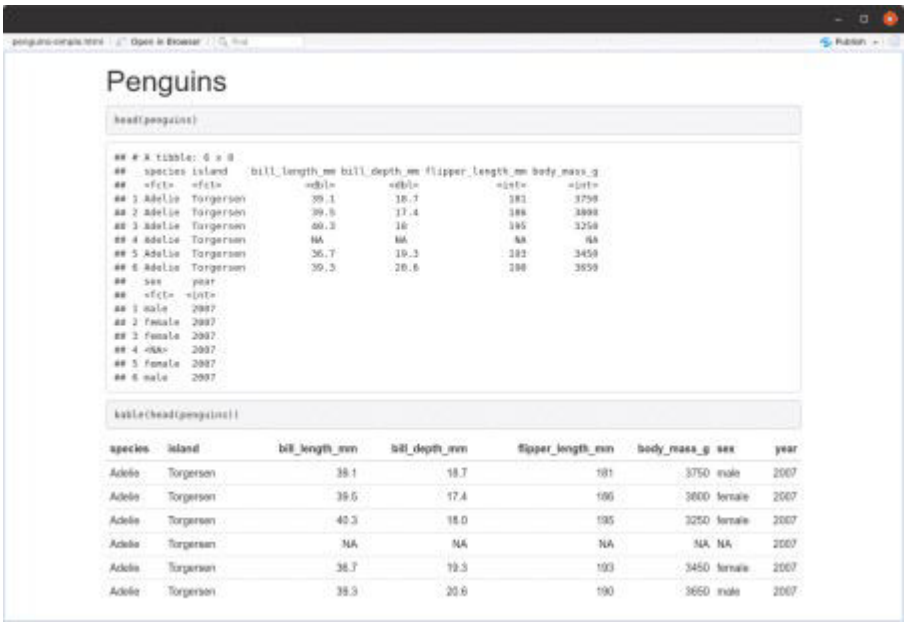
Static Document

Let's start with a `.Rmd` document which renders two different *static* views of the Palmer Archipelago (Antarctica) Penguin Data.

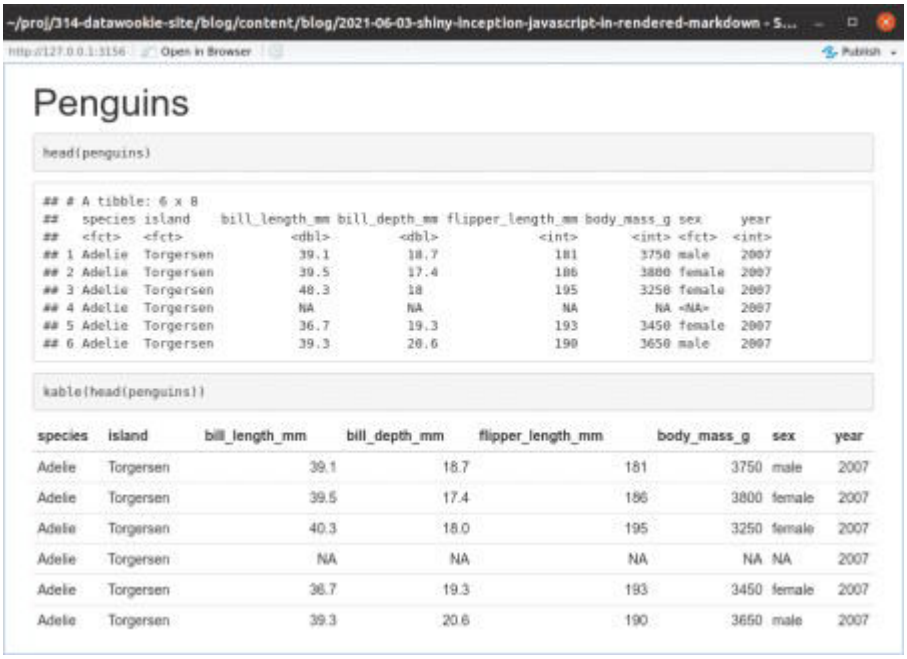
This is what the rendered document looks like:



The following Shiny application embeds this document perfectly using `includeHTML()` to read in the HTML file and `htmlOutput()` to inject it into the UI.

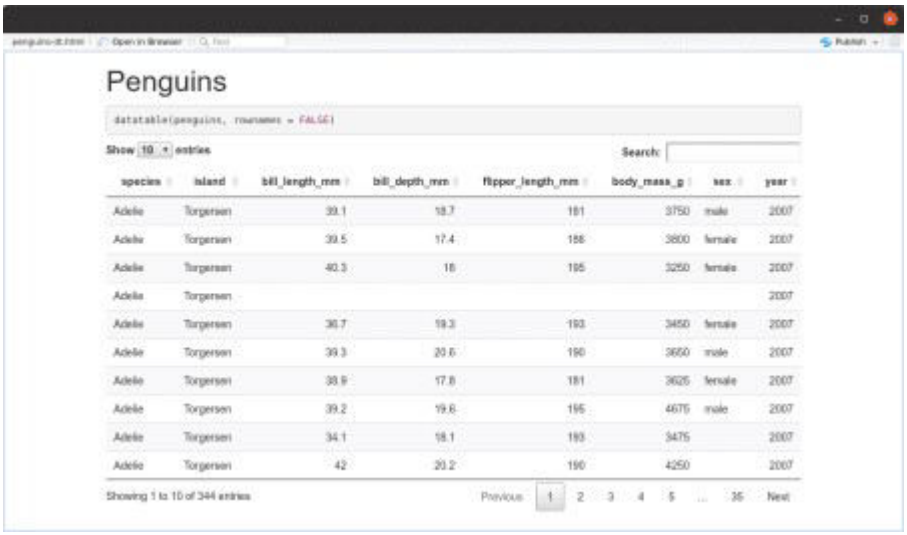
```
library(shiny)
library(rmarkdown)

ui <- fluidPage(
    htmlOutput("document")
)

server <- function(input, output) {
    output$document <- renderUI({
        path_rmd <- "penguins-static.Rmd"
        path_html <- tempfile(fileext = ".html")
        render(
            path_rmd,
            output_file = path_html
        )
        includeHTML(path_html)
    })
}

shinyApp(ui = ui, server = server)
```

And this is what the application looks like. Precisely as expected.



Dynamic Document

Now, let's replace the static document with a `.Rmd` document using the `{DT}` package to create a *dynamic* table. Simply knitting the document gives the output below, a responsive table with a selection of bells and whistles.

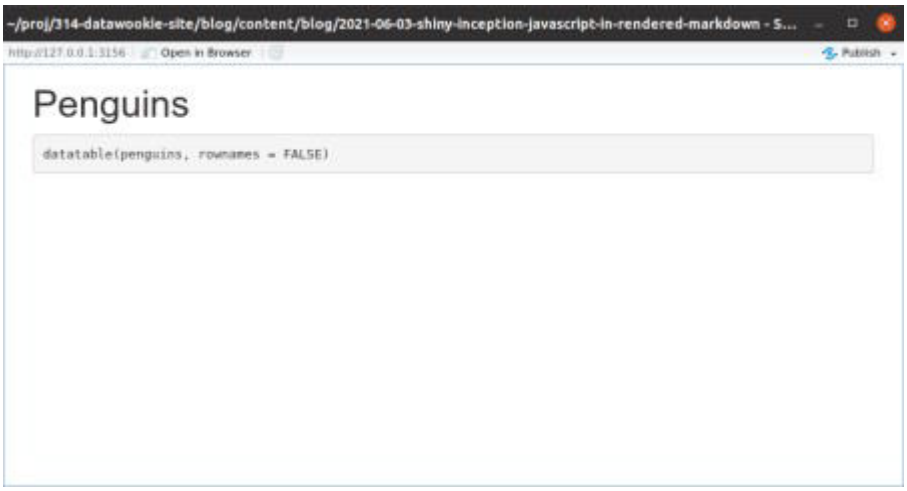


How about simply plugging this into the original Shiny application?

```
library(shiny)
library(rmarkdown)

ui <- fluidPage(
    htmlOutput("document")
)

server <- function(input, output) {
    output$document <- renderUI({
        path_rmd <- "penguins-dt.Rmd"
        path_html <- tempfile(fileext = ".html")
        render(
```

```
            path_rmd,
            output_file = path_html
        )
        includeHTML(path_html)
    })
}

shinyApp(ui = ui, server = server)
```

Let's give that an optimistic whirl.



Doh! The table doesn't appear at all. Why? I'm guessing that the viewer is using JavaScript to run the application but it doesn't extend to running JavaScript in an embedded document.

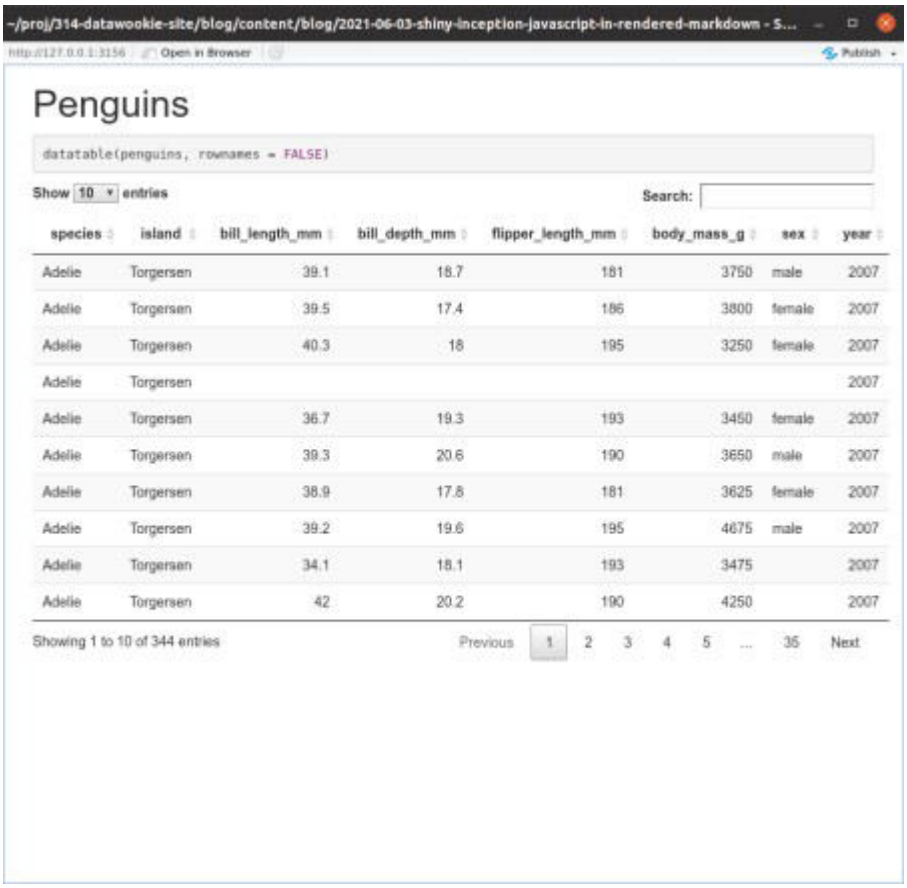
To get this to work I used an `<iframe>`, which in retrospect is the obvious solution for embedding a complete HTML page into the application. Getting it to work though is not entirely trivial.

```
library(shiny)
library(here)
library(rmarkdown)

dir.create("www")

ui <- fluidPage(
    htmlOutput("document")
)

server <- function(input, output) {
    output$document <- renderUI({
        path_rmd <- "penguins-dt.Rmd"
        # Render into www/ folder.
        path_html <- tempfile(fileext = ".html", tmpdir = "www")
        render(
            path_rmd,
            output_file = path_html
        )
        tags$iframe(
            style = "border-width: 0;",
            width = "100%",
```

```
            height = 800,
            # Filename relative to the www/ folder.
            src = basename(path_html)
        )
    })
}

shinyApp(ui = ui, server = server)
```

There's a bit of plumbing required: the document must be rendered into the `www/` folder so that it's visible to the application at runtime.

Here's the result:



🚀 Success!