MTH 459, Quarto Tutorial #1, Spring 2025

Some aspects of these Quarto Tutorials will be review, but it will be good to have some clear and specific examples/templates for you to work with going forward and for presenting the results of your capstone project. Each of these Quarto tutorials will render a different type of document: HTML, Word, PowerPoint, and LaTeX/pdf.

Render your document as an HTML page.

Quarto Document: Mandelbrot and Julia Sets
File Name: **mandelbrot-julia.qmd**

Here is the raw source code you can begin with:

````
---
title: "Visualizing Mandelbrot and Julia Sets"
format: html
---

# Introduction

Fractal geometry offers a fascinating glimpse into complex mathematical systems. Two iconic examples are the **Mandelbrot Set** and **Julia Sets**, which are visualized using iterative mathematical computations. This document explains these concepts and visualizes them in black and white for clarity.

# Mandelbrot Set

The Mandelbrot Set is defined by the equation:\
\[ z\_{n+1} = z_n\^2 + c \] where (z_0 = 0) and (c) is a complex parameter. A point (c) belongs to the Mandelbrot Set if the sequence remains bounded as (n) approaches infinity.

Below is a black-and-white visualization of the Mandelbrot Set.

```{r}
library(ggplot2)

#Mandelbrot calculation

mandelbrot <- function(xlim, ylim, res, iter) { x <- seq(xlim[1], xlim[2], length.out = res)
y <- seq(ylim[1], ylim[2], length.out = res)
grid <- expand.grid(x = x, y = y)
z <- complex(real = grid$x, imaginary = grid$y)
c <- z
count <- matrix(0, nrow = res, ncol = res)

for (i in 1:iter) { z <- z^2 + c
count <- count + (Mod(z) <= 2) }
````
````

```
grid$iterations <- as.vector(count)
grid }

mandelbrot_data <- mandelbrot(c(-2, 1), c(-1.5, 1.5), 500, 50)

##Plot Mandelbrot Set

ggplot(mandelbrot_data, aes(x = x, y = y, fill = iterations)) + geom_raster() +
scale_fill_gradient(low = "white", high = "black", na.value = "white") + theme_minimal() +
labs(title = "Mandelbrot Set", x = "Re(c)", y = "Im(c)")
```

Julia Set

Julia Sets are closely related to the Mandelbrot Set. For a fixed c, the Julia Set shows the behavior of the iterative function:

zn+1=zn2+c

Below is a randomly generated black-and-white Julia Set.

```{r}
julia <- function(c, xlim, ylim, res, iter) {
  x <- seq(xlim[1], xlim[2], length.out = res)
  y <- seq(ylim[1], ylim[2], length.out = res)
  grid <- expand.grid(x = x, y = y)
  z <- complex(real = grid$x, imaginary = grid$y)
  count <- matrix(0, nrow = res, ncol = res)

  for (i in 1:iter) {
    z <- z^2 + c
    count <- count + (Mod(z) <= 2)
  }

  grid$iterations <- as.vector(count)
  grid
}

set.seed(42)  # Ensures reproducibility
julia_data <- julia(complex(real = -0.7, imaginary = 0.27015), c(-1.5, 1.5), c(-1.5, 1.5), 500, 50)

# Plot Julia Set
ggplot(julia_data, aes(x = x, y = y, fill = iterations)) +
  geom_raster() +
  scale_fill_gradient(low = "white", high = "black", na.value = "white") +
  theme_minimal() +
  labs(title = "Julia Set", x = "Re(z)", y = "Im(z)")
```

````
```

# Conclusion

This document demonstrates how mathematical concepts like the Mandelbrot and Julia Sets can be visualized and explained in clear terms. You can use this template to start your own research-focused projects and render them across multiple formats.
````

For your assignment, complete the following tasks:

1. Make sure that the code renders as it should as is.
2. Add an author name (yours) and date information.
3. Expand on the explanation of the Mandelbrot set: find a reference and add a little bit about the history of the set. Include the reference at the bottom of the page in your choice of reference format (it should be complete, and not just a link). Also update the information on the Julia Set.
4. Update the code to change the colorization of the graphs, for example, you could do white and blue, or green to yellow, etc.
5. Change the starting point of the Julia set to a different complex number (try using a random number initialization as long as it's in the range of your previous graph).
6. Create a new conclusion (that does not reference your other assignments in this course, just the topic of the page—perhaps you could mention other fractals that might also be worth exploring?).
7. Render the document as an html page, and then print that page to a pdf.
8. Submit the pdf file to the dropbox in Blackboard.

Hang on to your qmd file since we'll use it as the basis for future quarto tutorials.
````