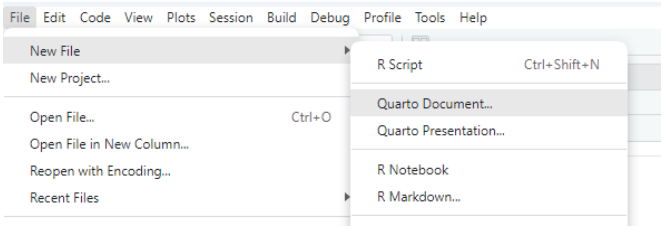
MTH 400, Quarto Lab, Fall 2024 Name _____

Instructions: Follow along with the tutorial portion of the lab. Replicate the code examples in R on your own, along with the demonstration. Then use those examples as a model to answer the questions/perform the tasks that follow. Copy and paste the results of your code to answer questions where directed. Submit your response file and the code used (both for the tutorial and part two). Your code file and your lab response file should each include your name inside.

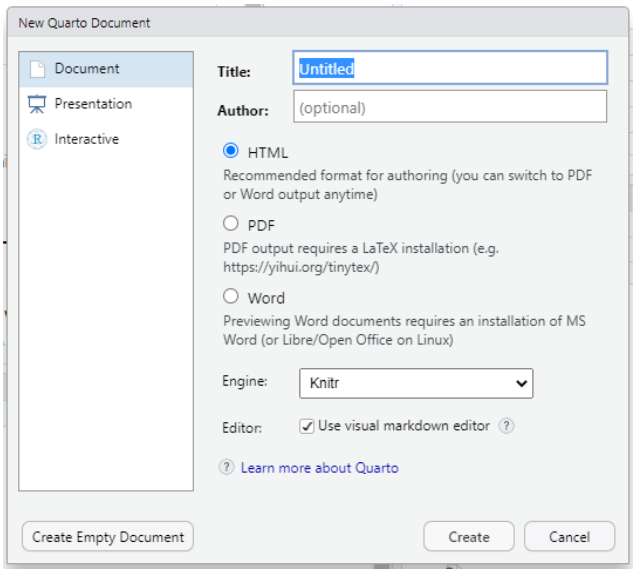
RMarkdown/Quarto

In older versions of R Studio, you could use RMarkdown to create documents such as Word documents, pdfs, presentations or webpages. While you can still use RMarkdown, Quarto is the latest version of this feature. We're going to use Quarto to render some R code and create an HTML webpage with the code and the output. In the updated version of Quarto, you can write the source code yourself or use the visual editor. We'll look at both.

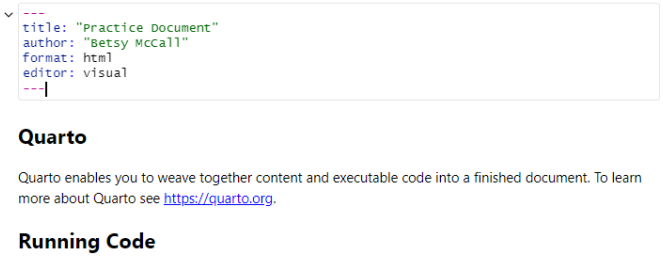
To start creating your Quarto document, open up RStudio. From the File tab select New File and then Quarto Document.



A popup menu appears and here you can name your document and put down your name as the author. You can also specify what type of document you are trying to create. You will have different options if you select Presentation or Interactive, but for now, we'll stick with HTML. You can leave the other options as they are. If you want to work entirely from scratch, you can select Create Empty Document, but if you select Create, you'll get a document with some formatting already in place that will give you something to work from.

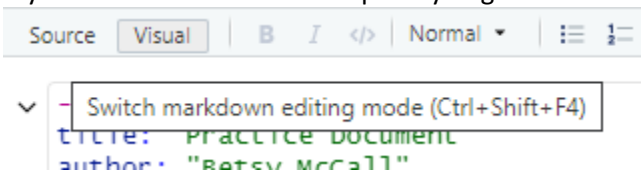


The top of the document that pops up will look something like this:



There is some prefilled content here that we are going to overwrite for our exercise. At the very top you can see that the information from the popup menu is prefilled in with our document title, author name, the format we are converting to and the type of editor we are using. There are also some section headers and text. We can replace these with an introduction to our assignment. Do that now. You can leave the Quarto section alone, but replace "Running Code" with a new section name, and the text that follows with a description. For our example, we are going to compare the normal and student-T distributions to each other. You may want to keep editing this section as we work through the exercise, but for now, fill in a new section title and placeholder text.

If you hover over the Source option you get this message:



We can look behind the scenes at what the Quarto document would look like if we were writing Markdown code ourselves.

```
---
title: "Practice Document"
author: "Betsy McCall"
format: html
editor: visual
---

## Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see <https://quarto.org>.

## Running Code

When you click the **Render** button a document will be generated that includes both content and the output of embedded code. You can embed code like this:

```{r}
1 + 1
```

```

It looks a little like an R code file, but the ## indicate a section heading, and the ```{r} introduces an R code chunk that we can run and output into the document. If you've ever used Markdown in WordPress or Jupyter Notebook, it's similar. The visual editor in RStudio, though, means we don't have to worry about Markdown code too much because we can copy and paste from the document or insert code chunks from the menu.

We are going to replace the first code chunk (that is currently showing 1+1 with the code for plotting points on our curves.

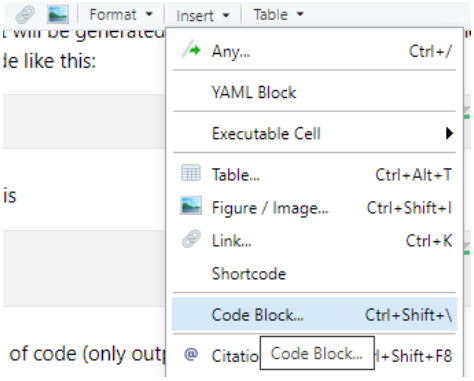
```r
{r}
#| include: true
x<-seq(-4,4,by=0.1)
y<-dnorm(x)
y1<-dt(x,1)
y2<-dt(x,4)
y3<-dt(x,50)
```

Replace the 1+1 with the code shown above.

In the second block of code, we are going to plot the curves using base R graphing functions. Update the text ahead of this code chunk to reflect that, then replace the code with the following:

```r
{r}
#| include: true
plot(x,y,type='l', main=" Standard Normal Distribution compared to \n Student-T for
n=1 (blue),4 (red),50 (green)")
lines(x,y1,col="blue")
lines(x,y2,col="red")
lines(x,y3,col="green")
```

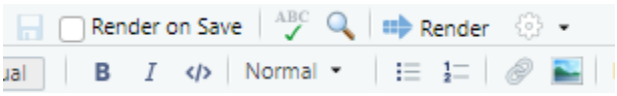
We are going to add one more code block to plot the curves in ggplot. There are several ways to get another code block. You can copy and paste an existing one, or use the Insert option at the top. If you use the insert option, you'll have to specify the language (use lowercase r and it will appear in the list of options). However, copying and pasting is probably easier.



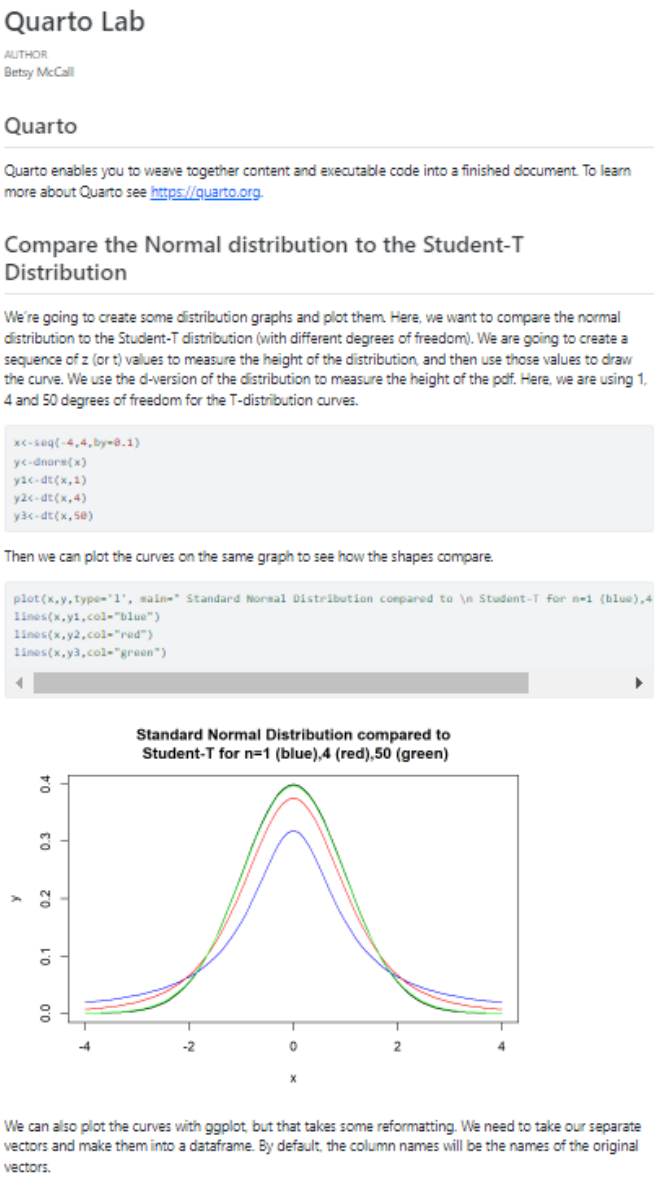
In this third code chunk, we need to create a dataframe from our vectors so that ggplot can work with it. Then we can make the plot. Update the description and add the final chunk of code shown below.

```r
{r}
#| include: true
library(ggplot2)
datadist<-as.data.frame(cbind(x,y,y1,y2,y3))
ggplot(data=datadist,aes(x=x))+geom_line(aes(y=y))+geom_line(aes(y=y1),color="blue"
)+
  geom_line(aes(y=y2),col='red')+geom_line(aes(y=y3),color='green')+
  labs(title="Normal distribution vs. student-t distribution with n=1,n=4, n=50")
```

Now we are ready to render the document. Save your file and then select Render next to the arrow.



This process will take a couple of seconds and then the browser should open up to display your html page. Mine looks like this on the right. I've zoomed out so that you can see more of it on the screen.



Quarto Lab

AUTHOR
Betsy McCall

Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see https://quarto.org.

Compare the Normal distribution to the Student-T Distribution

We're going to create some distribution graphs and plot them. Here, we want to compare the normal distribution to the Student-T distribution (with different degrees of freedom). We are going to create a sequence of z (or t) values to measure the height of the distribution, and then use those values to draw the curve. We use the d-version of the distribution to measure the height of the pdf. Here, we are using 1, 4 and 50 degrees of freedom for the T-distribution curves.

```
x<-seq(-4,4,by=0.1)
y<-dnorm(x)
y1<-dt(x,1)
y2<-dt(x,4)
y3<-dt(x,50)
```

Then we can plot the curves on the same graph to see how the shapes compare.

```
plot(x,y,type='l', main=" Standard Normal Distribution compared to \n Student-T for n=1 (blue),4
lines(x,y1,col="blue")
lines(x,y2,col="red")
lines(x,y3,col="green")
```


We can also plot the curves with ggplot, but that takes some reformatting. We need to take our separate vectors and make them into a dataframe. By default, the column names will be the names of the original vectors.

If you don't want the code to appear in the document, you can change the include option to false in the code chunks. Then it will not display the code or the results on screen. This is useful if you need to do some processing in the background. You can also use an echo command that will suppress the code chunk itself, but will display any output.

```
#| echo: false
```

Try changing these options in your code chunks and then re-render the document to see what happens. There are times when including the code is important, and other times when only the result matters.

When you are happy with the rendered page, right click on the page and select Save As.

Tasks:

1. We are going to add two new sections to this page to compare other distributions. For this problem, display all the code chunks.
   a. Compare a χ^2-distribution with different degrees of freedom and create a graph similar to the one we did with the normal and t-distributions. You will need to change your x-vector to include only positive values. I suggest an initial range of [0,10], but you may need to experiment with the upper bound, depending on the specific degrees of freedom selected, to see all the relevant features. Choose $df = 1$ and three other values to plot and compare on one graph. You can use either ggplot or base R graphing functions to make the plot. Be sure to include a descriptive title. (Reference [3] has a list of common distributions in R.)
   b. Compare binomial distributions with $n = 100$, and $p = 0.05, p = 0.5, p = 0.95$. Plot them. (See our distributions lab for examples. You may need to adjust the range of the graphs to make the peaks easy to see.)

2. Create a new Quarto document. Load in the mtcars dataset and describe the mpg variable in detail. Create several plots such as a qqplot, histogram, boxplot. Calculate some statistical summaries. Display only the output, not the code. Include a summary and description of what you see, and the results of your analysis.

Resources:

1. https://quarto.org/docs/get-started/hello/rstudio.html
2. https://quarto.org/docs/computations/r.html
3. https://www.stat.umn.edu/geyer/old/5101/rlook.html