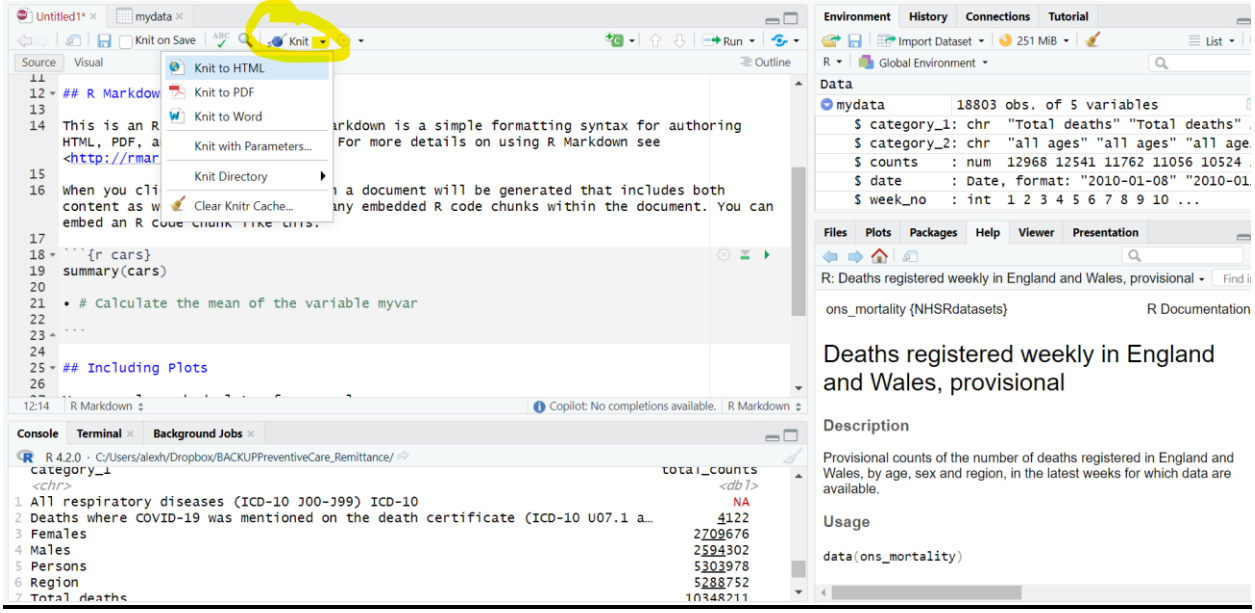
Assignment 1: Introduction to R

Some helpful notes:

- Create your assignment as an .Rmd file and submit only the knitted output (which will include both code and text).
- Your answers should include a code chunk that does the work and (at least one) text chunk that explains the work you've done and presents your answers.
- To run code that you've typed into your .Rmd file while working (so before you knit the entire document), use the green triangles to run a chunk of code. The output will come out just below the code chunk.
- I recommend using the "visual" option in your .Rmd files (highlighted above in green). It makes it much easier to see what's plan text and what's code, as well as to format your text in a way that you might be used to (e.g., like a Word doc).
- Some common compiling errors: Make sure you comment out any lines in your .Rmd file that run the "install.packages()" command *prior to knitting* your file. Also make sure no two code chunks have the same name.

Compiling your file

Once you have written up all the code chunks (in the gray areas of your .Rmd file) and any text that you want to describe those outputs (in the white areas of the file), you are ready to combine them into a single document. To do this, you use the "Knit" button near the top of RStudio:



Most of the time, it will be most convenient to click the arrow next to the ball of yarn and make sure that you are selecting "Knit to Word." Once you click this, your .Rmd file will **run all code in your document** and then produce a Word file with the outputs of your code and the text you wrote to describe it. Note: in order to knit, your R software will start entirely from scratch. This means that **every line of code you needed to complete the analysis must be in your .Rmd file**. Frequently, we switch to the console to play around or test/debug something we're trying to do. If you do this, you have to make sure your code ends up back in the .Rmd file or your file will not knit.

Assignment Problems:

1. **Simple calculations.** Use R as a calculator to compute the following values.
   a. 27*(38-17)
   b. $\sqrt{\frac{\mathbf{436}}{\mathbf{12}}}$

2. Create the following **vectors**.
   a = (5, 10, 15, 20, ..., 160)
   b = (87, 86, 85, ..., 56)

Use pairwise multiplication to define $d = a * b$ (as a vector, not the scalar dot product).

(a) What are the 19th, 20th, and 21st elements of d?
(b) What are all of the elements of d which are less than 2000?
(c) How many elements of d are greater than 6000?

3. **Summary Statistics.** Compute the following statistics of d and report them in a nice, well-formatted table:
   (a) sum
   (b) median
   (c) standard deviation

4. **Matrices.** Use R to create the following two matrices and do the indicated matrix multiplication.

$$\begin{bmatrix} 7 & 9 & 12 \\ 2 & 4 & 13 \end{bmatrix} \times \begin{bmatrix} 1 & 7 & 12 & 19 \\ 2 & 8 & 13 & 20 \\ 3 & 9 & 14 & 21 \end{bmatrix}$$

   What is the resulting matrix?

5. **Loading data.** Install the package NHSRDatasets, which includes sample datasets from the NHS. Load the library and then load the ons_mortality data into your environment.
   a. What are the means and standard deviations for the numeric variables in this dataset?
   b. What are the levels of the categorical variables?
   c. Remember you can use "?ons_mortality" for a data dictionary.

6. **Summarizing data (again).** Use the tidyverse notation to take your NHSRDataset::ons_mortality data to find the average number of deaths by week_no. You may find it helpful to add a step in here that uses the command "filter(category_1 == "Total deaths")" in order to ignore some extraneous information. Report in a nice, well-formatted table.

Note when making tables: you can always format your tables more *ex-post* after you knit to Word. Building tables completely in R can be very time-consuming!