Module 2 – lesson 02: Document Formatting

If you need to, go ahead and log in to your Github account, open RStudio, and open the RStudio project for “Module2\_rmd1”. We’re going to keep working with the R markdown file you just created “module2\_rmd1.Rmd”.

[DEMO – computer]

Previously we discussed the YAML header that contains the title, author, date and output format. Now we’re going to concentrate on the TEXT in the main BODY of your document. You’re going to learn some syntax for how to format parts of your text in different ways.

For now, we’ll ignore the part in between the 3 backticks ```{r} xxxxxxxxxx ``` that may be highlighted in a light grey box. This is a chunk of R code that we’ll be discussing later.

Let’s start with the line that begins with 2 hashtags ## followed by the text R Markdown. The hashtags are markdown SYNTAX used to indicate that the text that follows is a HEADER. The number of hashtags indicates what level the header should be. For example, 2 hashtags ## indicate that the text R Markdown should be formatted and treated as a level 2 header.

Let’s look at an overview of R Markdown syntax. In RStudio, go to HELP/R Markdown Quick Reference or look at the Markdown cheat sheets – this opens a special HELP window at the lower right. Take a few moments to review the information and examples provided.

You can also learn more at the R Markdown website for “Markdown Basics” (<http://rmarkdown.rstudio.com/lesson-8.html>)

And this site links you to more detailed information on the markdown syntax for use by Pandoc (<http://rmarkdown.rstudio.com/authoring_pandoc_markdown.html> )

I mentioned Pandoc in Module 1 as the “universal document converter”. When we process or compile the R Markdown document using the R Markdown package in R, it uses Pandoc to process the document into the final desired formats. So, it’s important to understand how the formatting syntax works so that Pandoc understands it.

Let’s try out some of the basic markdown syntax to get you started formatting your text the way you want it.

Above the ## R Markdown, let’s add a level 1 header and then below that we’ll also add a level 3 header.

# This is a level 1 header

### This is a level 3 header

Click KNIT to HTML and see the result in the viewer.

If you scroll down a little further, you’ll notice that there is a URL shown between the brackets – the less than, and greater than symbols. This is one way to add a web link. The use of these <> symbols is a short hand or abbreviated HTML approach, which is allowed. However, the markdown syntax for adding a link is to put the word or words you want to highlight between square brackets [] followed immediately by the web-address in parentheses (). Let’s add this line to your document.

Click KNIT to HTML and see the results. From the viewer window, test out the weblink – it should open your web browser.

In the next paragraph, the word KNIT has 2 asterisks \*\* at the beginning and at the end of the word. These are used for emphasis. Two asterisks are used for strong emphasis, which results in the work being formatted in BOLD. If we use 1 asterisk, the word is shown in italics. You can also use one or two underscores \_ to emphasize a word.

Let’s add two more lines of text to test this out.

Here is a word in \*\*bold\*\* and a word in \_\_bold\_\_.

Here is a word in \*italics\* and a word in \_italics\_.

Notice that the asterisk or underscore must come immediately before and after the word with no spaces in between for the syntax to work.

Go ahead and click KNIT to HTML to see the results.

Emphasis can also be added using backtick ` marks before and after a word or series of words. This is usually done to highlight text that refers to computer code or a command. The text placed between two backtick marks ` is normally formatted in a non-proportional font and is sometimes highlighted in a different color, like light grey.

Let’s add this line of text and put the name of the R markdown package in between 2 backtick marks.

When we compile our document we are using the `rmarkdown` R package.

You can extend this concept by highlighting a whole block of text to be shown in non-proportional font between 3 backticks on separate lines at the beginning and end of the text block you want to highlight. For example, if we wanted to show an example of 2 lines of R code, we could type in the following.

Here are some example R commands:

```

2+2

mean(c(1,2,3,4,5))

```

Click KNIT to HTML and see the result

NOTE: Putting text between 3 backticks only changes the formatting of that text. This looks very similar to an R code chunk but doesn’t have the {r} after the 1st 3 backticks. We’ll cover R code chunks later.

Let’s also make a bulleted list. We can make non-numbered bullets using asterisks and dashes and the plus symbol. Try the following list. To make a line indented, you must add 4 spaces. To indent twice, add 8 spaces and so on. This is the 4-space rule.

Here is an example of a non-numbered list:

\* Breakfast

- food

+ eggs

+ toast

+ bacon

- drink

+ apple juice

\* Lunch

- taco

\* Dinner

- baked chicken

- broccoli

- rice

We can make this same list numbered, but simply using numbers or letters.

Here is an example of a numbered list:

1. Breakfast

a. food

i. eggs

ii. toast

iii. bacon

b. drink

i. apple juice

2. Lunch

a. taco

3. Dinner

a. baked chicken

b. broccoli

c. rice

Again, KNIT to HTML and view the results

You can also format quotes within R markdown using blockquotes that are highlighted by beginning each line in the quote with a > greater than symbol.

Here are some examples to try.

> This is a block quote. This

> paragraph has two lines.

>

> 1. This is a list inside a block quote.

> 2. Second item.

You can also nest blockquotes - you need a space in between the 2 greater than symbols >'s.

> This is a block quote. This

> paragraph has two lines.

>

> > This text is nested

To indent code within a blockquote, you need to add 5 spaces after the greater than symbol >.

> 2+2

> mean(c(1,2,3,4,5))

KNIT to HTML and see how these come out.

Also try KNIT to WORD and (optionally) KNIT to PDF if you have LaTeX installed to see how each of these various text formatting syntaxes appear in the final documents. You’ll notice that each document format (HTML, DOC and PDF) render each type of text formatting slightly differently – for example, notice how the blockquotes look different in HTML, DOC and PDF.

Now that we’ve made a bunch of changes, let’s remember to stage, commit, and push our changes up to your Github account.

Open Git Bash and change to the directory for your Github repository created for “Module2\_rmd1” – so go to:

C:\RepTemplates\Module2\_rmd1

Once in that directory, type in the following 4 Git commands to check the status of your local files compared to your Github cloud repository; add or stage the modified files; commit your changes; and then push the changes to your Github cloud repository.

git status

git add .

git commit –m “changes to rmd file”

git push