Module 3 – lesson 03: Customizing Other Document Formats

Script

In addition to HTML and WORD documents, you can also customize PDF documents. Creating PDF documents is optional but if you have LaTeX installed, try creating a PDF document and experiment with changing the YAML options to customize your PDF document. You can learn more about customizing PDF documents at <http://rmarkdown.rstudio.com/pdf_document_format.html>

It is worth noting that PDF documents do have the option to create a table of contents like an HTML document. The table of contents option is NOT available for the WORD document format. So, not all options for customizing your documents are interchangeable across all formats. PDF documents can be further customized using external LateX TEX documents like a WORD reference templates or Cascading Style Sheets CSS files for HTML documents. Creating these custom TEX templates is beyond the scope of this course. However, at the end of this course we will see demonstrations of what is possible using existing custom TEX templates. Luckily, other people have already created some very useful custom templates that are easy and ready to use.

We just saw how to customize WORD documents for the DOCX document format. And, to fully customize a DOCX format, it is assumed you have access to and use the Microsoft WORD document creation software. However, in RStudio for R markdown documents, we also have the option of creating ODT (open document text) and RTF (rich text format) document formats which are not dependent on a single software platform.

Let’s create an ODT document. To get to this format, we need to make some changes in the YAML header. To start, let’s create a new HTML document File/New File/R Markdown/HTML. Put in a Title “ODT Document” and click OK. Then change the output option in the YAML header to

output: odt\_document

Click save “odt\_document.Rmd” and KNIT to ODT to see the result in your preview window. You should be able to view this ODT file in a compatible document software like Microsoft WORD or Google Docs. [DEMO Google Docs here]

Save the file under another name “rtf\_document.Rmd”. And change the YAML header one more time to

output: rtf\_document

and change your title. Save and KNIT to RTF to see the resulting RTF document. Similar to the ODT document, you can view RTF documents in Microsoft WORD and Google Docs and other document editing software. [DEMO Google Docs here]

There are also several other formats that have been contributed from the R user community. These formats include custom HTML documents – see <http://rmarkdown.rstudio.com/formats.html>

The html\_pretty template comes from the prettydocs R package.

The next 3 listed: readthedown, html\_docco, and html\_clean templates all come from the rmdformats R package.

Let’s test these out. To gain access to these community formats, we need to first install the R packages they come from.

In RStudio, click on Tools/Install Packages – type in prettydocs from the CRAN repository and click Install. Once the package installs, you should have access to the R markdown templates that were included. Go to File/New File/R Markdown/choose Templates and look for the “Lightweight and Pretty Document (HTML)” template. Click OK.

When this opens, change the title and your author name. This template defaults to a theme called architect with a highlight option set to github. To see what the themes look like, see the link on the R markdown formats webpage. There are 3 themes available “architect” “cayman” and “tactile”

Save your file and KNIT to html\_pretty to see the resulting HTML file. Change the theme to tactile in the YAML header; save the file and KNIT to html\_pretty to see the changes.

Next, let’s look at what is available in the rmdformats R package. In RStudio, click Tools/Install Packages and type in rmdformats from the CRAN repository and click Install.

Once the rmdformats package is installed you will have access to the R markdown templates available from that package. Click on File/New File/R Markdown/ choose Template and look for the “HTML readthedown” template and click OK. Put in a title for your document and add your author name in the YAML header.

This template is very basic and bare bones. We need to add content here to really appreciate the functionality of this template. So, let’s add a few sections to the document. Take a few moments to type in what you’d like to see or cut and paste sections over from other R markdown documents you’ve previously created.

Save the document and KNIT to readthedown. This creates a nice HTML document with a floating table of contents. Let’s also open this HTML file in a browser to fully appreciate the functionality of this new HTML template.

If you wish to access these new HTML documents as webpages served from Github remember to move copies of these HTML files into your /docs directory to view these HTML files online.

Now let’s go ahead and back everything up to your Github account.

Open Git Bash and make sure you are in the correct directory:

C:\RepTemplates\Module3

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 “add custom HTML templated documents

git push

Now go to your Github repository, refresh to see your newly committed files.