Module 1 – lesson 04

Script

This lesson will break down the components needed for reproducible templates.

We will discuss:

* 1. "Data"
* 2. Documentation
* 3. Organization
* 4. Automation
* 5. Dissemination

Data can be thought of as many different things. For example, we often think of data as numbers or even short text in a spreadsheet. But more often than not data is “unstructured”. Unstructured data includes text, which could come from multiple sources including not only reports and documents but books, blogs and websites. Other kinds of data could be:

* images
* video and other media
* interview transcripts
* artwork
* and any other "RAW" materials needed to complete your project.

Regardless of what kind of “data” you have, it should be:

* high quality
* reviewed for completeness
* reviewed for mistakes and errors
* checked for changes or updates

Ideally, your final workflow will allow up these changes and updates to be automatically incorporated into your final product.

No matter what kind of data you have and what type of product you want to produce, you should have

* good documentation of the materials included in your project
* and how the project should be put together
  + the what, why and how and maybe the who (team members and stakeholders)

Often, the main component of your documentation is text. It should be well written. Your documentation should have good organization and workflow. The documentation should be easily accessible and understood by team members at all levels. The documentation could also follow literate programming principles combining the code + text + figures all together in 1 document. Typically, the LAST STEP in the documentation process is the formatting during which final layouts and styles are applied. This course will focus on using "rmarkdown" for final formatting.

You should have an organization structure and guidelines or recommendations that should be followed for your project. This is a good idea even if your project only includes yourself. Because:

* projects grow
* supporting documentation and files - numerous
* relationships change and can grow more complex
* you need recommendations for file organization and naming schemes
* file names should be:
  + - readable by the computer, easy to search, easy to sort (especially by date and author if needed)
    - human readable with logical naming schemes and contain enough info so human knows what is in the file/what the file is for
    - and short enough to be reasonably manageable
* consider user-based access and security (partitioned by "need to know" *[users with editing and write permissions versus users with read-only access]*

There is an R package called “rrrpkg” which was created to “facilitate reproducible research”. Their focus is on creating a Research Compendium. Here are some illustrations of directory structure and file organization they use … <https://github.com/ropensci/rrrpkg>

*{switch to the github site and highlight the examples}*

As much as possible you should try to automate your workflow. Or, at a minimum, you should have a diagram or instructions on your workflow, what documentation is expected and how the components should be assembled.

If possible, you should

* write code/scripts to automate
  + - data raw to processed output
    - creating and removing temporary files
    - creating tables, figures, other components
    - assembling the components into final documents, products
    - rendering documents into multiple desired formats

Once your project is complete, you should disseminate your work. Why should you disseminate your work?

* Store and share your data and code – even if it is only for your future reference
* sometimes expectation/requirement to disseminate your findings by the funding agency or publisher of your work
* you should disseminate your work for increased visibility - when you are listed as the source, you become, by default, THE subject matter expert!
* You should disseminate your work to increase the speed of collaboration - faster advancement of science and/or knowledge in your field
* Dissemination increases good will with the community and public

Here are some ways to disseminate your work using cloud-based solutions. Some Cloud-based "File Storage" options include:

* Dropbox
* Google drive
* Github (better with version control and tracking)

Some examples of data repositories are:

* GenBank
* PDB

In addition to Github, other data and code sharing repositories include:

* Bitbucket
* Dryad
* Figshare
* Zenodo

To whom should you disseminate? First and foremost should be to yourself. Not only is this important for your records and tracking your achievements and completed work projects, but odds are you will reuse something from this project in a future project.

Other key stakeholders in your work may be your organization for which you are creating and disseminating reports internally.

Other ways to disseminate your work may be to:

* Journals - articles, manuscripts
* Books
* Blogs/Websites
* RSS (Rich Site Summary) feeds – like news feeds
* Rpubs – which we will discuss and try out in future lessons in this course
* Other online book platforms such as Gitbook and Bookdown

Next in lesson 05, we will focus on getting you up and running with both R and RStudio.