# Introduction to open science

These materials are meant to introduce you to the principles of open science, effective data management, and data archival with the DataONE data repository. It also provides an overview on the tools we will be using (remote servers, Rstudio, R, Troubleshooting, Exercises) throughout the training. This document is meant to take multiple days to complete depending on your previous knowledge on some of the topics.

If you see anything that needs fixing, submit a issue in the github issues

## Background reading

Read the content on the Arctic Data Center (ADC) webpage to learn more about data submission, preservation, and the history of the ADC. We encourage you to follow the links within these pages to gain a deeper understanding.

- about
- submission
- preservation
- history

## Effective data management

Read Matt Jones et al.'s paper on effective data management to learn how we will be organizing datasets prior to archival.

(Please note that while the tips outlined in this article are best practices, we often do not reformat data files submitted to our repositories unless necessary. It is best to be conservative and not alter other people's data without good reason.)

You may also want to explore the DataONE education resources related to data management.

## Using DataONE

**Data Observation Network for Earth** (DataONE) is a community driven initiative that provides access to data across multiple member repositories, supporting enhanced search and discovery of Earth and environmental data.

Read more about what DataONE is here and about DataONE member node (MN) guidelines here. Please feel free to ask Jeanette any questions you have about DataONE.

We will be applying these concepts in the next chapter.

## Working on a remote server

All of the work that we do at NCEAS is done on our remote server, datateam.nceas.ucsb.edu. If you have never worked on a remote server before, you can think of it like working on a different computer via the internet.

We access RStudio on our server through this link. This is the same as your desktop version of RStudio with one main difference is that files are on the server. Please do all your work here. This way you can share your code with the rest of us.

> ℹ Note
>
> If you R session is frozen and unresponsive check out the guide on how to fix it.

## A note on paths

On the servers, paths to files in your folder always start with `/home/yourusername/...`.

**Note** - if you are a more advanced user, you may use the method you prefer as long as it is evident where your file is from.

When you write scripts, try to avoid writing relative paths (which rely on what you have set your working directory to) as much as possible. Instead, write out the entire path as shown above, so that if another data team member needs to run your script, it is not dependent on a working directory.

## A note on R

This training assumes basic knowledge of R and RStudio. If you want a quick R refresher, walk through Jenny Bryan's excellent materials here.

Throughout this training we will occasionally use the namespace syntax `package_name::function_name()` when writing a function. This syntax denotes which package a function came from. For example `dataone::getSystemMetadata` selects the `getSystemMetadata` function from the `dataone` R package. More detailed information on namespaces can be found here.

## A note on effective troubleshooting in R

We suggest using a combination of **M**inimal **R**eproducible **E**xamples (MRE) and the package `reprex` to create **rep**roducible **ex**amples. This will allow others to better help you if we can run the code on our own computers.

A MRE is stripping down your code to only the parts that cause the bug.

How to generate a reprex:

1. copy the code you want to ask about
2. call `reprex()`
3. fix until everything runs smoothly
4. copy the result to ask your question

When copy and paste code slack message or github issues, use three backticks for code blocks and two backticks for a small piece of code will prevent issues with slack formats quotation.

For more information and examples check out more of Jenny Bryan's slides or watch the video starting at about the 10 min mark.

**Note for EML related MREs:**

- Generating a reprex for these situations (ie. tokens) might be complicated but you can should still follow the MRE principles even if the reprex won't render fully.
- You can include a minimal EML to avoid some `get_package` issues:

```r
me <- list(individualName = list(givenName = "Jeanette", surName = "Clark"))

attributes <- data.frame(attributeName = 'length_1',
                         attributeDefinition = 'def1',
                         measurementScale = 'ratio',
                         domain = 'numericDomain',
                         unit = 'meter',
                         numberType = 'real')

att_list <- set_attributes(attributes)


doc_ex <- list(packageId = "id", system = "system",
            dataset = list(title = "A Mimimal Valid EML Dataset",
                           creator = me,
                           contact = me,
                           dataTable = list(entityName = "data table",
                                         attributeList = att_list)))
```

## A note on Exercises

The rest of the training has a series of exercises. These are meant to take you through the process as someone submitting a dataset from scratch. This is slightly different than the usual workflow but important in understanding the underlying system behind the Arctic Data Center.

Please note that you will be completing everything on the test site for the training. In the future if you are unsure about doing anything with a dataset, the test site is a good place to try things out!

## Exercise 1

This part of the exercise walks you through submitting data through the web form on the development version of our website: test.arcticdata.io

### Part 1

- Download the csv of Table 1 from this paper.
- Reformat the table to meet the guidelines outlined in the journal article on effective data management (this might be easier to do in an interactive environment like Excel).
- Note - we usually don't edit the content in data submissions so don't stress over this part too much

### Part 2

- Go to "test.arcticdata.io" and submit your reformatted file with appropriate metadata that you derive from the text of the paper:
  - list yourself as the first 'Creator' so your test submission can easily be found,
  - for the purposes of this training exercise, not every single author needs to be listed with full contact details, listing the first two authors is fine,
  - directly copying and pasting sections from the paper (abstract, methods, etc.) is also fine,
  - attributes (column names) should be defined, including correct units and missing value codes.
  - submit the dataset