Steps:

* ~~set up some work sessions to go through the code~~
* ~~Naomi runs scripts from start to finish with last version and also with versions running different reactors~~
* ~~Kirsten runs it again with different reactors~~
* start to articulate what the reactor spreadsheet file needs so we track information while running samples and it’s easy to input the reactor spreadsheet into R script with minimal clean up
* take data from the current reactors and run it through the code
* modify the instructions as we go through the code: [D17O\_dataReduction\_Instructions\_20220817](https://docs.google.com/document/d/1O66BjJBM2WDTGlgh96ZVmhq_LNYiGTpqfzicoYU4J-8/edit?usp=sharing)
* workup datasets to generate needs and questions:
  + Naomi works up the Eric Klein data (then Million’s dataset, then the Atacama data…)
  + Kirsten works up the data from Tara and also from the Michigan and the Western US waters
  + performance of USGS waters and see if there are any that we want more data from
  + evaluate the issue of how many times we need to run a sample, what is the best approach

Things to do right now

* keep tally of typical errors or questions that people encounter in the code
  + e.g. the setwd line, solution - can always hard code in the file path
  + modifying the code for just one reactor, just the current one
* document the typical pitfalls that happen with the code and how to navigate them
  + annotate code clearly on how to select reactors to run both in the correction script and also in the reactor formatter script (which is useful to modify when initially importing a reactor to check for formatting issues/errors)
* work on a guide to the outputs and how to use them as you review the the data
  + summaryStd\_basic.csv file to make sure the stdv look reasonable
  + postProcessInput\_backup look at the different segments used for the corrections (1 - basic correction, 2 -
    - prefer user section, forced to update this and fill in the -9999 with a chosen correction
    - Julia always saves the backup because every time you run the script the post-processing gets overwritten
    - another code runs this postProcessInput.csv
    - need to add a new line with the new reactor
  + Rxx\_1d33d34\_SMOW.pdf
    - Raw d33 and d34 values of SMOW, useful for examining spread in values and linear variation through run.
  + Rxx\_2allCorr\_SMOW.pdf
    - Reports SMOW D17O values using different correction types that could be applied for the run.
  + Rxx\_3allCorr\_std.pdf
    - plots all standard D17O data with different correction scenarios
  + Rxx\_4box\_WaterStd.pdf
  + Rxx\_4box\_Water.pdf:
    - box plot of replicate analysis for each sample type, grouped by project. A way to evaluate performance of analyses of a project. Can also check the sample names and project designation.
* clarify best steps for running the code and then how to use it - update and revise [D17O\_dataReduction\_Instructions\_20220817](https://docs.google.com/document/d/1O66BjJBM2WDTGlgh96ZVmhq_LNYiGTpqfzicoYU4J-8/edit?usp=sharing)
* run the whole code, look at the summaryStd\_basic file to make sure the stdv values look reasonable for the reactor of interest
* Clarify rules for PostProcessInput.csv (i.e., when to indicate 0 vs 1 vs 2 vs 3 and how to handle more than one segment)
* clarify system for updating reactor spreadsheet if you see an error in how something was classified (e.g. in R21, some USGS46 analyses are classified as Contract Waters but should be included in the water standard category).
* create a wishlist for what the different outputs have and the documentation that we want to see in the code
  + create a stitched together PDF
  + put the summary stats (n, avg, stdv) for each group on the outputs with the box plots
  + In R23\_2allCorr\_SMOW.pdf the legend didn’t print. Not sure why and how to interpret it.

Things to do in the near term

* consider revisiting how we name reactor files (can we use numbers, instead of spelling out the reactor #?)
* need system for updating a reactor file if find an error in it and where to document this. Document this in the Reactor Summary?
* improve summaryStd\_basic output file so the names of the standards are included in a column of this output
* need a really clear steps for uploading recent versions of code and output to the shared drive and who does this

Things to do in future

* make system for running code on the lab computer used for looking at the reactor spreadsheets in real time etc.
* need protocol for how to name and archive the old versions of the scripts
  + when is it appropriate to save a new code and upload it to the drive? only if you made key modifications that will be useful for others and if they are clearly documented in the notes section
* add a narrative for the correction approach for each reactor (what was done)
  + started - see [0READ ME | ReactorSummaries](https://docs.google.com/document/d/1NjxtUvnbxUCM0xh3ypRkkLhNQEDc2pk0pVtsAx5CY_4/edit?usp=sharing) and folder [0003\_ReactorSummaries](https://drive.google.com/drive/folders/1iqhcXNbIZhQwr1zMw05Ss8abqiRl5v3i?usp=share_link)
  + include a guide to the decisions made for the different post processing choices and the numbers used in the postProcessInput.csv file
* get the code up on GitHub, that helps with version control
* if stay using Google Docs structure, make sure we have a clear protocol for how it’s documented

Notes from meeting with Julia about Crunching data:

After running correction R script, you will get a line that says "Reminder: select preferred correction for Reactor 23 in postProcessInput.csv" (this will be in black writing - not red or blue), what to do next:

* Open 0002\_LabFileCorrectedOutput Folder
  + Open the latest post process input backup csv file with the most recent date (ex: postProcessInput\_backup\_20220906) and then select and copy all rows and columns starting at prefer.user into the postProcessInput csv file and edit the last reactor that you just reduced data for to reflect the correction type that you would like to apply in the columns prefer.auto, prefer.auto.seg1, prefer.auto.seg2
* To determine what type of correction to put for the postProcessInput.csv document for the new reactor, you can open and look at all of the pdfs that get made from running the correction R script (one that is really useful is R\_\_3allCorr\_sdt.pdf.)

The sorting R script makes the cor.data.all.csv AND stdsummary\_box.pdf in the output folder and these should be looked at after running the script successfully.

When uploading stuff back into the google drive:

* Sort by date modified and upload:
  + 0002\_LabFileCorrectedOutput
    - cor.data.all.csv
    - stdSummary\_box.pdf
    - IPL\_17O\_Sorting\_V02.R
    - postProcessInput.csv
    - All of the new reactor cvs files (don’t have to upload previous ones that have already been uploaded)
  + 0001\_ReductionCode
    - The updated IPL\_17O\_correction\_V09.2.R script
  + 0000\_LabFileFormatting
    - 001\_Reactor Spreadsheet Formatted then Reactor23Final.csv (or reactor that you are currently doing)

Make sure to write the date and that you updated the R script to include the new reactor and reactor numbers into the script before uploading the R script back into the drive with the new reactor in it.

**9/29/22:**

Notes and thoughts about formatting a reactor for data corrections:

* Make sure the type of sample you run (e.g. water standards, waters, carbonate standards, phosphates, ect.) in column C, titled “Type 1” matches with the project title or name of the samples that are being run, in column D, tilted “Type 2”.
  + This project name should be listed under the type of sample being run in the “Data Sorting” tab of the Reactor spreadsheet.
    - You know this was added correctly when you get a drop down option on the “All data” tab when choosing the type of sample you ran.
      * For exam, if I ran a water standard and it was a SMOW, I should be able to select “Waterstd” from the Type 1 column and then after selecting that I should get the option to select “SMOW” from the drop down selections in the column titled Type 2.
* In the name column make sure the extra 1 in the sample name is deleted.
  + For exam, after initially reducing the run and putting the data line into the Reactor spreadsheet, you may get something that looks like this

“Data\_2165 IPL-17O-4319 SLAP2-B8-R24-8 1”

This extra 1 at the end is not a part of the sample name and therefore should be deleted.

* Make sure columns AK-AN are filled in for every sample. And if a sample is flagged in column AN titled “flag.analysis” make sure to write the reason for why this sample is getting flagged in the comments column of the reactor spreadsheet (column AJ).

Notes for when trying to run the IPL\_17O\_correction\_V09.2.r Script:

* When trying to reduce reactor 24 (current reactor), I had to go into the “ReactorSpreadsheetFormatter\_func\_v2.R” script and change line 58 from reactor 23 to reactor 24 so that instead of reading

“for (k in 23) { # use this line to pick one reactor” it reads

“for (k in 24) { # use this line to pick one reactor” which therefore allows reactor 24 to get pulled and formatted.

* Make sure to add the new reactor number as a new row in the “postProcessInput.csv” or else the R script also won’t be able to run correctly.

Questions for Naomi: