Procedures and R Scripts for QCing, Formatting and Deriving Summary Outputs for Continuous Temperature and Hydrologic Data (for Advanced R Users)

National Water Quality Monitoring Conference

Erik Leppo, Jen Stamp, Tetra Tech, Inc.
Britta Bierwagen, David Gibbs, EPA/ORD

The views expressed in this presentation are those of the author and they do not necessarily reflect the views or policies of the U.S. Environmental Protection Agency or other collaborating agencies

Office of Research and Development
Air, Climate and Energy Program, National Center for Environmental Assessment

May 5, 2016

Agenda

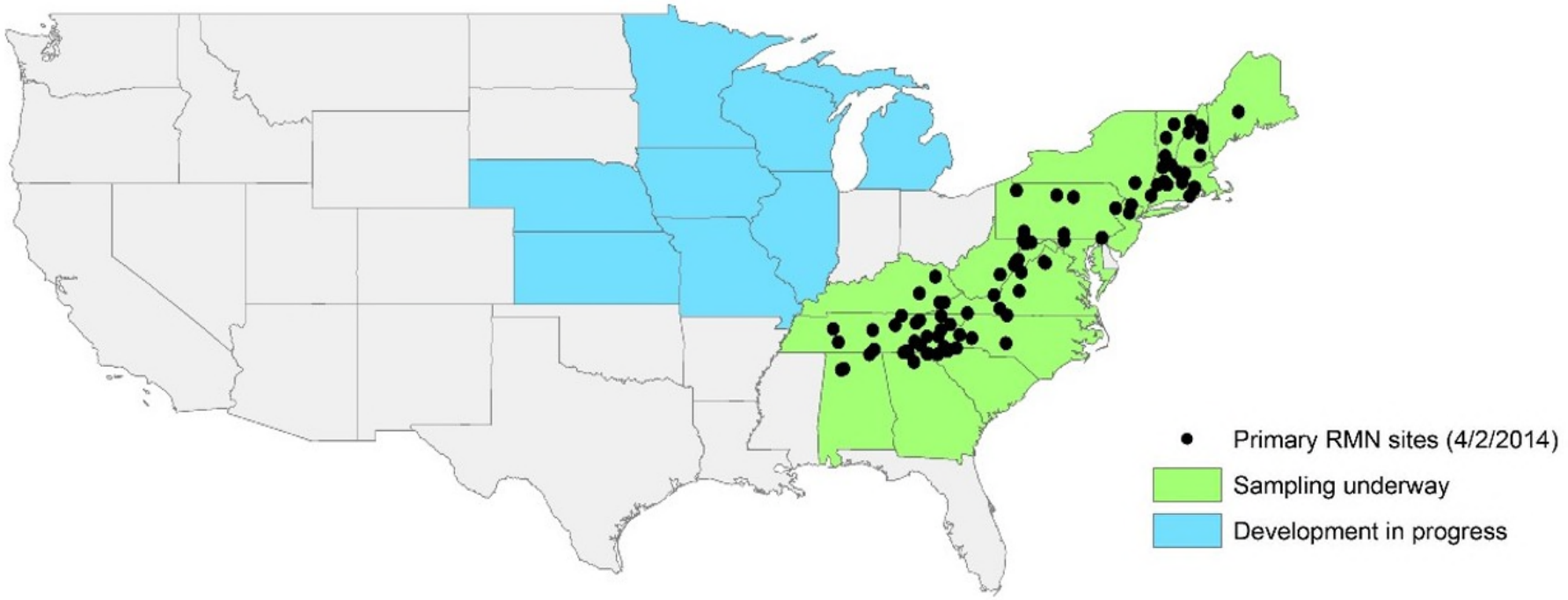
4:00-4:10	Intro/background
4:10-4:50	Test dataset #1 - Non-vented pressure transducer (ECO66G12_AW files) • Modify (reformat, rename) 'raw' files (10 minutes) • QC script (15 minutes) • Aggregate (5 minutes) • Stats scripts (10 minutes)
4:50-5:10	Test dataset #2 – Standalone temperature sensors + gage (WSR01 files) • Run the 'raw' files (already formatted) through the scripts (10 minutes) • Add in gage data • GetGage script (5 minutes)

5:10-5:20	Potential places to make modifications
	• UserDefinedValues
	  • Thresholds
	  • Including or excluding flagged values
	  • Adding more variables (e.g., conductivity, other data from gages)
	• Rmarkdown (rmd) files
	  • Reports (QC, stats, aggregate)

5:20-5:30	Feedback - any additional features and functionality you'd like to see? Other improvements? Question and answer

Why did we develop these scripts?

Working with **EPA Regional offices, states, tribes** and other biomonitoring programs in the **Northeast, Mid-Atlantic, Southeast & Midwest** on **Regional Monitoring Networks (RMNs)**



Sampling efforts began in the Northeast in 2012, followed by the Southeast in 2013, the Mid-Atlantic in 2014, and parts of the Midwest in 2016

Regional Monitoring Networks (RMNs)

Objective: detect potentially small, **climate-related trends** at a **regional scale**, in a **decision-relevant timeframe**, in the context of routine biomonitoring

Sampling goals:

- At least **30 sites with similar environmental and biological characteristics in each region**
  - Average 5 sites per state (range:1-15)
  - Design informed by power analyses on Northeast dataset (2012)
- Sample on an **annual basis** for **10 or more years**, using as **consistent** and **comparable** methods as possible (QAPP)
- Initial focus is on **freshwater wadeable streams; could expand to inland lakes and wetlands**

Data Collection Targets

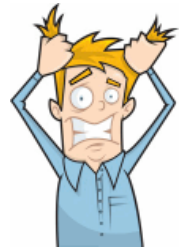
- **Biological**: macroinvertebrates.
  - Optional: fish and periphyton
- **Temperature**: year-round water and air temperature
- **Hydrologic**: year-round water level data (from USGS gages or pressure transducers), ideally converted to discharge.
- **Habitat**
- **Water chemistry**

RMN partners have varying levels of experience with continuous sensors

Data Management

Ok great! now we are collecting the desired data.…

But what are we going to do with all these data?



Challenges!

Lots of different data sources –

Data being collected with **different types of equipment**, sometimes at **different recording intervals**, sometimes in **different formats**

- **Standalone temperature** sensors (e.g. water and air temperature come in separate files)
- **Non-vented pressure transducers** (e.g., temperature and water level data together in one output; sometimes air temperature in a separate file)
- **Vented pressure transducers**
- **USGS discharge data** (needs to be pulled from USGS website and combined with temperature data)

Challenges!

Where to put the data?

There is not a single existing system that can accommodate all of the types of data being collecting at RMN sites.

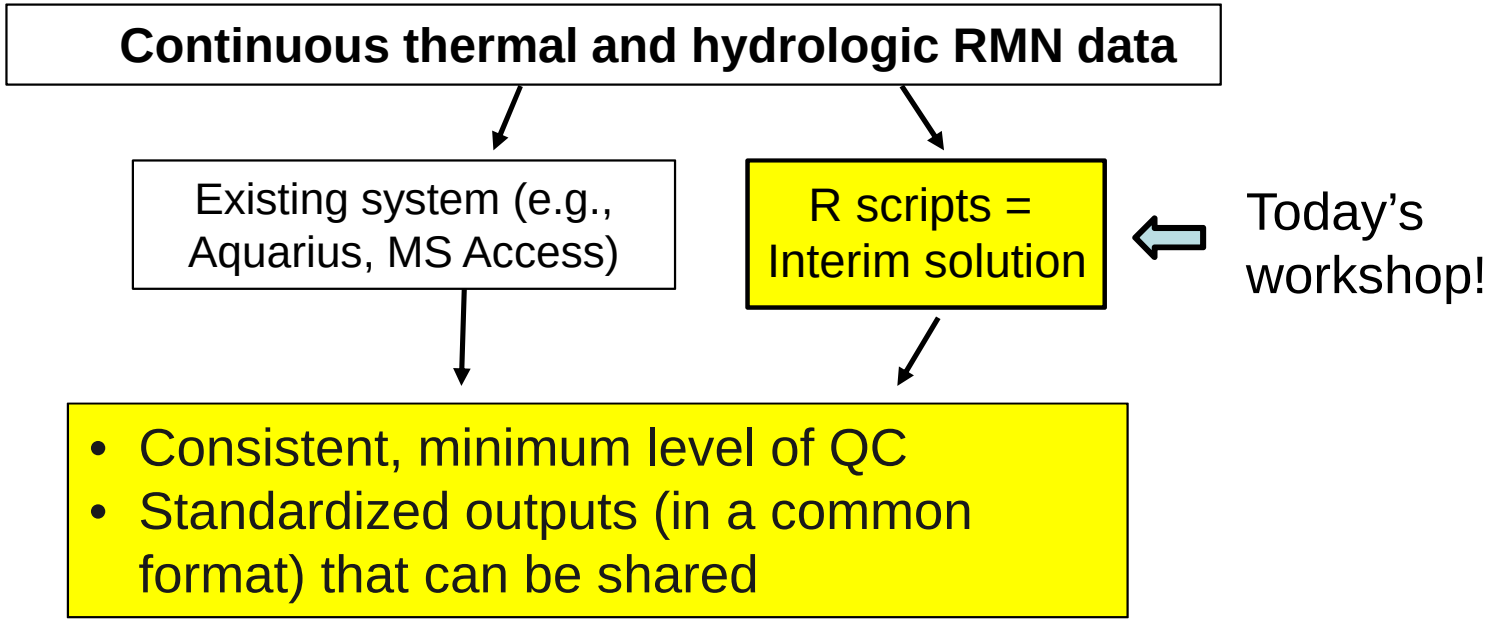
For continuous data, there is an ongoing multi-agency effort (led by Dwane Young, EPA) to address this but full implementation is a few years off…

Until a permanent data management system is in place, the individual organizations will be the custodians and owners of the continuous thermal and hydrologic monitoring data.

- Some people already have a system in place (e.g., Aquarius)
- Some do not

Overall Goals of the R scripts

- Ensure that a certain **(minimum) level of QC** is performed on the continuous thermal and hydrologic RMN data
- Produce **standardized sets of outputs** (summary statistics, plots) that can be shared



Why R?

Our preliminary exploration of performing QC checks and summary statistic calculations in Excel and Access proved too burdensome. Some of the **biggest hurdles** were –

- Excel and Access getting **bogged down by the large data files**, or maxing out capacity
- **Formatting issues** (e.g., inconsistencies across files)
- **Difficulties with combining the different data files**

We did our best **to keep procedures and scripts as simple and generic as possible**, and set them up in a way that **people with limited R experience can understand and run**.

We also tried to utilize existing scripts and resources whenever possible…

Disclaimers

- These R scripts should be regarded as a **proof of concept**. We're learning as we go and won't 'wow' you with these scripts.
- This is not necessarily the 'best' solution; rather this is the **best we could come up with at this time**, with the information we have.
- This was done with **limited time and resources**
  - We won't be able to do everything we'd like to do in this phase of work. Please give us feedback on what additional features you'd like us to add in. We're keeping a 'wish list' and hope to accomplish more through additional phases of work.
- These are **minimum QC standards**. Some organizations may choose to do more.

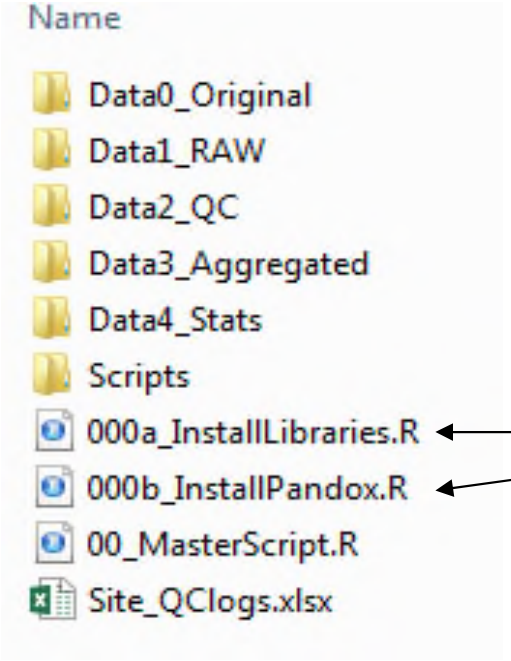
EPA
United States
Environmental Protection
Agency

Ready to go?

Install the data packages

In order for the R scripts to work, you'll need to run the **Install** scripts (**000a_InstallLibraries** & **000b_InstallPandox**).

Unzip the 'Rscripts_NWQMC_20150513' file



Run these 2 scripts
(first run 000a, then 000b)

Upgrading R to a New Version?

United States Environmental Protection Agency

Libraries don't transfer when installing a new version of R.

Here is a 'painless' way to update your libraries -

- After installing the new version, run this in the **old version** -

```
setwd("C:/Temp/")
packages <- installed.packages()[,"Package"]
save(packages, file="Rpackages")
```

- Then run this in the **new version** -

```
setwd("C:/Temp/")
load("Rpackages")
for (p in setdiff(packages, installed.packages()[,"Package"]))
install.packages(p)
```

You should now have updated versions of each library in your new version of R.

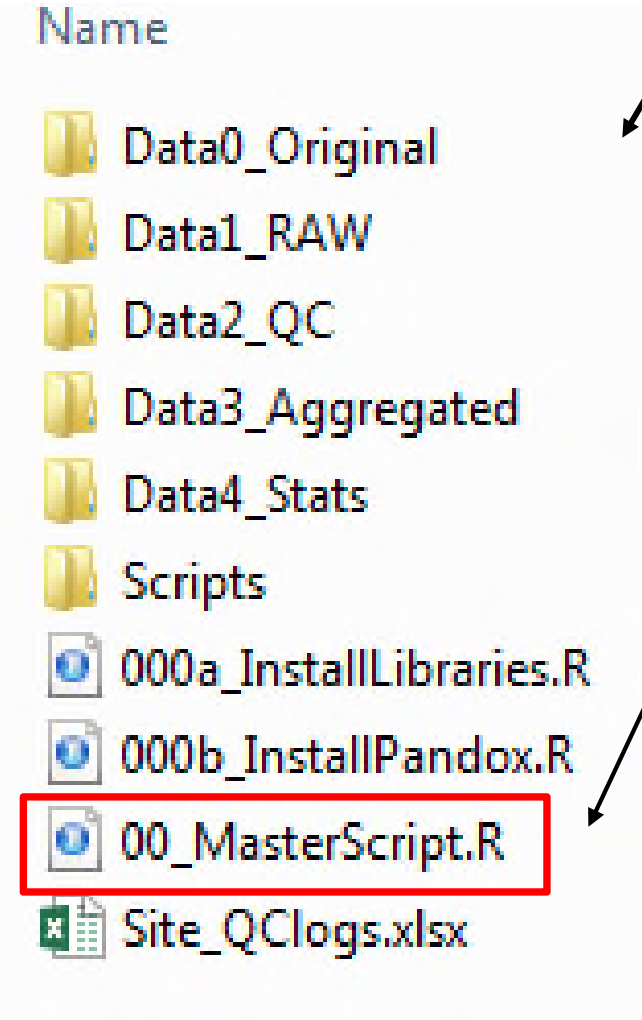
http://stackoverflow.com/questions/1401904/painless-way-to-install-a-new-version-of-r-on-windows

EPA
United States
Environmental Protection
Agency

Overview of the R scripts

R scripts and directory

Unzip the ‘Rscripts_NWQMC_20150422’ file –it should look like this…



You’ll be doing most of your work with the **MasterScript**.

The MasterScript needs to be located in the same directory where the data files are located.

The MasterScript reads through the files and pulls out data based on file names (siteID, dates, data type).

The MasterScript produces outputs that go into certain folders (so the folders have to be named a certain way)

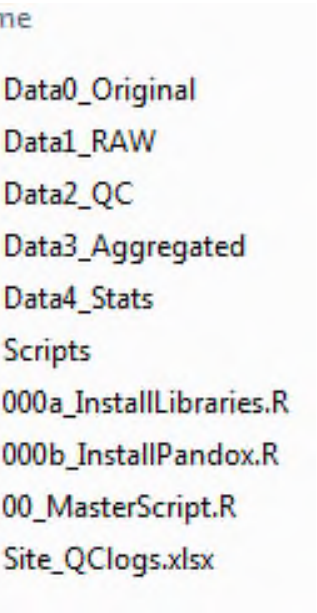
The MasterScript…

Performs 4 different operations (one at a time).
You specify which operation you want to run and what files you want to run it on.

Operation	Description	Import Directory*	Export Directory**	Notes
GetGageData	Retrieves gage data	USGS NWIS website	Data1_Raw	Optional (if you don't need to bring in USGS gage data, you can skip this step and go right to the QCRaw operation)
QCRaw	Performs QC on Raw data	Data1_Raw	Data2_QC	You need to run this portion of the R script before you can run the Aggregate script
Aggregate	Aggregates QC'd data	Data2_QC	Data3_Aggregated	You need to run this portion of the R script before you can run the SummaryStats script
SummaryStats	Generates summary stats on Aggregated data.	Data3_Aggregated	Data4_Stats	

*The R script looks at files in these folders when the script is run
**Output files will appear in these folders after the script runs

Folder structure



- **0_Original**
  - Master files. Do not change these! If you make a mistake and need to start over, you'll need these (unaltered) files.
- **1_RAW**
  - Files are copied from the Original folder and pasted here. These are your working files. Files need to be edited so that they have a standardized format and file naming scheme, and are .csv.
- **2_QC**
  - When the QC portion of the MasterScript is run, the QC data outputs will go here (2 outputs: Word & .csv).
- **3_Aggregate**
  - When the Aggregate portion of the MasterScript is run, the outputs go here (2 outputs: Word & .csv).
- **4_Stats**
  - When the SummaryStats portion of the MasterScript is run, the outputs go here (3 outputs: 2 .csv, 1 PDF)
- **Scripts**
  - R files that you will rarely need to work with, with the possible exception of the fun.UserDefinedValues.R file

There are only 5 places in the MasterScript where you'll need to make entries…

1) Select the **operation**
   - GetGageData, QCRaw, Aggregate or SummaryStats
2) Enter the **SiteID**
3) Enter the **data/sensor type**
   - Air, water, air/water, gage, air/water/gage, air/gage, water/gage
4) Enter a **start date**
5) Enter an **end date**

EPA
United States
Environmental Protection
Agency

Game plan

Go through 2 types of datasets

United States Environmental Protection Agency

Test Dataset #1

Non-vented pressure transducers

- Water temperature
- Air temperature
- Water level
- Barometric (air) pressure
- Water pressure

2 Files –

- Same site, different date ranges

Test Dataset #2

Standalone temperature sensors

- Water
- Air

Gage

3 Files –

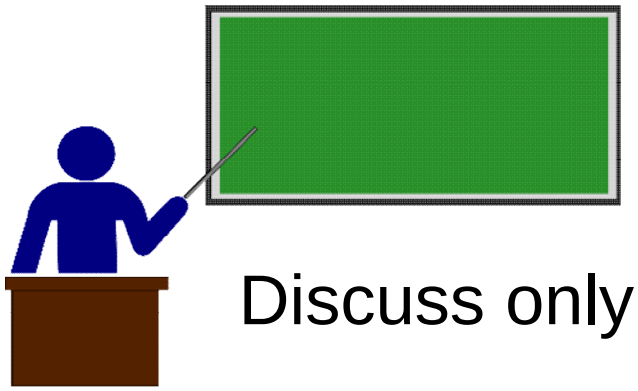
- Same site, parameters in separate files, same date range

Work flow

1. Preparing your data
   - Moving it into the correct folders
   - Organizing, formatting and naming the files
2. Running the QC portion of the MasterScript
   - Checking the data, correcting if needed
3. Running the Aggregate portion of the MasterScript
4. Running the SumStats portion of the MasterScript

Note: we QC one file at a time.
Why not combine them first, then do the QC?
We feel it's best to check files before manipulating and combining them. Also it is easier to QC smaller 'chunks' of data.

Two types of slides



Discuss only

When you see this symbol on a slide, please listen and watch.

You do it!

When you see this symbol on a slide, please perform the task.

EPA
United States
Environmental Protection
Agency

Questions or problems running the R script?

Raise your hand and we'll send someone over to try and help you

EPA
United States
Environmental Protection
Agency

Test Dataset # 1

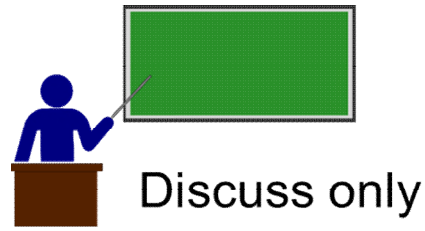
Test Dataset #1

Non-vented pressure transducer data

- Water temperature
- Air temperature
- Water level
- Barometric (air) pressure
- Water pressure

2 Files –

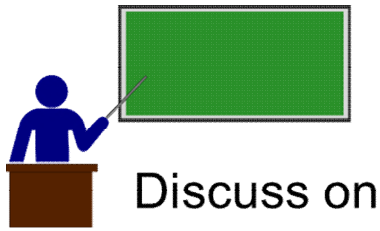
- Same site (ECO66G12)
- Different date ranges
  - April 26, 2013 – July 25, 2013
  - July 25, 2013 – October 15, 2013



Work flow for Test Dataset #1

1. **Prepare your data** (demonstrate only*)
   - Move it into the correct folders
   - Organize, format and name the files
2. **Run the QC** portion of the MasterScript (on both files)
   - Check the data, correct if needed
3. **Run the Aggregate** portion of the MasterScript to merge the QC'd data into one file (for full date range)
4. **Run the SumStats** portion of the MasterScript to generate the summary statistics

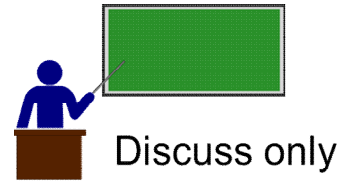
*We prepared the data files in advance in order to maximize the amount of time we can spend running the actual scripts



EPA
United States
Environmental Protection
Agency

Preparing your data

Preparing your data



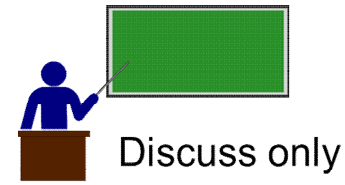
Why is this important?

In order for the R script to work, the files need to be named and formatted a certain way, and located in the appropriate folders.

The MasterScript reads through the files in the appropriate folders and pulls out the desired data based on file names (siteID, dates, data type).

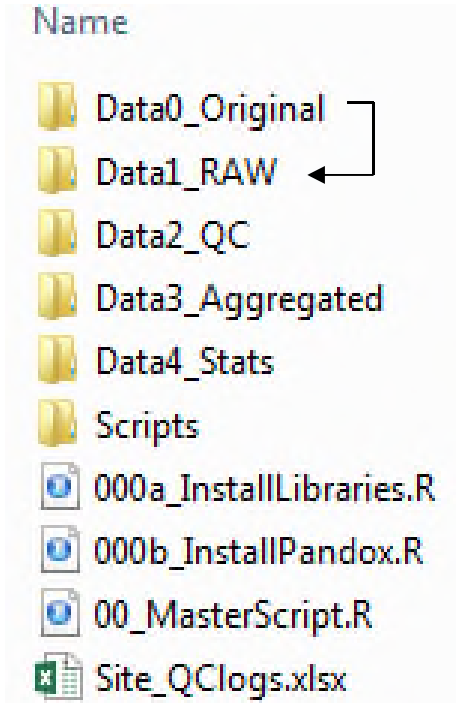
This allows for multiple files to be kept in the same directory.

Preparing your data



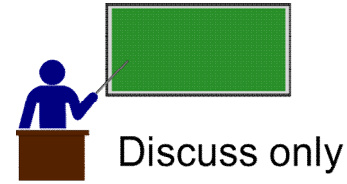
General work flow

1. Put the ‘original’ (unaltered) continuous sensor files that you want to run the R scripts on in the Data0_Original folder.
2. Copy and paste these ‘original’ files into the Data1_RAW folder.
3. Prepare the files in the RAW folder for the R scripts
    - Required & optional fields; naming scheme
    - Check for rows with no data
    - Name files
    - Save as .csv



For today’s workshop, to help speed things along, we’ve already done these steps on the test files. However, we will talk you through these steps so that you’ll know how to prepare your own data.

1. Move data into the Data0_original folder



These are your ‘original’ continuous sensor files*

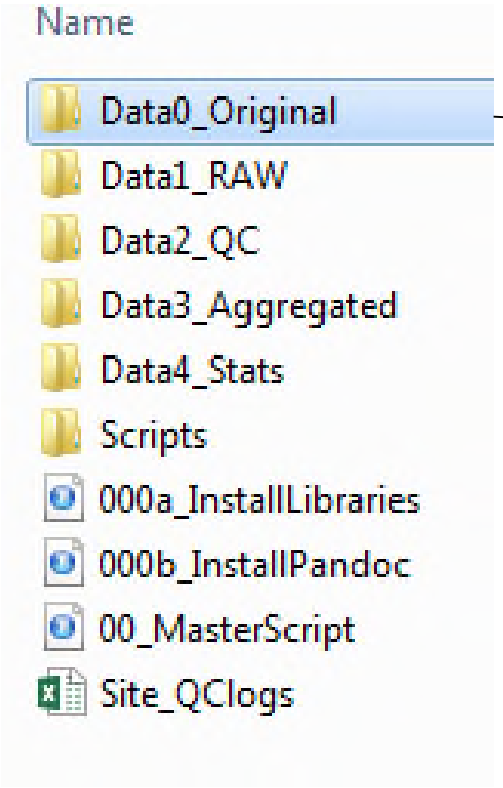
Once the files are in this folder, ***do not change them!*** If you make a mistake, you might need to go back to these (unaltered) files and start over.

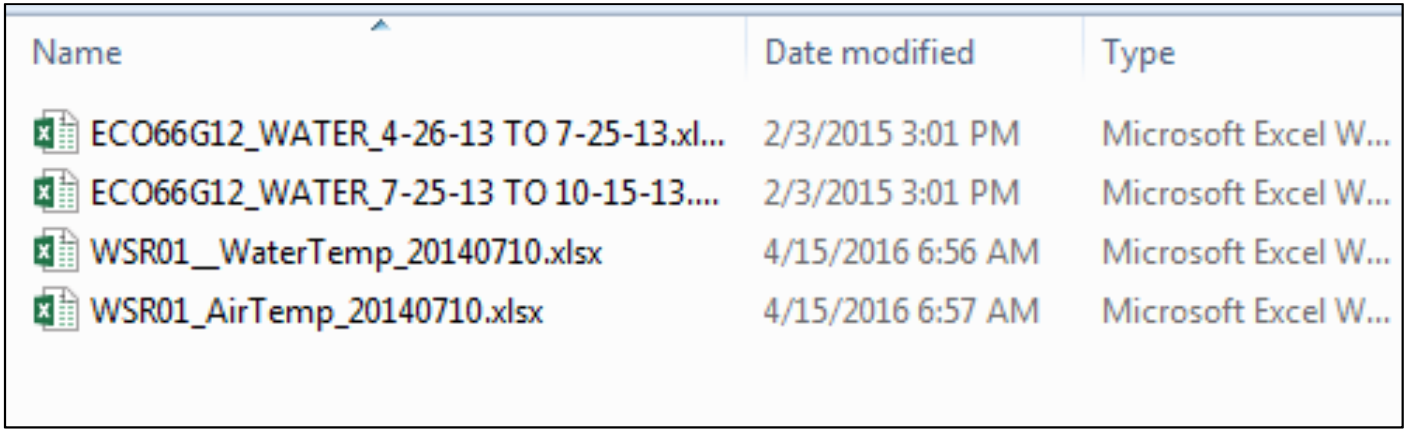
The files in the **Data0_Original folder** are *not* used by the R scripts, so the names and formats do not have to be standardized (e.g., it’s ok if the “Original” files are in Excel).

*in some cases, you may need to run the data files through the sensor software (e.g., HOBOWare) before putting them into the Data0_Original folder.

Let's take a look at the test files that are in the Data0_Original folder…

Open the Data0_Original folder





You should see 4 Excel files –

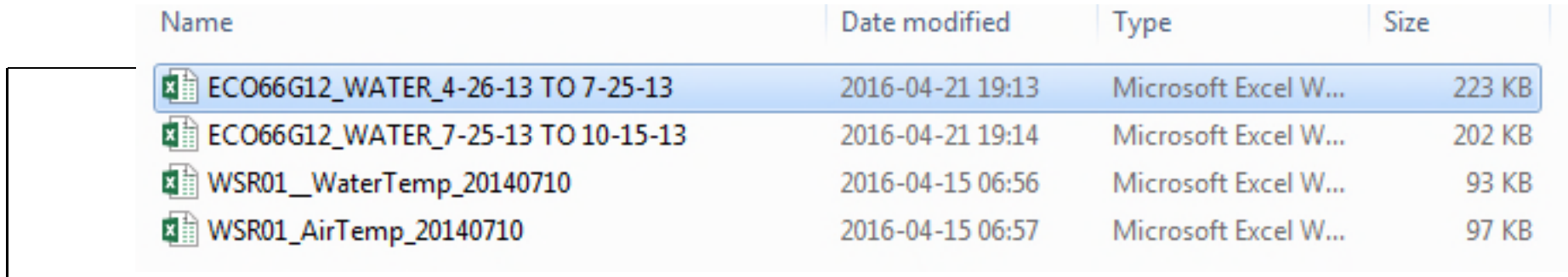
- 2 for site ECO66G12 (test dataset #1)
- 2 for site WSR01 (test dataset #2)

Note: these are examples of typical continuous sensor files. Outputs are for 1 site and for a given time period.

Examples of ‘original’ continuous sensor files…

United States Environmental Protection Agency

Open one of the ECO66G12 files

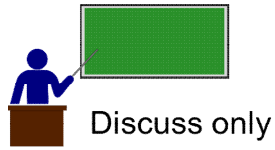


Name	Date modified	Type	Size
ECO66G12_WATER_4-26-13 TO 7-25-13	2016-04-21 19:13	Microsoft Excel W...	223 KB
ECO66G12_WATER_7-25-13 TO 10-15-13	2016-04-21 19:14	Microsoft Excel W...	202 KB
WSR01__WaterTemp_20140710	2016-04-15 06:56	Microsoft Excel W...	93 KB
WSR01_AirTemp_20140710	2016-04-15 06:57	Microsoft Excel W...	97 KB

	A	B	C	D	E	F	G
1	Plot Title: ECO66G12-Water						
2	#	Date Time, GMT-0	Abs Pres, psi	Temp, °C (LGR S	Abs Pres Bar	Water Level, f	Temp, °C (LGI v
3	1	2013-04-26 10:30	14.8024	10.063	14.3638	0.985	15.282
4	2	2013-04-26 11:00	14.7461	10.357	14.3644	0.854	14.134
5	3	2013-04-26 11:30	14.7428	10.748	14.3603	0.855	14.23
6	4	2013-04-26 12:00	14.7441	11.139	14.3627	0.853	15.473
7	5	2013-04-26 12:30	14.7431	11.528	14.3612	0.854	17.284
8	6	2013-04-26 13:00	14.7421	11.916	14.3628	0.848	18.806
9	7	2013-04-26 13:30	14.7453	12.207	14.3629	0.855	19.662

This is an example of a continuous data file from a non-vented pressure transducer (this came out of HOBOWare; other sensor outputs may look slightly different).

2. Copy and paste the files from the Data0_Original folder into the Data1_Raw folder



Data0_Original folder

Copy & paste

Name	Date modified	Type	Size
ECO66G12_WATER_4-26-13 TO 7-25-13	2016-04-21 19:13	Microsoft Excel W...	223 KB
ECO66G12_WATER_7-25-13 TO 10-15-13	2016-04-21 19:14	Microsoft Excel W...	202 KB
WSR01__WaterTemp_20140710	2016-04-15 06:56	Microsoft Excel W...	93 KB
WSR01_AirTemp_20140710	2016-04-15 06:57	Microsoft Excel W...	97 KB

Data1_Raw folder

The files in the Data1_RAW folder are your ‘working’ files.

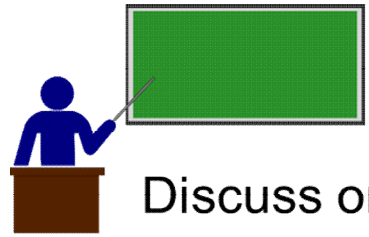
When you run the QC portion of the MasterScript, the R script pulls data from this folder.

In order for the R scripts to run, the RAW data files need to be formatted, named and saved in a specific way.

3. Required fields in the RAW files & naming scheme

Fields do not have to be in any specific order.

- **SiteID**
- "**Date Time**" (one field) OR "**Date**" **AND** "**Time**" (in 2 separate fields) OR **all 3** (in 3 fields)
  - Prefer 24H time (i.e., military). R scripts will assume any time without AM/PM is 24-hr time.
- **Parameters (as available) must be named as follows -**
  - Water Temp C
  - Air Temp C
  - Water BP psi
  - Air BP psi
  - Water Level ft

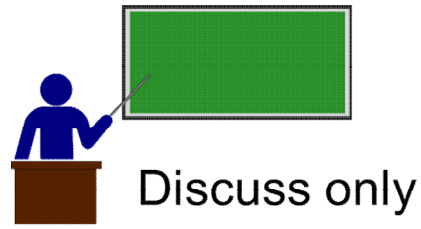


Discuss only

Optional fields

You can also include additional (optional) fields -

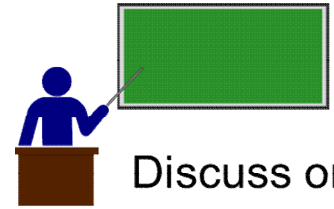
- **Air LoggerID** – unique identifier (e.g., serial number) for an land-based sensor
- **Water LoggerID** – unique identifier (e.g., serial number) for an in-water sensor
- **Air RowID** or **Water RowID** (this shows up as # in the HOBOWare output; you cannot leave it as "#" because R will treat this as a comment and ignore the entire line)



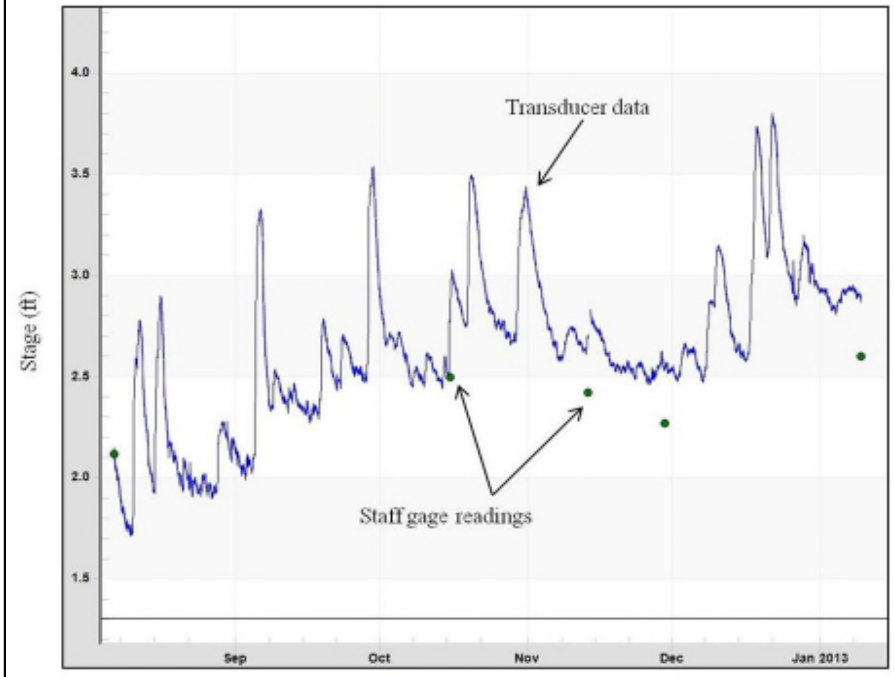
Optional but strongly encouraged - *Discrete measurements*

Discrete measurements are instantaneous measurements taken during field visits that are compared to the closest continuous sensor measurement. They are used for –

- Accuracy checks
- Making corrections to your data (as needed)
- Validating your data (making them more defensible)



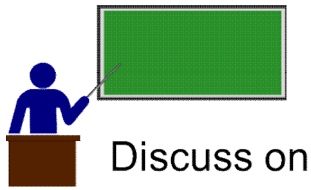
Discuss only



EXAMPLE –

Staff gage readings can provide a quality check of transducer data. In this example, staff gage readings stopped matching transducer readings in November, indicating that the transducer or gage could have changed elevation.

Adding discrete measurements into RAW data files



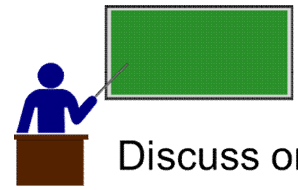
Enter the discrete value in the sensor measurement row with the **closest date time** (no need to create a new line for just the discrete measurement; in fact, the way the scripts are set up, we prefer that you don't enter a new line with only the discrete measurement).

The column heading needs to have the **prefix "Discrete**" (e.g., Discrete Water Temp C).

	A	B	C	D	E	F	G	H
1	Water RowID	Date	Time	Water Temp C	SiteID	Water LoggerID	Date Time	Discrete.Water Temp C
2	3	22-Oct-13	13:00	9.632	HRCC	10406272	10/22/2013 13:00	9.39
3	4	22-Oct-13	13:30	9.632	HRCC	10406272	10/22/2013 13:30	
4	5	22-Oct-13	14:00	9.682	HRCC	10406272	10/22/2013 14:00	
5	6	22-Oct-13	14:30	9.706	HRCC	10406272	10/22/2013 14:30	
6	7	22-Oct-13	15:00	9.78	HRCC	10406272	10/22/2013 15:00	
7	8	22-Oct-13	15:30	9.854	HRCC	10406272	10/22/2013 15:30	
8	9	22-Oct-13	16:00	9.952	HRCC	10406272	10/22/2013 16:00	

IMPORTANT!
MAKE SURE YOUR DISCRETE AND MATCHING CONTINUOUS MEASUREMENT ARE BOTH IN EITHER STANDARD TIME OR DAYLIGHT SAVINGS TIME.

Field naming scheme



Non-vented pressure transducer (after water and air sensor data were combined in HoboWare)

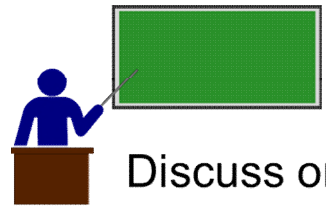
Before…

	A	B	C	D	E	F	G
1	Plot Title: ECO66G12-Water						
2	#	Date Time, GMT-0	Abs Pres, psi	Temp, °C (LGR S	Abs Pres Bar	Water Level, f	Temp, °C (LGI
3	1	2013-04-26 10:30	14.8024	10.063	14.3638	0.985	15.282
4	2	2013-04-26 11:00	14.7461	10.357	14.3644	0.854	14.134
5	3	2013-04-26 11:30	14.7428	10.748	14.3603	0.855	14.23

And after…

A	B	C	D	E	F	G	H	I	J
AW RowID	SiteID	Date Time	Water Temp C	Air Temp C	Water Level ft	Water BP psi	Air BP psi	Water Logger ID	Air Logger ID
1	ECO66G12	4/26/2013 10:30	10.063	15.282	0.985	14.8024	14.3638	10229557	10229571
2	ECO66G12	4/26/2013 11:00	10.357	14.134	0.854	14.7461	14.3644	10229557	10229571
3	ECO66G12	4/26/2013 11:30	10.748	14.23	0.855	14.7428	14.3603	10229557	10229571
4	ECO66G12	4/26/2013 12:00	11.139	15.473	0.853	14.7441	14.3627	10229557	10229571
5	ECO66G12	4/26/2013 12:30	11.528	17.284	0.854	14.7431	14.3612	10229557	10229571
6	ECO66G12	4/26/2013 13:00	11.916	18.806	0.848	14.7421	14.3628	10229557	10229571
7	ECO66G12	4/26/2013 13:30	12.207	19.662	0.855	14.7453	14.3629	10229557	10229571
8	ECO66G12	4/26/2013 14:00	12.594	20.424	0.853	14.7351	14.3537	10229557	10229571

4. Check for rows with no data



Discuss only

FILE HOME INSERT PAGE LAYOUT FORMULAS DATA REVIEW VIEW DEVELOPER Nuance PDF

Cut Copy Format Painter Paste — Clipboard

Calibri 11 — Font

Wrap Text Merge & Center — Alignment

Custom — Number

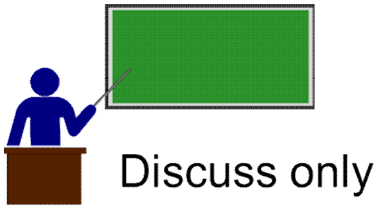
Conditional Formatting Format as Table Cell Styles — Styles

B4 7/25/2013 12:30:00 PM

	A	B	C	D	E	F	G	H	I	J	K	L
4020	4017	10/17/2013 4:30	14.0234	15.473	13.9275	0.426	15.951					
4021	4018	10/17/2013 5:00	14.0257	15.473	13.9275	0.431	15.951					
4022	4019	10/17/2013 5:30	14.0257	15.473	13.9275	0.431	15.951					
4023	4020	10/17/2013 6:00	14.0234	15.473	13.9233	0.435	16.046					
4024	4021	10/17/2013 6:30	14.0257	15.473	13.9257	0.435	16.046					
4025	4022	10/17/2013 7:00	14.0234	15.473	13.9257	0.43	16.046					
4026	4023	10/17/2013 7:30	14.0234	15.473	13.9233	0.435	16.046					
4027	4024	10/17/2013 8:00	14.0238	15.569	13.9239	0.435	16.141					
4028	4025	10/17/2013 8:30	14.0238	15.569	13.9216	0.44	16.141					
4029	4026	10/17/2013 9:00	14.0285	15.569	13.9243	0.445	16.237					
4030	4027	10/17/2013 9:30	14.0331	15.569	13.9291	0.444	16.237					
4031	4028	10/17/2013 10:00	13.9306	15.664	13.9325	0.2	16.427					
4032	4029	10/17/2013 10:01							Logged			
4033	4030	10/17/2013 10:01								Logged	Logged	
4034												

Remove entries like these…

5. Rename the files in the Data1_RAW folder



Need to use consistent file naming convention.

Example (non-vented pressure transducer, with all parameters in one file):

- **ECO66G12_AW_20130426_20130725**
  - Site ID (no spaces or underscores) = ECO66G12
  - Data Type (Water/Air/AW/Gage) = AW (air and water)
  - Date, Start (YYYYMMDD) = 20130426
  - Date, End (YYYYMMDD) = 20130725
  - Each element separated by underscore ("_").

- Allows for multiple files to be kept in the same directory.
- R scripts will pull data from files based on Site ID, data type and date range.

File naming scheme

Discuss only

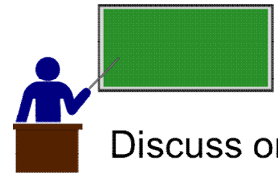
BEFORE	AFTER
ECO66G12_WATER_4-26-13 TO 7-25-13.xlsx	ECO66G12_AW_20130426_20130725.csv
ECO66G12_WATER_7-25-13 TO 10-15-13.xlsx	ECO66G12_AW_20130725_20131015.csv

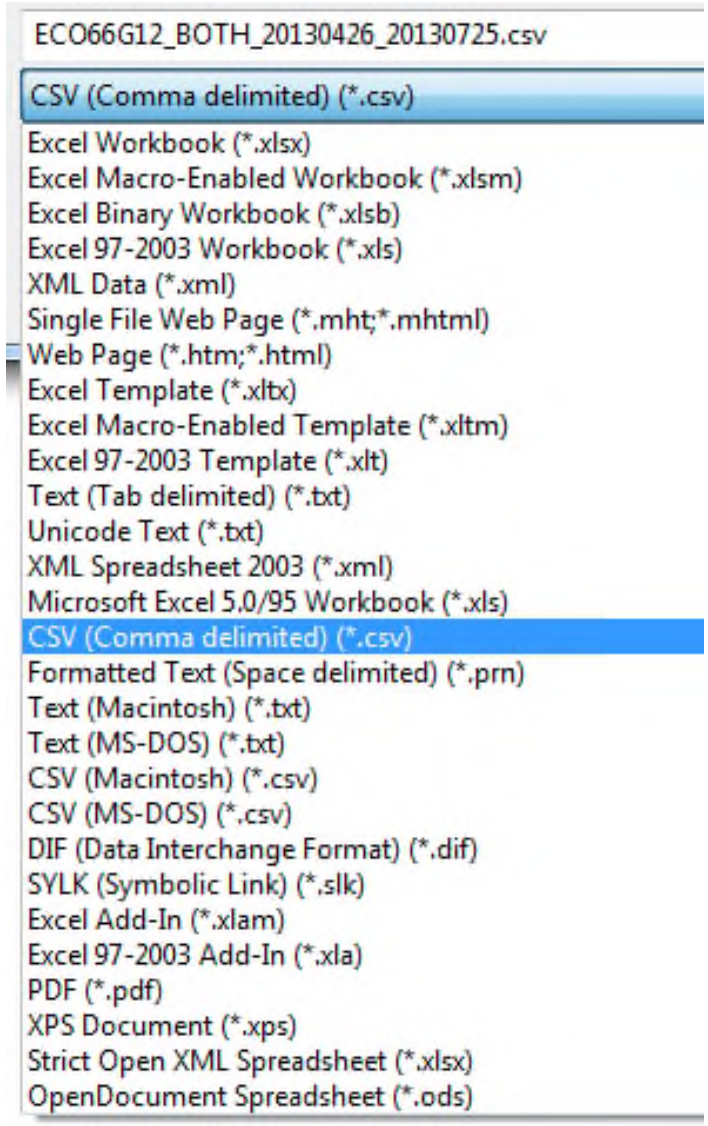
Example: this file is from site ECO66G12
It contains both air and water sensor data (=AW)
Period of record = April 26, 2013 through July 25, 2013

What if this was just water sensor data?
Data type would be '_Water'
What if this was just air sensor data?
Data type would be '_Air'
What if this was USGS gage data?
Data type would be '_Gage'

5. Save as .csv file

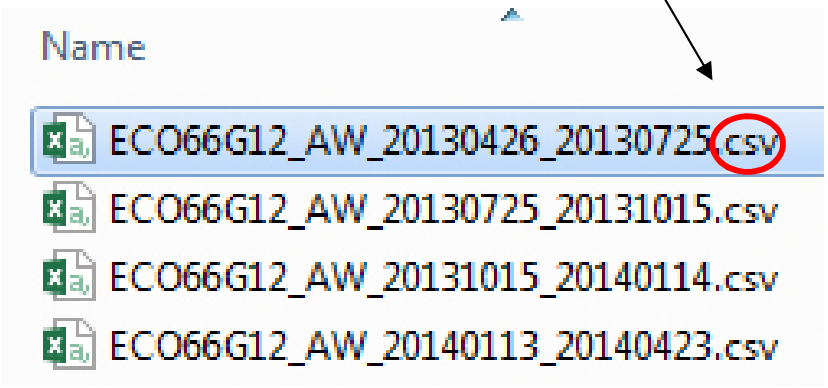
.csv = comma delimited





File

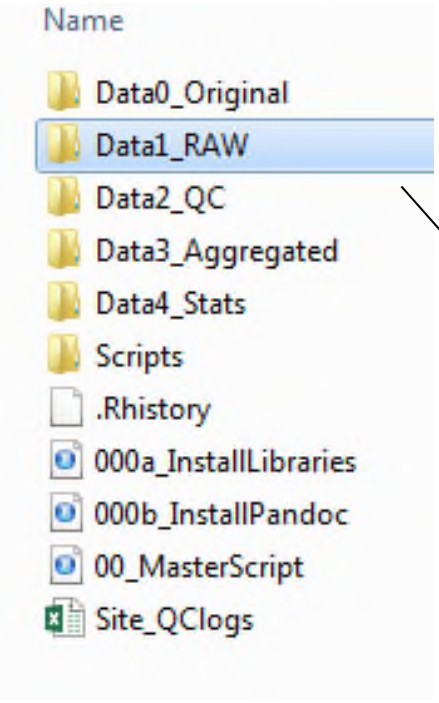
- Save As
  - CSV (Comma delimited) (*.csv)

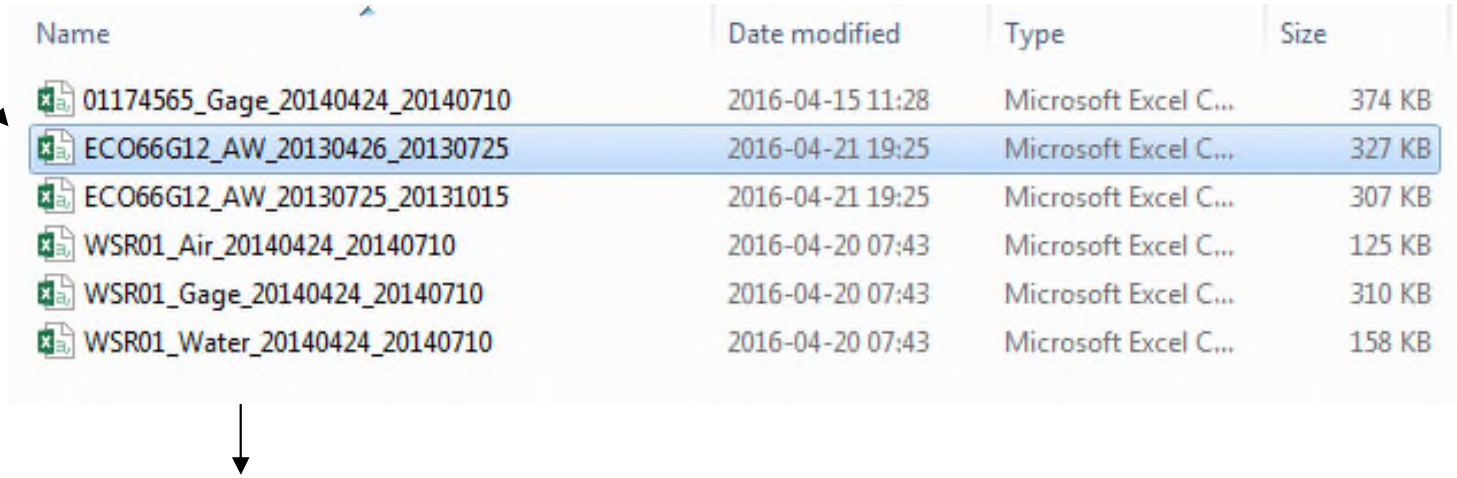


Let's take a look at the test files that are in the Data1_RAW folder…

Open the Data1_RAW folder.

Open one of the ECO66G12 files.



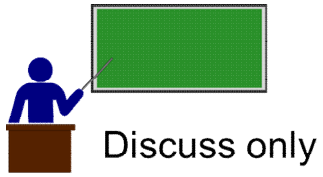


	A	B	C	D	E	F	G	H	I	J	K
1	AW RowID	SiteID	Date Time	Water BP psi	Water Temp C	Air BP psi	Water Level ft	Air Temp C	Discrete Water Temp C	Discrete Air Temp C	Discrete Water Level ft
2	1	ECO66G12	4/26/2013 10:30	14.8024	10.063	14.3638	0.985	15.282	10.055	15.24	0.98
3	2	ECO66G12	4/26/2013 11:00	14.7461	10.357	14.3644	0.854	14.134			
4	3	ECO66G12	4/26/2013 11:30	14.7428	10.748	14.3603	0.855	14.23			
5	4	ECO66G12	4/26/2013 12:00	14.7441	11.139	14.3627	0.853	15.473			
6	5	ECO66G12	4/26/2013 12:30	14.7431	11.528	14.3612	0.854	17.284			
7	6	ECO66G12	4/26/2013 13:00	14.7421	11.916	14.3628	0.848	18.806			

EPA
United States
Environmental Protection
Agency

Running the QC_Raw portion of the MasterScript

QC process - overview



1. Open the MasterScript (double click on it)
Make the appropriate entries (operation, SiteID, data type, date range)

2. Run the script, which searches the Data1_RAW folder for the appropriate files

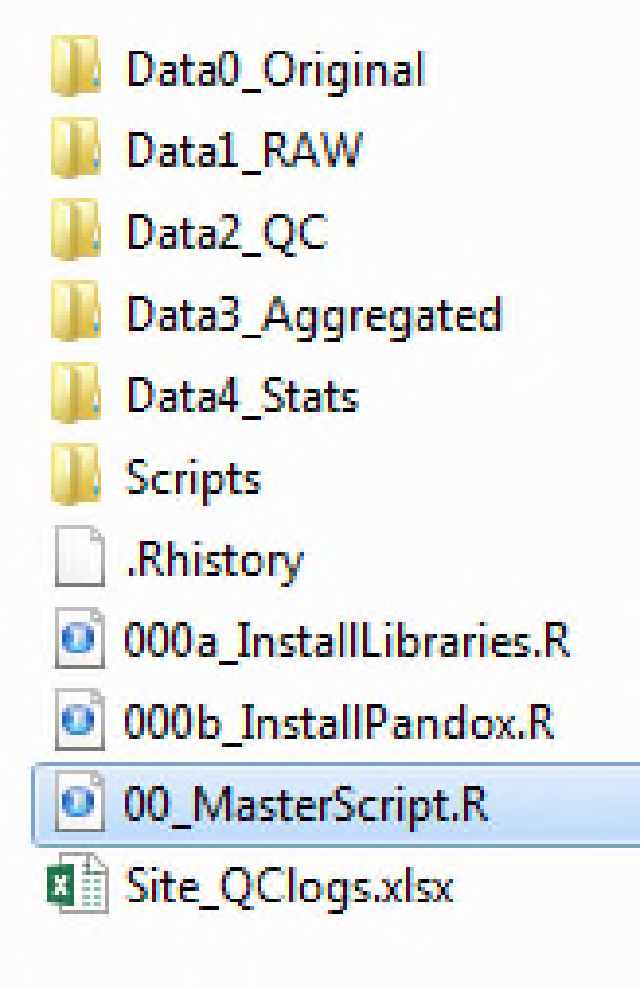
3. Open the **Data2_QC folder** – there should be 2 outputs for each input file – .**csv** file & **QC report** (Word document)

4. Check flagged values in the QC report and .csv file

5. Perform **'manual' visual checks** on the plots

6. **Make corrections as needed**

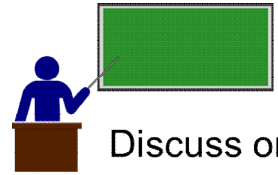
1. Open the MasterScript & make the appropriate entries



Go to the directory & double click on the **MasterScript** to open it.

Use this technique, otherwise you may encounter errors (e.g., sometimes people get error messages when they open the RStudio program then navigate to the file (vs. opening it from the directory)).

The Masterscript will look like this…



RStudio

File Edit Code View Plots Session Build Debug Tools Help

Project: (None)

00_MasterScript.R

Source on Save | Run | Source

```
# Manipulates Continuous Data Files (run different operations based on user inpu
# Erik.Leppo@tetratech.com (EWL), 2015-11-19
##################
# It is assumed that this R script is stored in a directory with the data files
# If this script is run in TINN-R or R-Studio there is no need to define the worl
#####################################################################
# clear the workspace
rm(list=ls())
# Define working Directory
# if specify directory use "/" not "\" (as used in Windows) and leave off final
#myDir.BASE  <- "C:/Users/Erik.Leppo/Documents/NCEA_DataInfrastructure/Erik"
myDir.BASE <- getwd()
setwd(myDir.BASE)
# library (load any required helper functions)
source(paste(myDir.BASE,"Scripts","fun.Master.R",sep="/"))
#####################################################################
# USER input in this section (see end of script for explanations)
#####################################################################
#
# PROMPT; Operation
Selection.Operation <- c("GetGageData","QCRaw", "Aggregate", "SummaryStats")
myData.Operation    <- Selection.Operation[2]  #number corresponds to intended o
#
# PROMPT; Site ID
# single site;          "test2"
# group of sites;       c("test2", "HRCC", "PBCC", "ECO66G12", "ECO66G20", "ECO68
myData.SiteID       <- "ECO66G12"
#
# PROMPT; Data Type
# Type of data file
Selection.Type      <- c("Air","Water","AW","Gage","AWG","AG","WG") # only one a
myData.Type         <- Selection.Type[3] #number corresponds to intended operati
#
# PROMPT; Start Date
# YYYY-MM-DD ("-" delimiter), leave blank for all data ("1900-01-01")
```

1:1 (Top Level) R Script

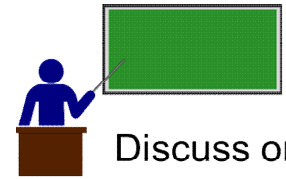
Console C:/Users/Jen.Stamp/Desktop/NWQMC_Rworkshop/Rscripts_NWQMC_Test_20160420/

```
R version 3.2.3 (2015-12-10) -- "Wooden Christmas-Tree"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: i386-w64-mingw32/i386 (32-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>
```

Viewer

Environment History Files Plots Packages Help

There are 5 places in the MasterScript where you'll need to make entries…



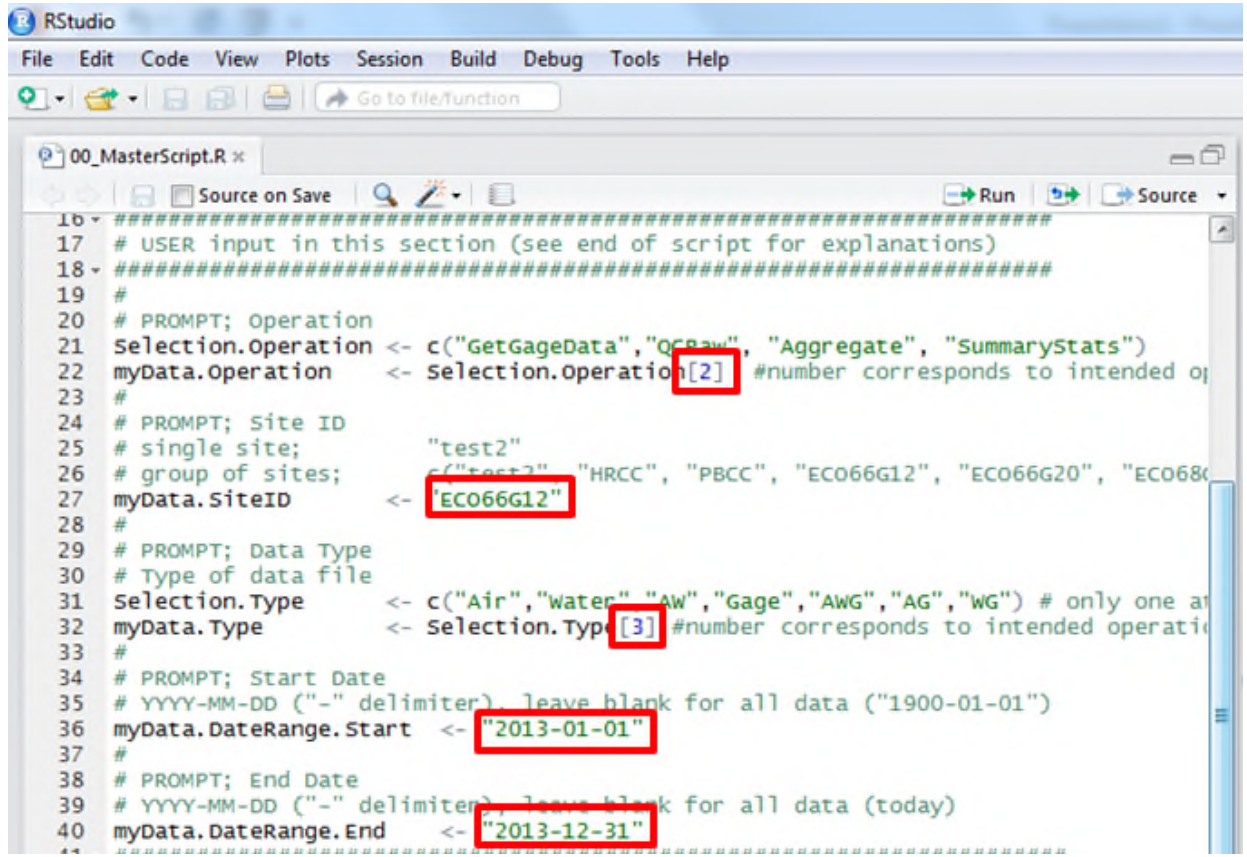
1) Select the **operation**
   - GetGageData, QCRaw, Aggregate or SummaryStats

2) Enter the **SiteID**

3) Enter the **data/sensor type**
   - Air, water, air/water, gage, air/water/gage, air/gage, water/gage

4) Enter a **start date**

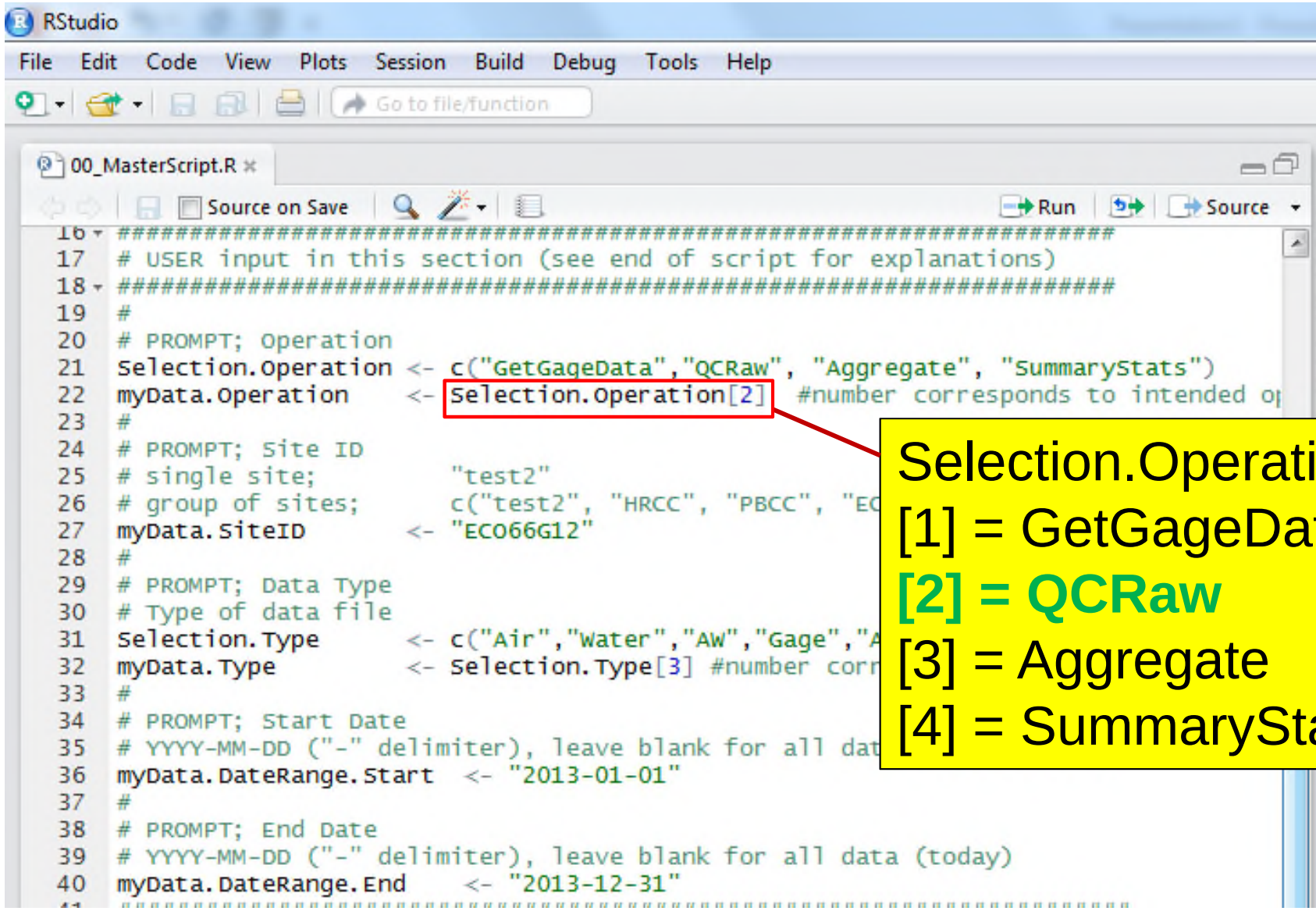
5) Enter an **end date**



This only shows the Source (left panel)

Select operation

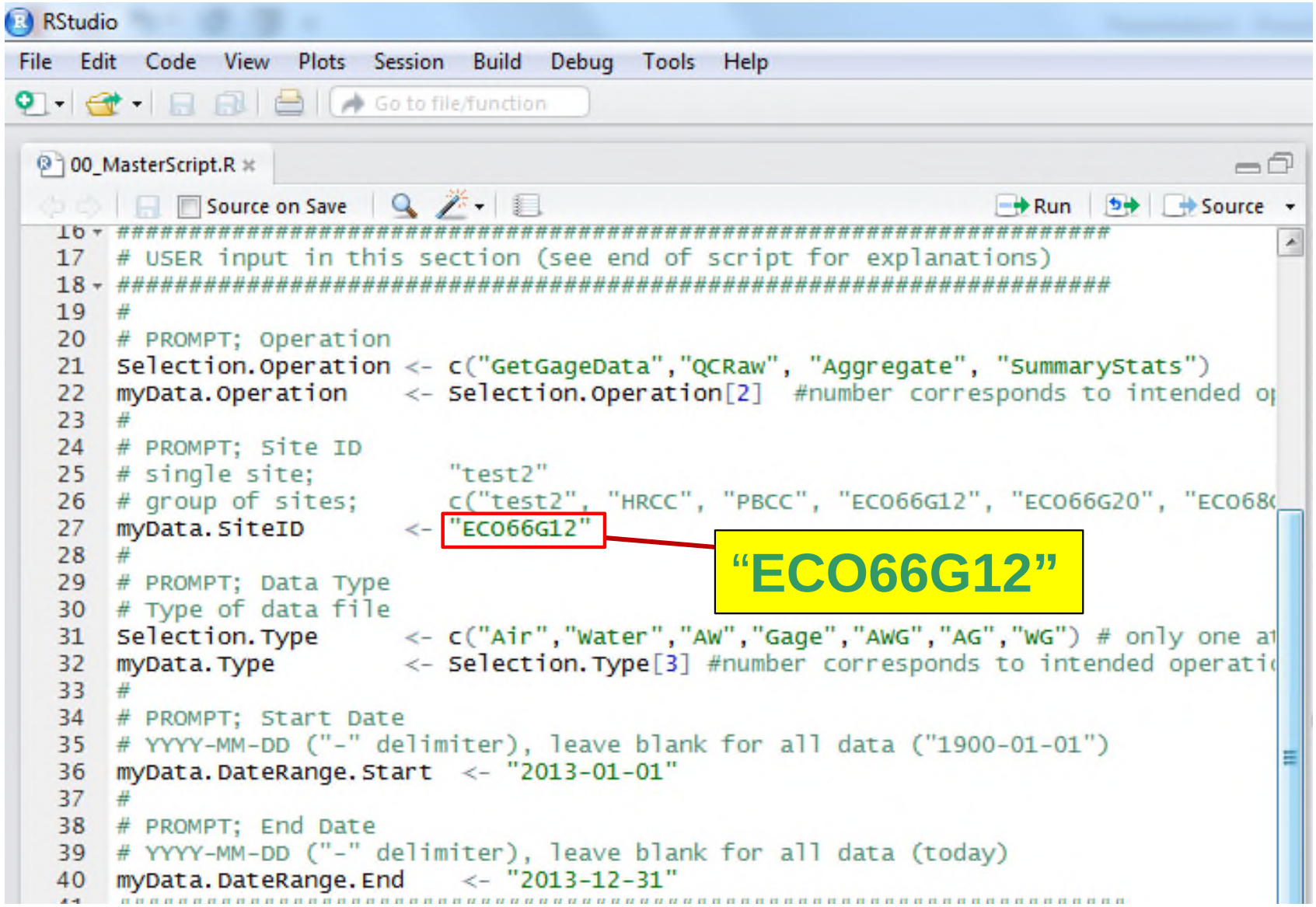
Enter [2]



This only shows the Source (left panel)

Enter the SiteID

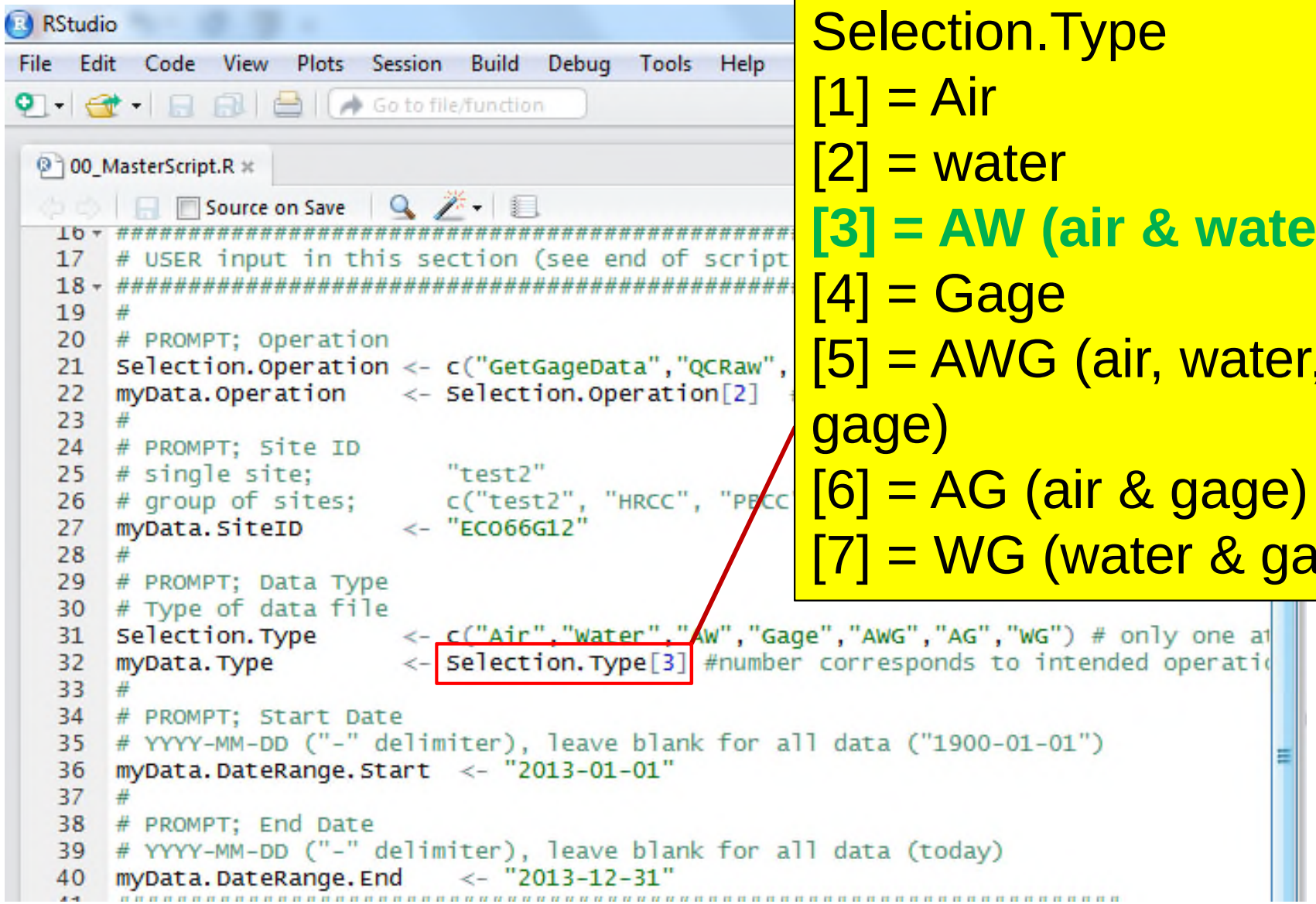
"ECO66G12"



This only shows the Source (left panel)

Enter the data/sensor type

Enter [**AW**]



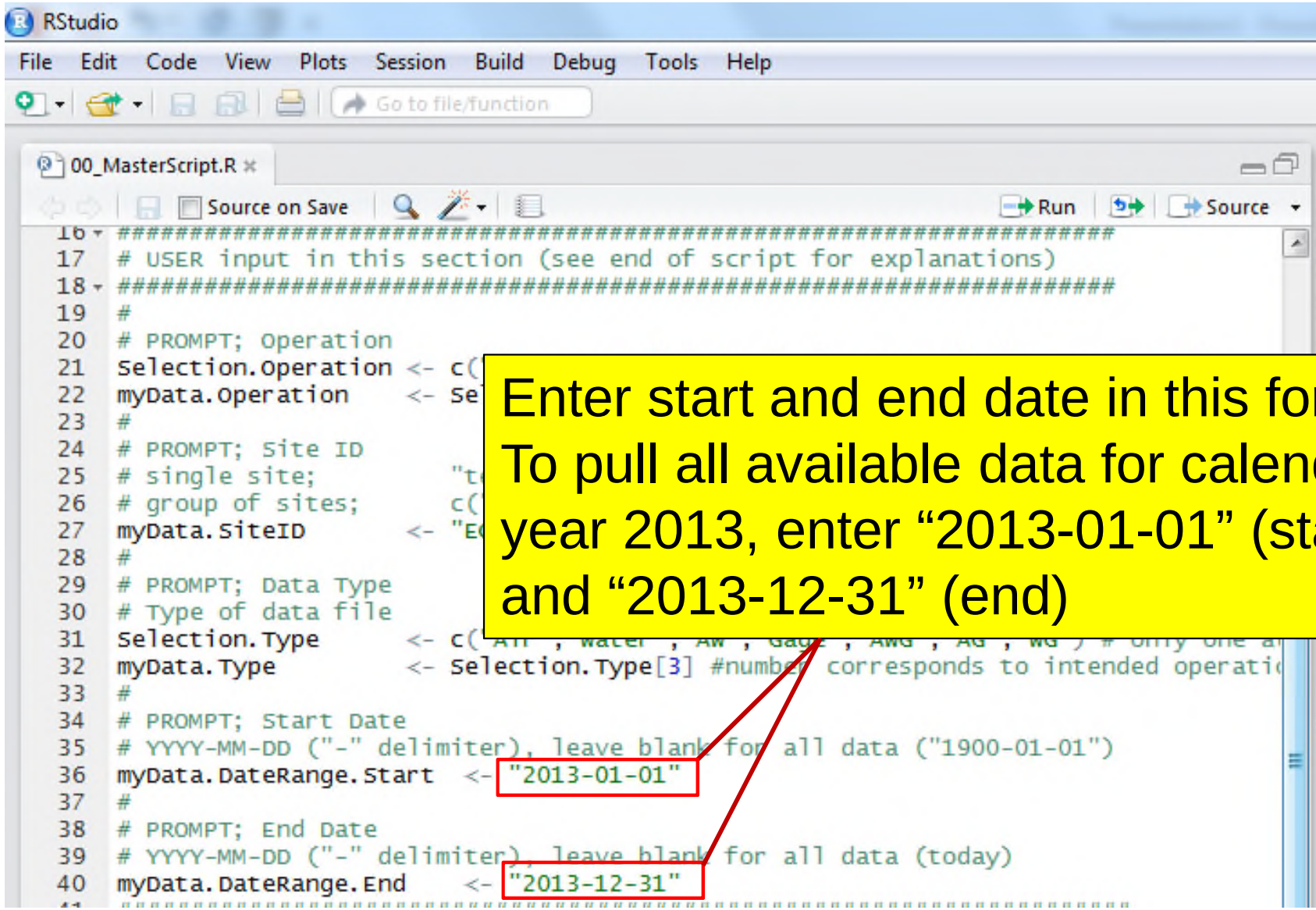
Selection.Type
[1] = Air
[2] = water
[3] = AW (air & water)
[4] = Gage
[5] = AWG (air, water, gage)
[6] = AG (air & gage)
[7] = WG (water & gage)

This only shows the Source (left panel)

Enter start & end dates

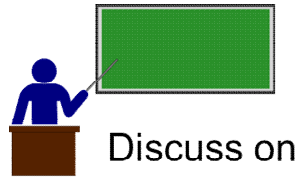
Start “**2013-01-01**”; End “**2013-12-31**”

See next slide for other options



This only shows the Source (left panel)

Date range



The script looks through the date ranges in the file names and runs QC on any individual files that have dates that overlap with the range that you enter.

For this exercise, let's run both files at once. You can do this several different ways.

- Enter 2013-01-01 (start) and 2013-12-31 (end) ← Let's use this
- Enter 2013-04-26 (start) and 2013-10-15 (end)

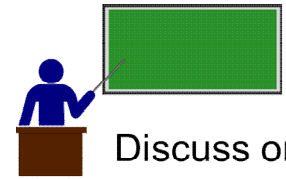
If you want to pull all available data for this site, leave the entry blank (you still need start and end quotes) - ""

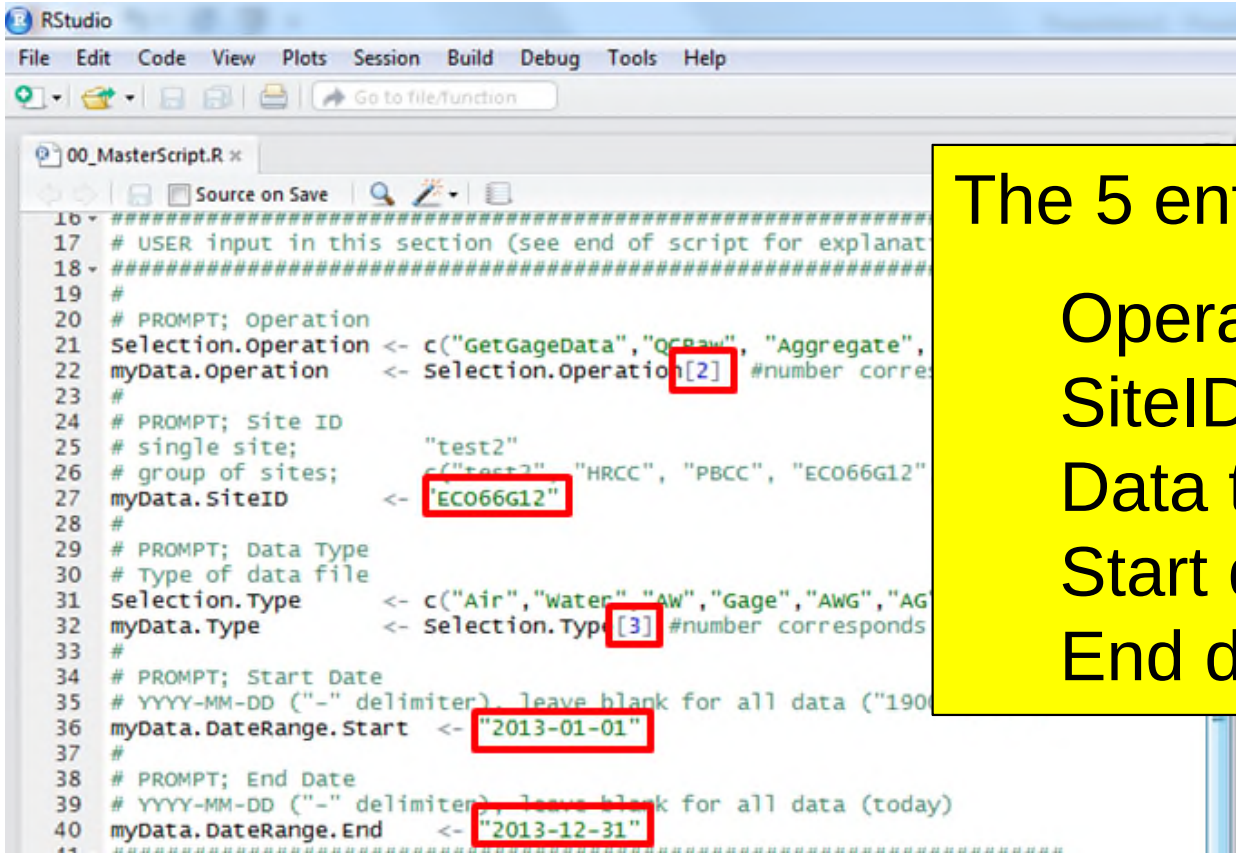
What if you only want to run one file at time? Pick a date range that covers only the desired file. You can do this several different ways. One example –

- Enter 2013-04-26 (start) to 2013-07-24 (end)*

*if you entered 2013-07-25 as the end date, it would include the 2nd file because the date ranges overlap.

Entries recap





The 5 entries should read –

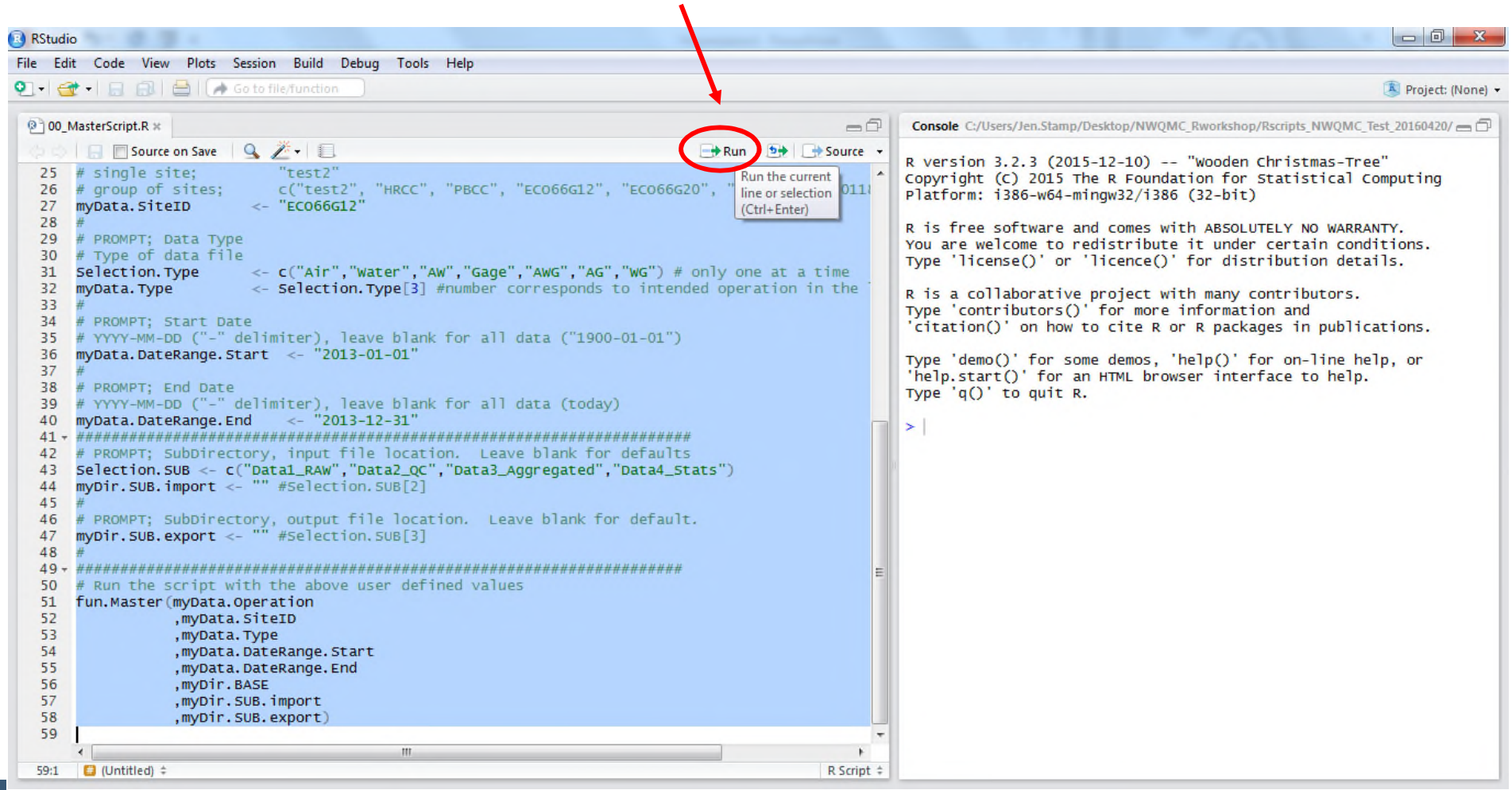
- Operation [**2**]
- SiteID “**ECO66G12**”
- Data type [**3**]
- Start date “**2013-01-01**”
- End date “**2013-12-31**”

This only shows the Source (left panel)

2. Run the script

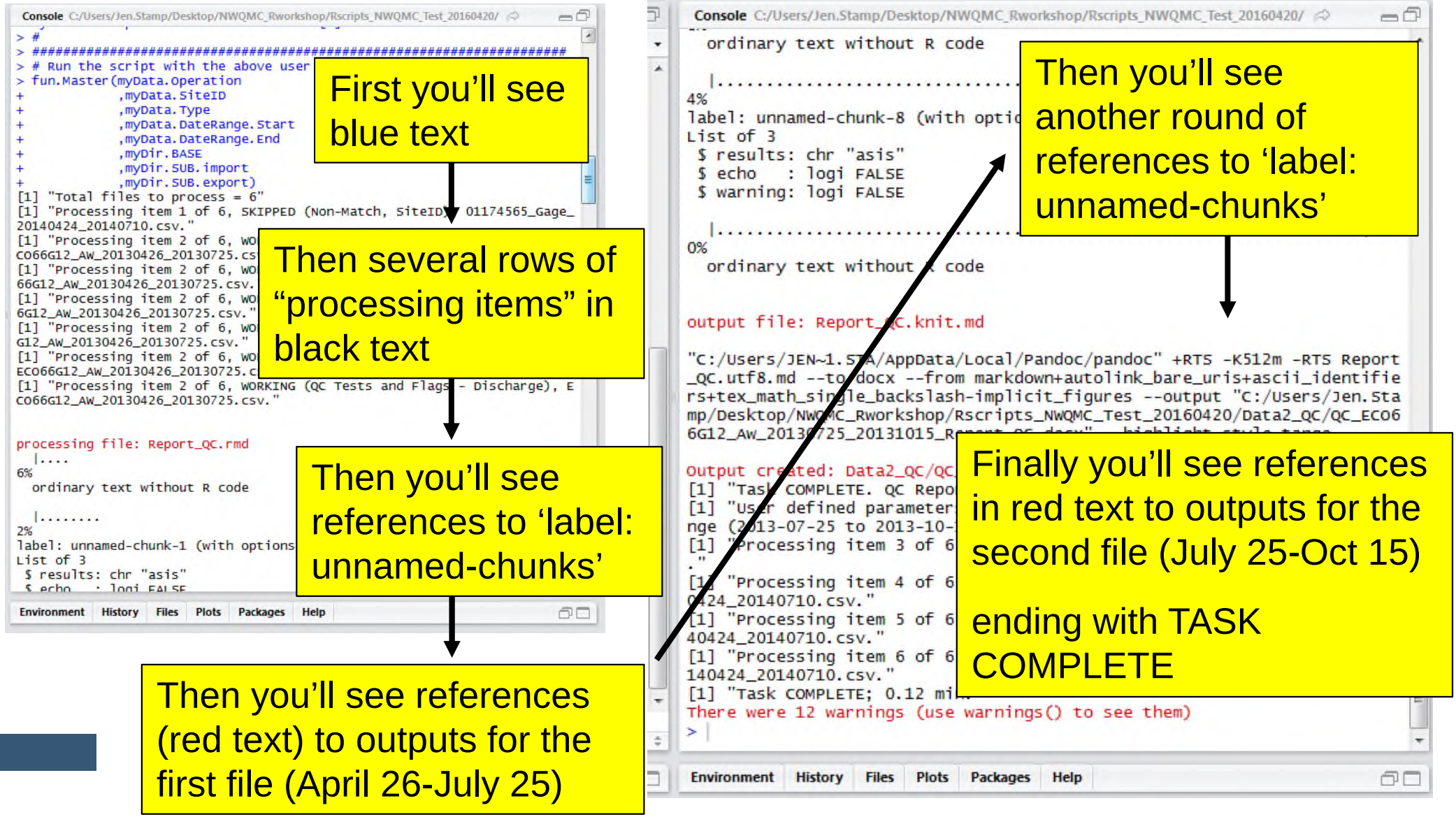
Using your cursor, highlight the script from line 7 to the end; it will turn blue)

Then click RUN



Check the Console (right) panel

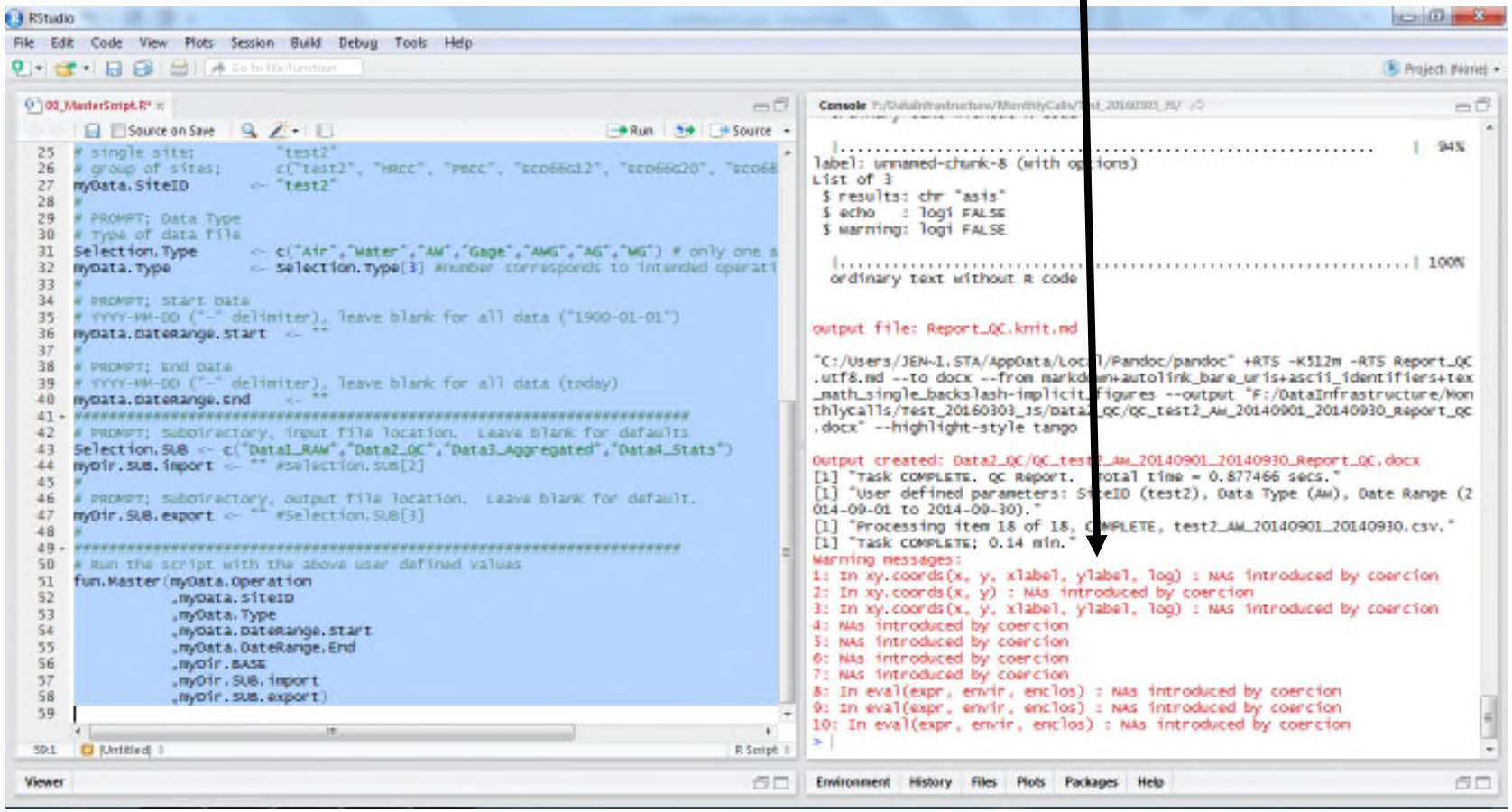
If the script ran successfully, you should see a running log like this…



Warning Messages

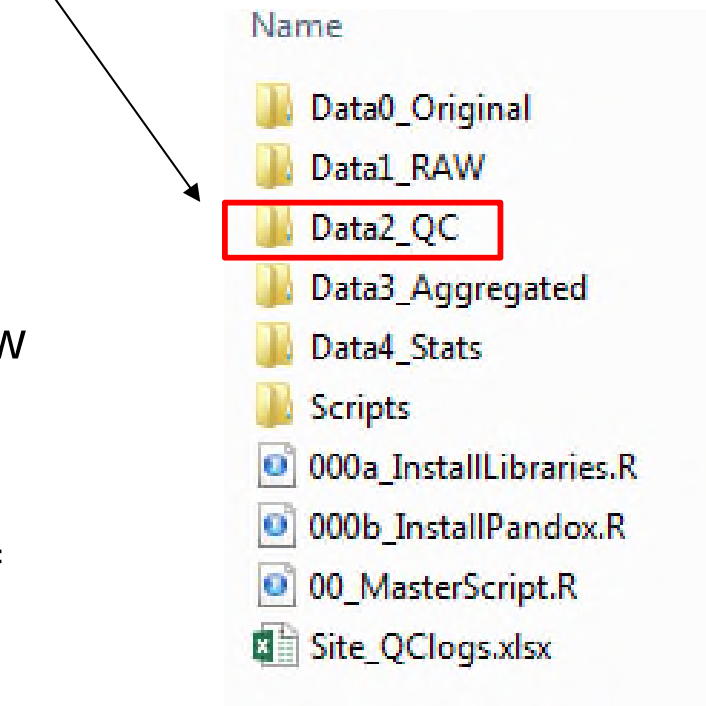
In the Console output you may see some warning messages in red text. These sometimes appear, even if the script runs properly.

If warning messages appear, check your **Data2_QC** folder to make sure the outputs were generated.



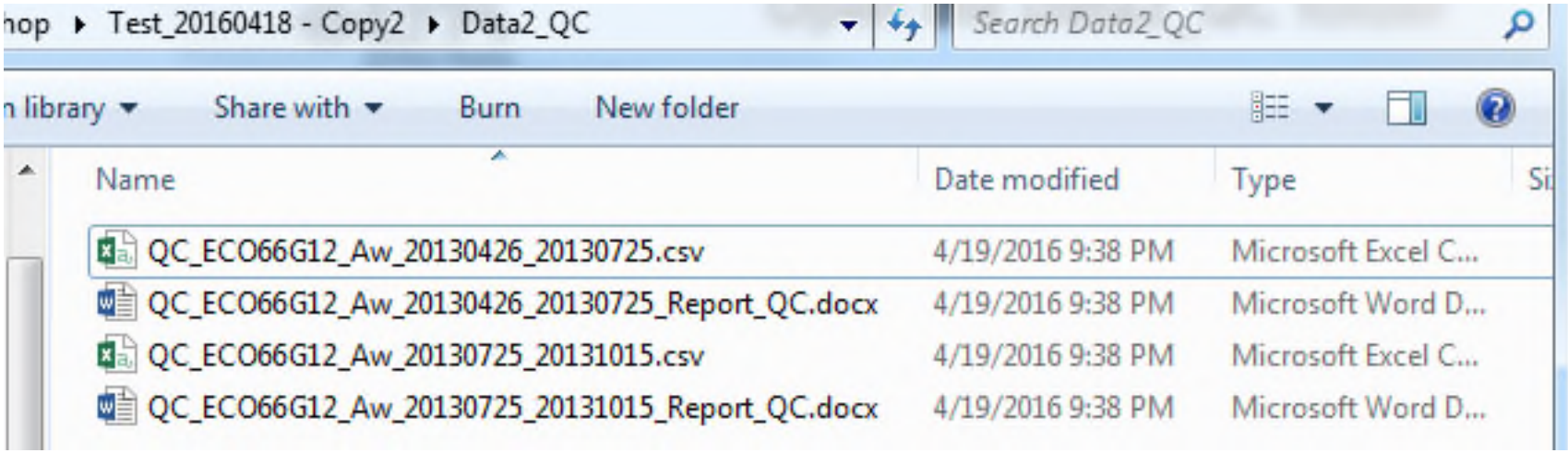
Note: it typically takes less than a minute for the script to run on these test data. If it takes longer, there may be a problem.

3. Open the Data2_QC folder



Verify that there are 2 outputs for each file (4 total files):

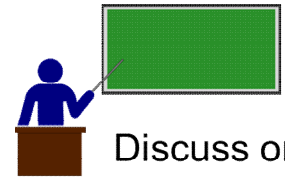
- **csv** file (=the original Data1_RAW file with new columns added in – flags, raw + final data, and comments fields)
- **QC report** (Word document) with summary of missing data, flags, time series plots



Overview of QC outputs

United States Environmental Protection Agency

2 outputs



Discuss only

QC report (Word document)

Data Quality Control Report

Report Date: 2016-02-05

DATA FILE INFORMATION

Filename: QC_test1_Aw_20130725_20131015.csv

SiteID: test1

Period of Record, Requested: 2013-07-25 to 2013-10-15

Period of Record, Actual: 2013-07-25 to 2013-10-15

Recording Interval: 30 minutes

Data Type: Aw

Parameters Included: Water.Temp.C, Air.Temp.C, Water.BP.psi, Air.BP.psi, Water.Level.ft,

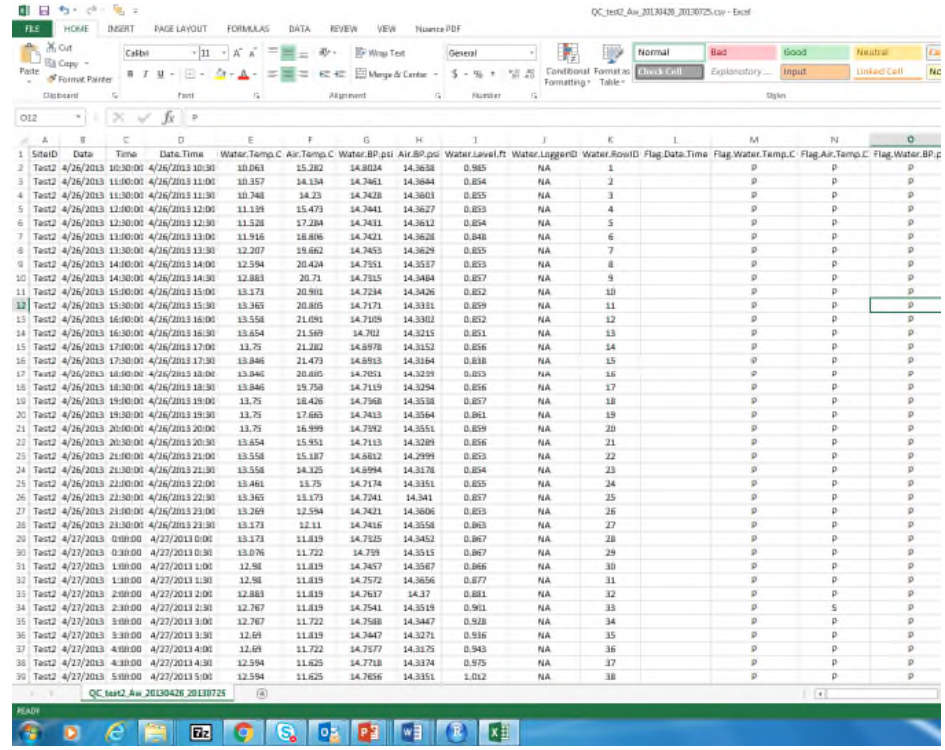
DATA SUMMARY, OVERALL

Month and Day versus number of records.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
8	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48
9	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48
10	48	48	48	48	48	48	48	48	48	48	48	48	49	48	21

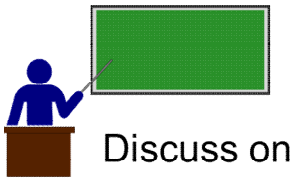
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
7	0	0	0	0	0	0	0	0	0	27	48	48	48	48	48	48

.csv file



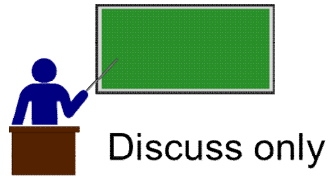
QC report (Word)

4 Main components



1. **Data file information** (siteID, date range, parameters, recording interval)
2. **'Count' tables (# measurements/day)** so that you can find inconsistencies (too few measurements (=missing data) or too many measurements)
3. **Results from QC tests** - # of entries marked as pass (P), suspect (S), fail (F) or no data (X) (see next slide)
4. **Time series plots** – for each individual parameter & several combined parameters (water & air temperature; water temperature and water level); includes discrete measurements (if available)

Flags/QC tests



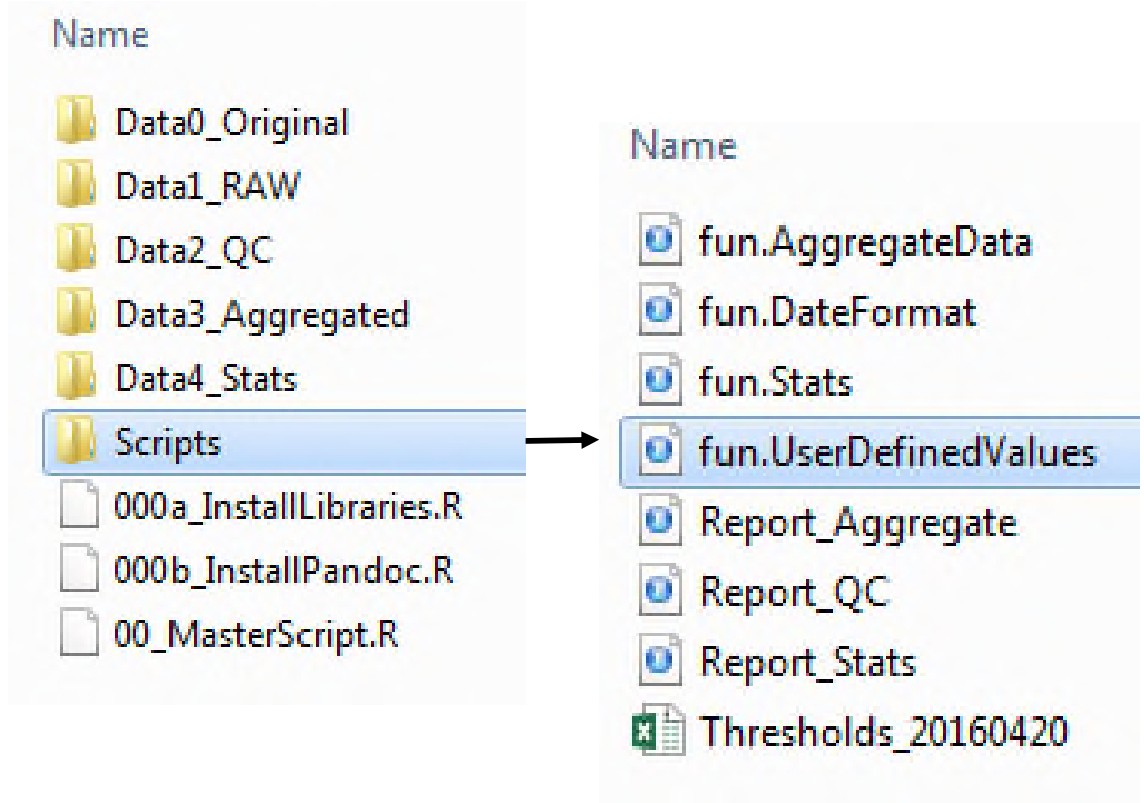
- **Unrealistic values** (‘Gross range’)
  - Entries are flagged if values are above or below upper and lower limits
- **Spikes**
  - Entries are flagged if adjacent points change by more than ‘x’ amount
- **Rate of change** (RoC)
  - Entries are flagged if the RoC exceeds a given threshold (e.g., ≥ 3 st dev within 25 hrs)
- **Flat line**
  - Entries are flagged if a certain # of consecutive measurements are within a certain amount of each other (e.g., >10 consecutive temperature measurements are within 0.01 degrees C of one another)

User-defined thresholds

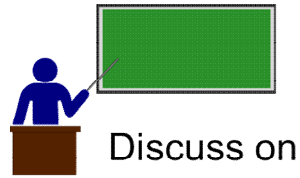
Discuss only

The thresholds for the QC tests (gross, spike, rate of change and flat line) are in the **fun.UserDefinedValues.R file** in the **Scripts folder**.

You can go into this file and change thresholds if desired.



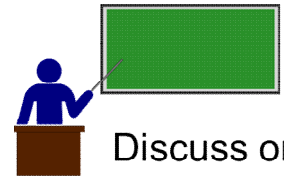
QC report – Organizational scheme



1. **Data file information** (siteID, date range, parameters, recording interval)
2. **Overall summary**
   - Count table
   - Overall flag (# entries marked as pass, suspect, fail or missing data, based on the *worst* flag across all of the QC tests)
3. **Individual parameters**
   - Count table
   - Flags by QC test (gross, spike, rate of change and flat line)
   - Time series plot
4. **Multi-parameter time series plots** (as available)

QC report – Section 1

Data file information



Discuss only

Data Quality Control Report

Report Date: 2016-04-21

DATA FILE INFORMATION

Filename: QC_ECO66G12_Aw_20130725_20131015.csv

SiteID: ECO66G12

Period of Record, Requested: 2013-07-25 to 2013-10-15

Period of Record, Actual: 2013-07-25 to 2013-10-15

Recording Interval: 30 minutes

Data Type: Aw

Parameters Included: Water.Temp.C, Air.Temp.C, Water.BP.psi, Air.BP.psi, Water.Level.ft, Discrete.Water.Temp.C, Discrete.Air.Temp.C, Discrete.Water.Level.ft,

QC report – Section 2

Overall Summary

Discuss only

‘Count’ tables
(# measurements/day)

DATA SUMMARY, OVERALL

Month and Day versus number of records.

Month

Day

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
8	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48
9	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48
10	48	48	48	48	48	48	48	48	48	48	48	48	49	48	21

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
7	0	0	0	0	0	0	0	0	0	27	48	48	48	48	48	48
8	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48
9	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	0

Expected number of records per day is 48.

Sensor was set to record at 30-minute intervals. This equates to 48 measurements per day.

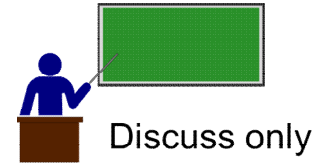
Missing measurements on days of download (July 25 and Oct 15).

October 13 had 1 extra measurement (49).

This is an example from a different data file (not the files you are working with)

QC report – Section 2

Overall Summary



Overall flags by parameter

Water.Temp.C

P	S
3905	31

Air.Temp.C

P
3936

Water.BP.psi

P	S
3934	2

Air.BP.psi

P
3936

Water.Level.ft

P	S
3911	25

For each parameter, # of entries marked as pass (P), suspect (S), fail (F) or missing data (X), for any of the QC tests.

In this example, 3905 of the water temperature entries were marked as pass, and 31 were marked as suspect.

QC report – Section 3

Individual parameters – e.g., water temperature

QC TESTS, Water.Temp.C

Parameter

Flags

Flag summary, overall number of records.

P	S
3905	31

Flag summary (overall)

Month and Day versus number of records.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
8	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48
9	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48
10	48	48	48	48	48	48	48	48	48	48	48	48	48	48	21

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
7	0	0	0	0	0	0	0	0	0	27	48	48	48	48	48	48
8	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48
9	48	48	48	48	48	48	48	48	48	48	48	48	48	48	48	0

Count table

Expected number of records per day is 48.

Flags by QC Test

Flags, Gross

P
3936

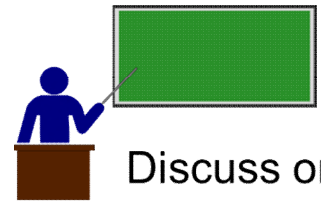
Flags, Spike

P	X
3935	1

Flags, RoC

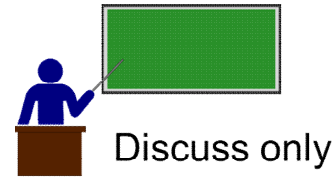
P	X
3935	1

Flag summary, broken down by QC test

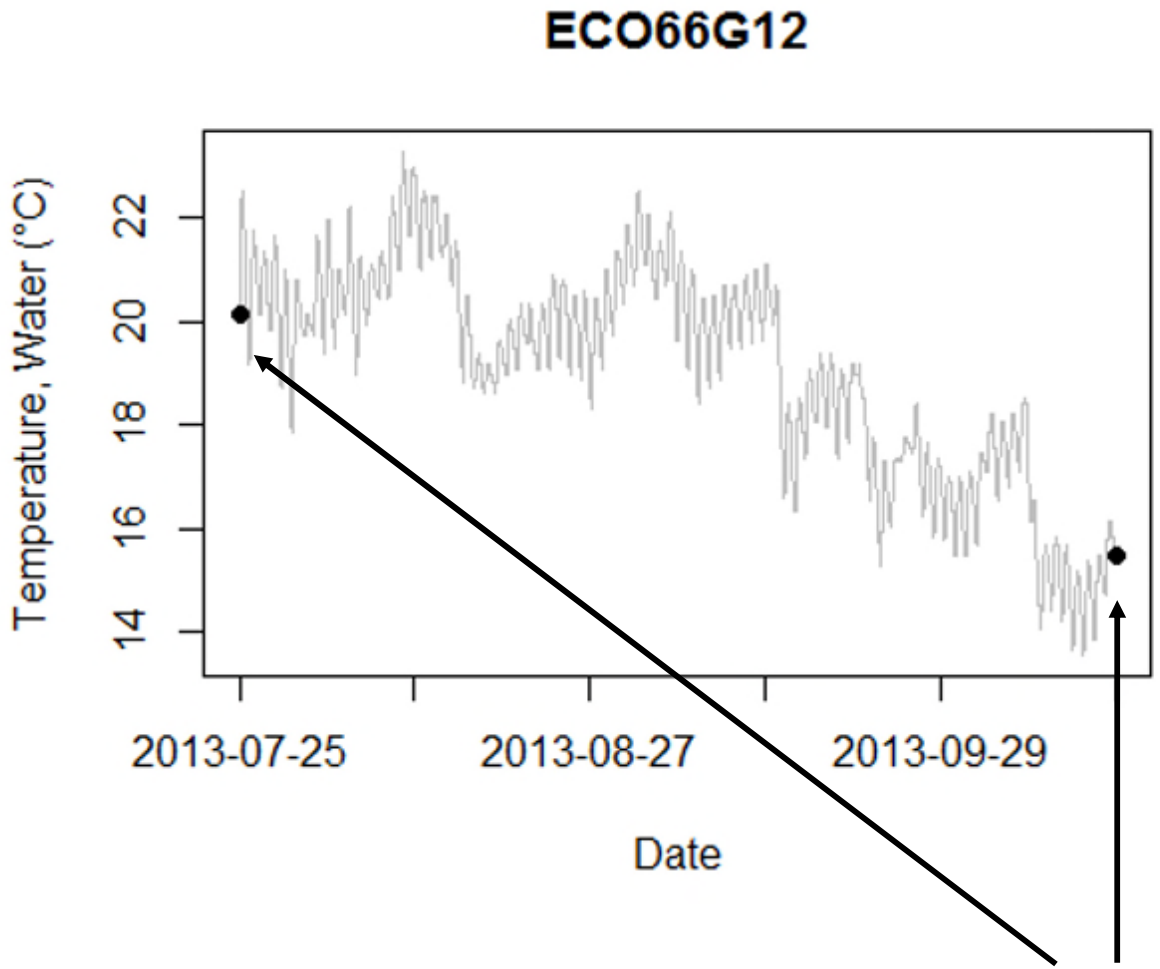


Discuss only

QC report – Section 3



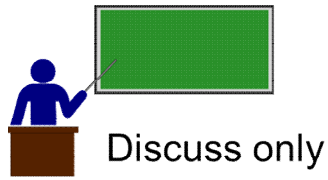
Individual parameters continued…



Time series plot (all values plotted vs. date/time) for water temperature

This example includes discrete measurements

Visual checks on time series plots

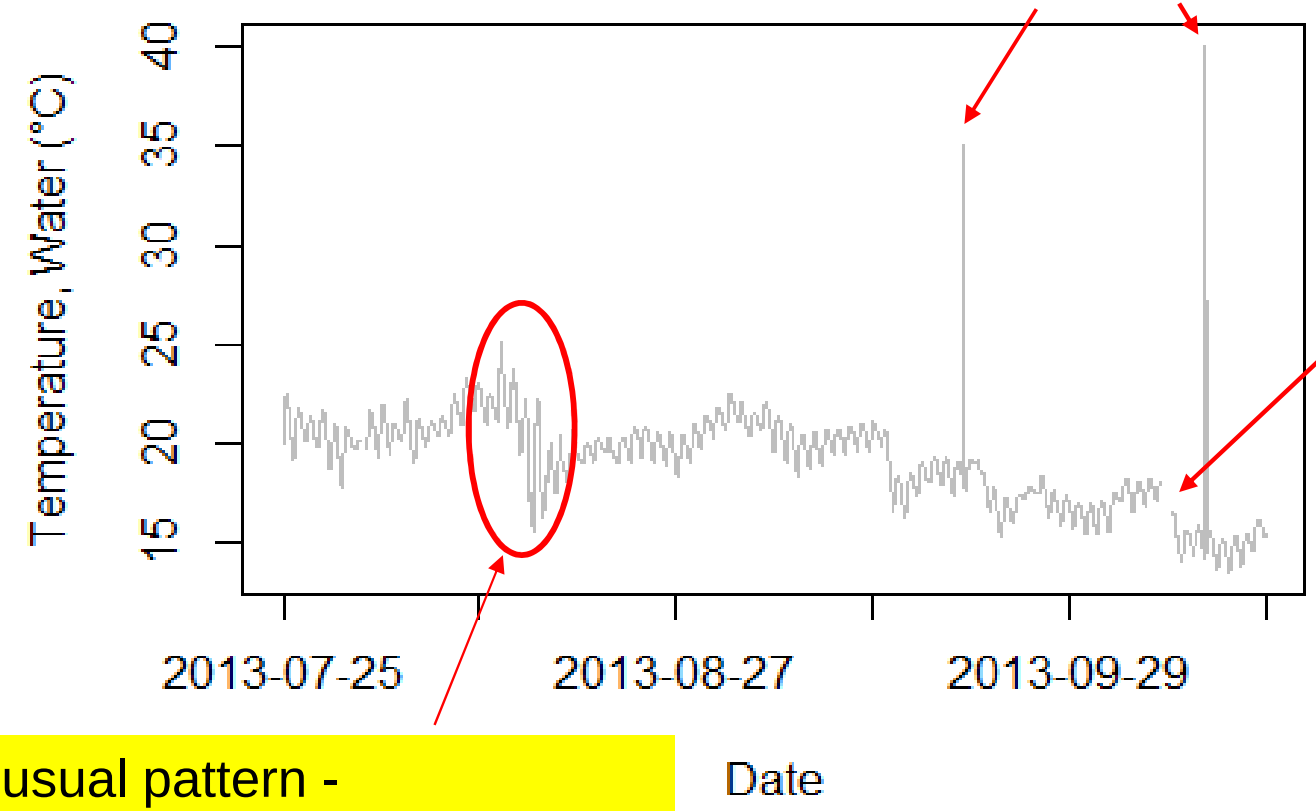


- **Missing data**
- **Closely matching water and air temperature** values (this indicates that the stream sensor may have been out of the water)
- **Diel fluxes with flat tops, or "chatter**" in the gage height record (this indicates that the sensor may have been buried in sediment)
- **Water level values of 0** (this could mean that the pressure transducer was dewatered. With vented transducers, another possibility is that moisture got into the cable and caused readings of zero water depth)
- **Water level values that are negative**
- **Outliers or rapidly fluctuating values** (it is possible that the sensor moved, e.g., due to a high flow event or vandalism)
- Values recorded before the sensor was correctly positioned & stabilized, or after the sensor was removed from its location **("trimming")**

You can also do these time series checks on plots generated in Hoboware, Aquarius, etc.

Examples of visual checks on time series plots

Discuss only



Extreme outliers

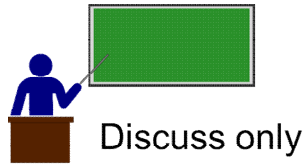
Missing data

Unusual pattern - Dewatered? Looks similar to pattern seen in air temperature measurements

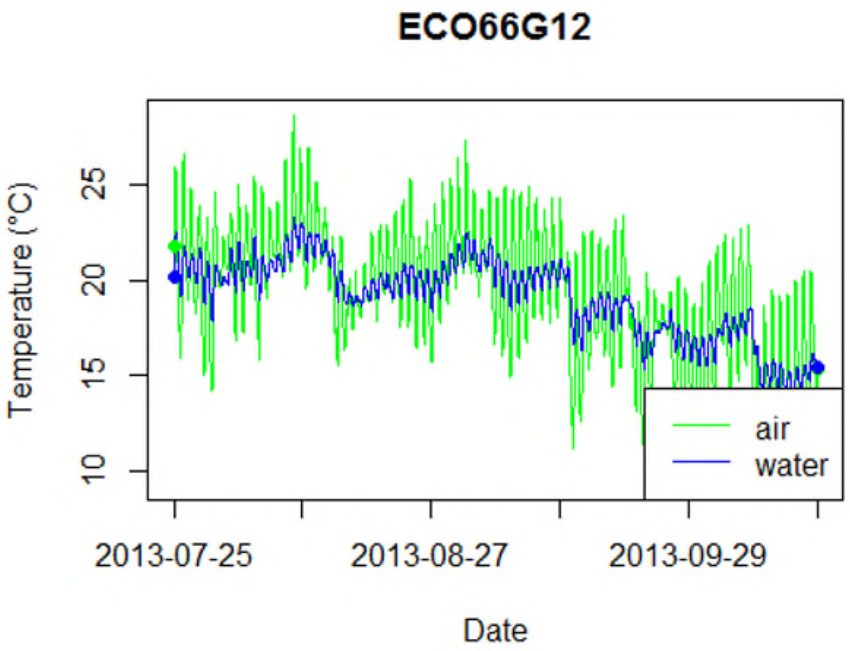
This is an example from a different data file (not the files you are working with)

QC report – Section 4

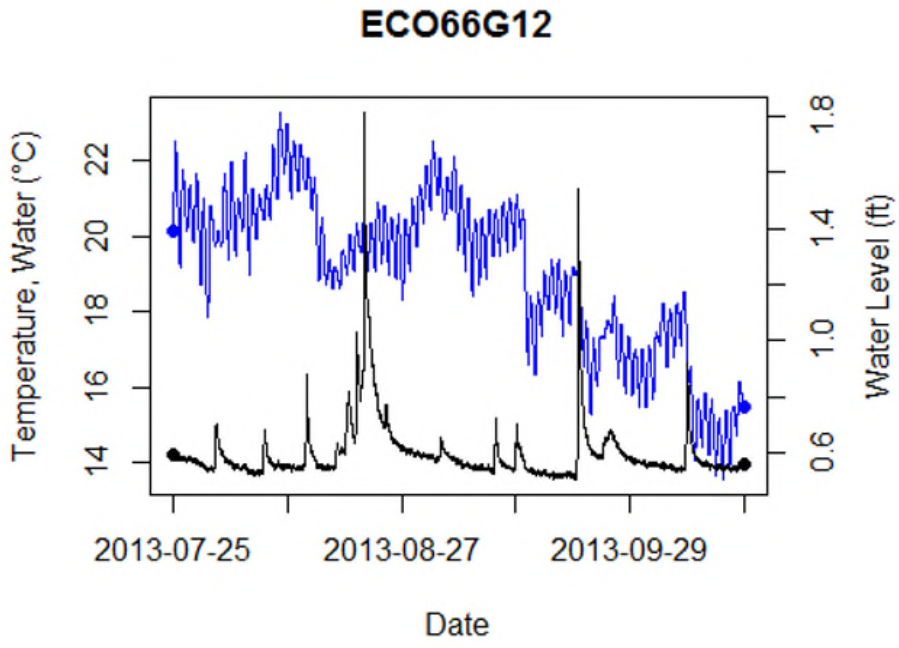
Multi-parameter time series plots



PLOT, Temperature (Air vs. Water)



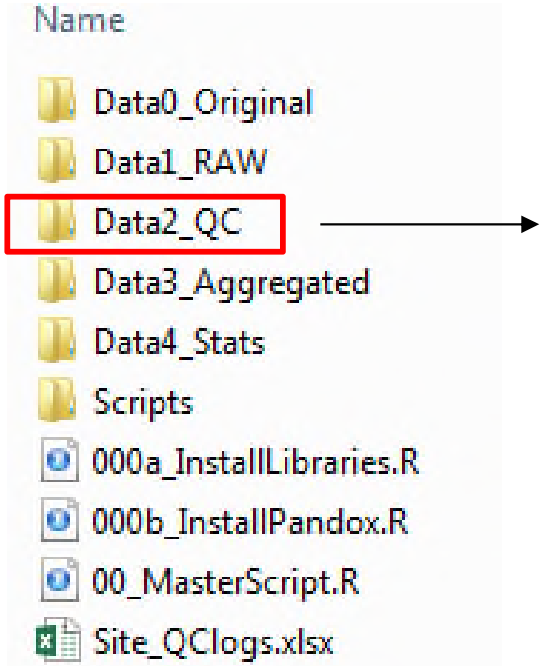
PLOT, Water Level vs. Water Temperature

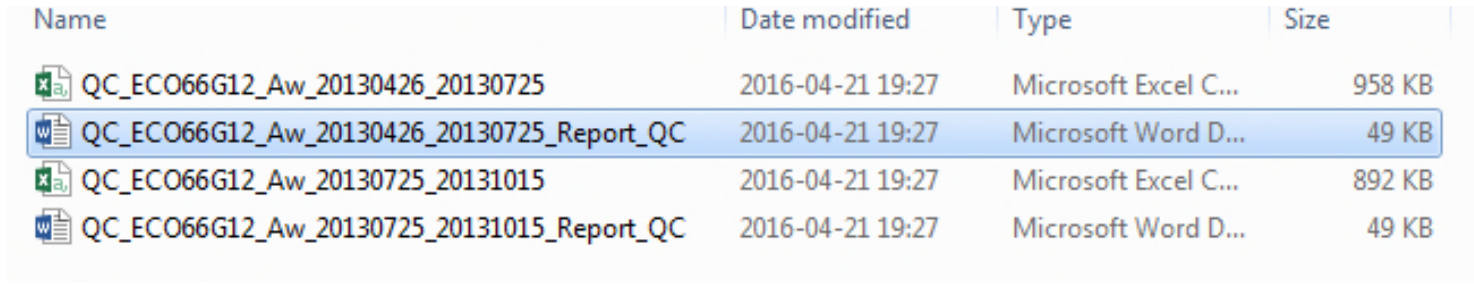


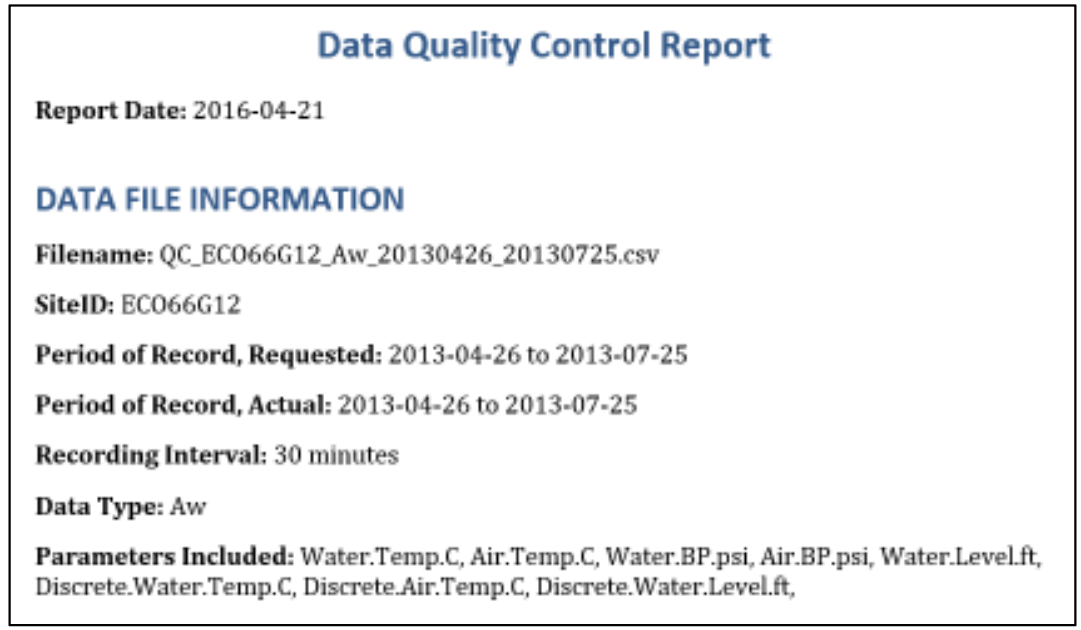
Do the patterns make sense? (e.g., water temperature should fluctuate less than air temperature; increases in water level may correspond with decreases in water temperature)

Your turn!
Open one of the Word documents & scroll through the QC report

Any questions? Comments?







Data Quality Control Report

Report Date: 2016-04-21

DATA FILE INFORMATION

Filename: QC_ECO66G12_Aw_20130426_20130725.csv

SiteID: ECO66G12

Period of Record, Requested: 2013-04-26 to 2013-07-25

Period of Record, Actual: 2013-04-26 to 2013-07-25

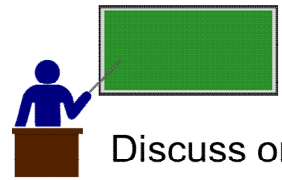
Recording Interval: 30 minutes

Data Type: Aw

Parameters Included: Water.Temp.C, Air.Temp.C, Water.BP.psi, Air.BP.psi, Water.Level.ft, Discrete.Water.Temp.C, Discrete.Air.Temp.C, Discrete.Water.Level.ft,

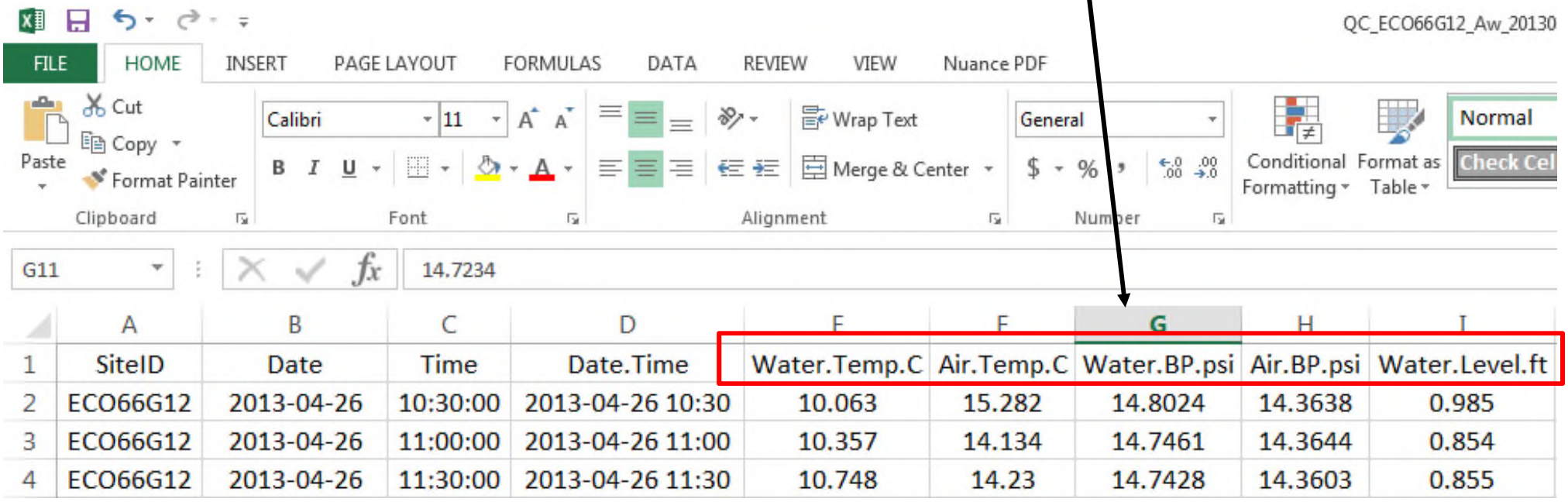
Now let's go through the other QC file…

.csv file



Discuss only

Values (water temp, air temp, water level, etc.)



	A	B	C	D	F	F	G	H	I
1	SiteID	Date	Time	Date.Time	Water.Temp.C	Air.Temp.C	Water.BP.psi	Air.BP.psi	Water.Level.ft
2	ECO66G12	2013-04-26	10:30:00	2013-04-26 10:30	10.063	15.282	14.8024	14.3638	0.985
3	ECO66G12	2013-04-26	11:00:00	2013-04-26 11:00	10.357	14.134	14.7461	14.3644	0.854
4	ECO66G12	2013-04-26	11:30:00	2013-04-26 11:30	10.748	14.23	14.7428	14.3603	0.855

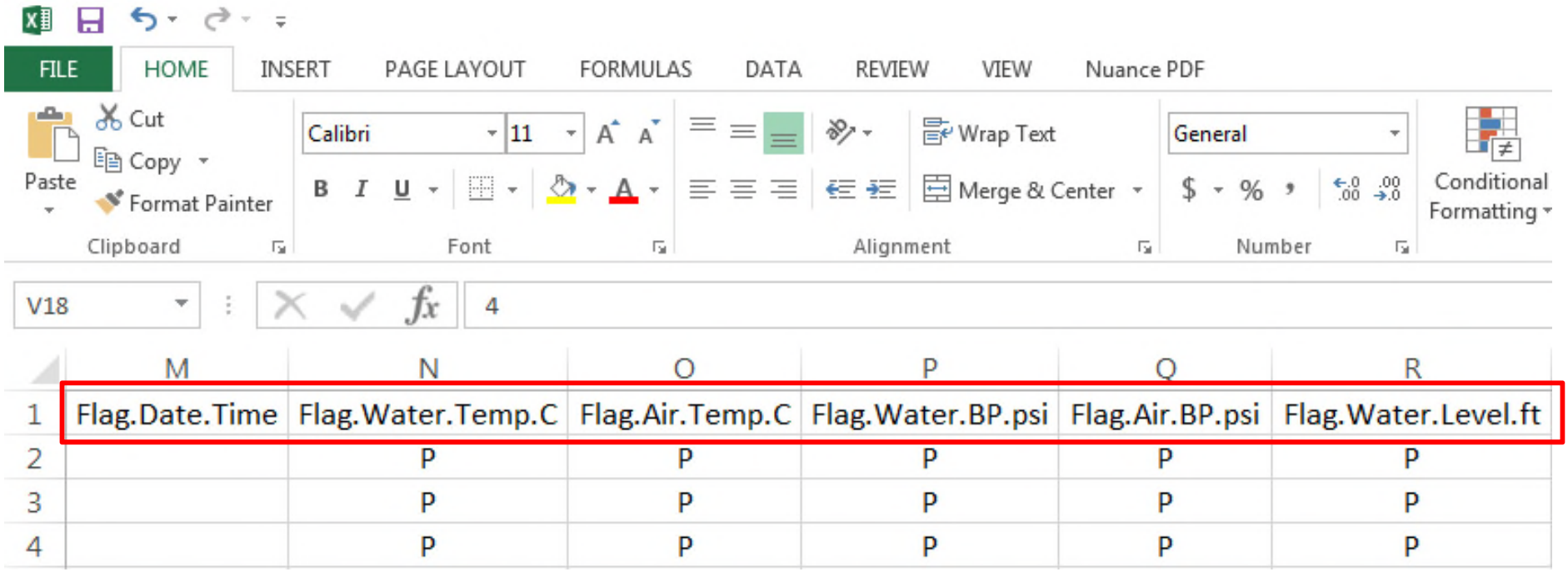
The R scripts add in many additional fields – flags (overall & each parameter/QC test), plus RAW and Comment fields

Scrolling to the right…

.csv file continued…

Discuss only

Overall flags (entries marked as pass, suspect, fail or missing data, based on the *worst* flag across all of the QC tests).



	M	N	O	P	Q	R
1	Flag.Date.Time	Flag.Water.Temp.C	Flag.Air.Temp.C	Flag.Water.BP.psi	Flag.Air.BP.psi	Flag.Water.Level.ft
2		P	P	P	P	P
3		P	P	P	P	P
4		P	P	P	P	P

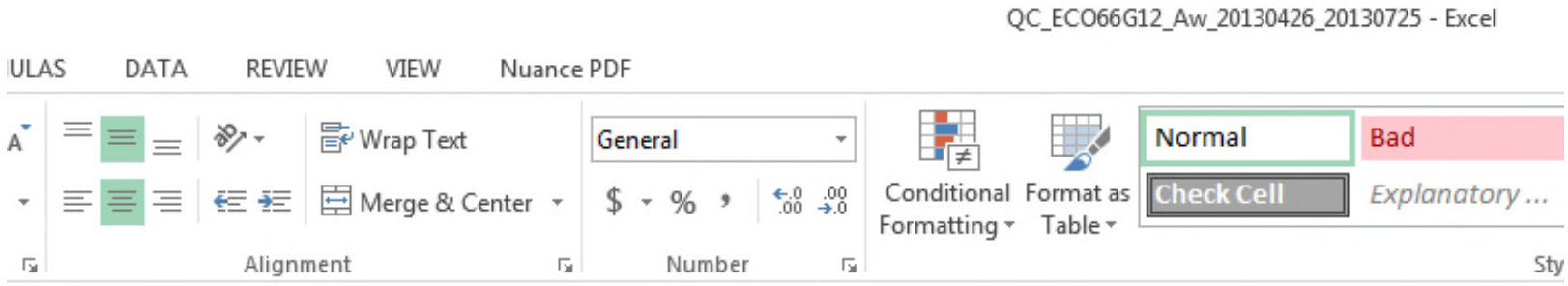
F = fail
P = pass
S = suspect
X = missing data/blank

Continue scrolling to the right

.csv file continued…

Discuss only

Flags for each parameter and each QC test (gross, spike, RoC, flat)



X	Y	Z	AA
Flag.Gross.Water.Temp.C	Flag.Spike.Water.Temp.C	Flag.RoC.Water.Temp.C	Flag.Flat.Water.Temp.C
P	X	X	P
P	P	P	P
P	P	P	P
P	P	P	P
P	P	P	P

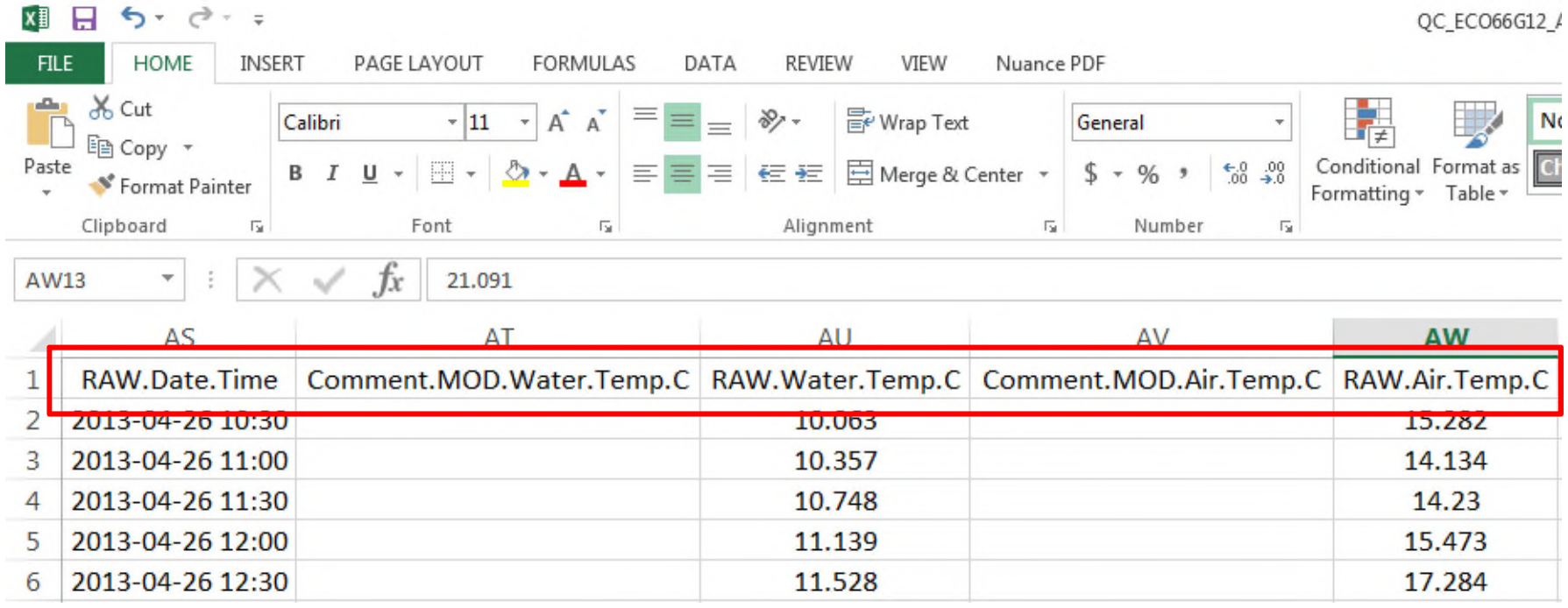
Example - water temperature

Continue scrolling to the right

.csv file continued…

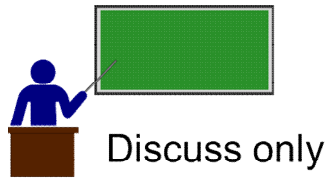
Discuss only

RAW = original values; if you end up making corrections in the primary value fields (in this example, Columns E-I), the RAW fields allow you to retain the original records and track any changes you make.
COMMENT.MOD = allows you to comment on any changes you make.



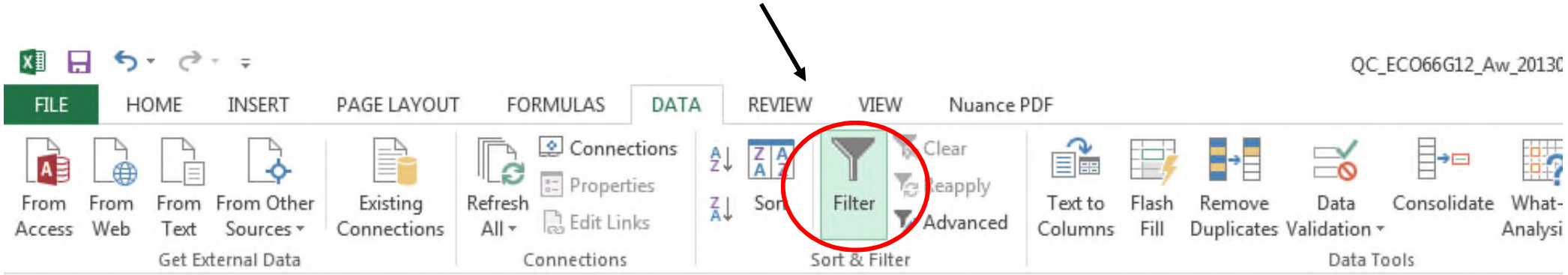
	AS	AT	AU	AV	AW
1	RAW.Date.Time	Comment.MOD.Water.Temp.C	RAW.Water.Temp.C	Comment.MOD.Air.Temp.C	RAW.Air.Temp.C
2	2013-04-26 10:30		10.063		15.282
3	2013-04-26 11:00		10.357		14.134
4	2013-04-26 11:30		10.748		14.23
5	2013-04-26 12:00		11.139		15.473
6	2013-04-26 12:30		11.528		17.284

Tips for easier navigation of the .csv files



Because the .csv files have so many columns, they can be difficult to work with.

Using the **Filter function** can help (e.g., you can filter for entries flagged as ‘F’ or ‘S’ and evaluate)



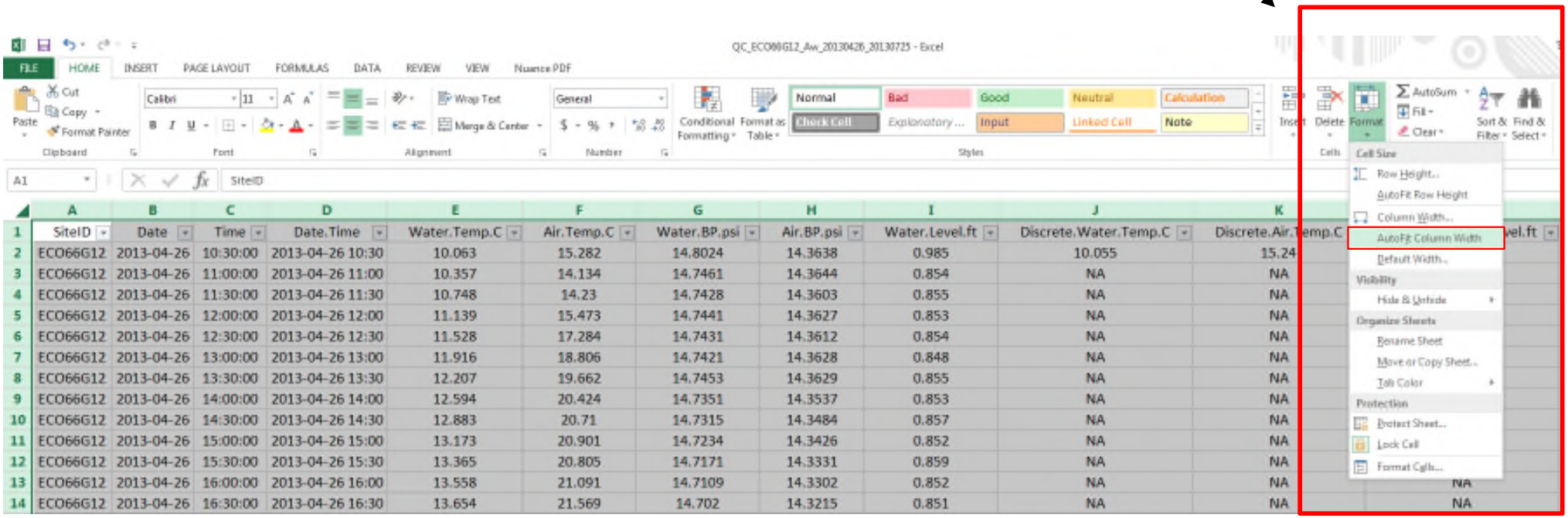
	A	B	C	D	E	F	G	H	I
1	SiteID	Date	Time	Date.Time	Water.Temp.	Air.Temp.	Water.BP.p	Air.BP.p	Water.Level.
2	ECO66G12	2013-04-26	10:30:00	2013-04-26 10:30	10.063	15.282	14.8024	14.3638	0.985
3	ECO66G12	2013-04-26	11:00:00	2013-04-26 11:00	10.357	14.134	14.7461	14.3644	0.854
4	ECO66G12	2013-04-26	11:30:00	2013-04-26 11:30	10.748	14.23	14.7428	14.3603	0.855

Tips for easier navigation of the .csv files

Discuss only

Using the **Autofit Column Width** function can also help.

Highlight all the fields, then click **Format – Autofit Column Width**. Excel then automatically adjusts column widths so that you can see the full field names.



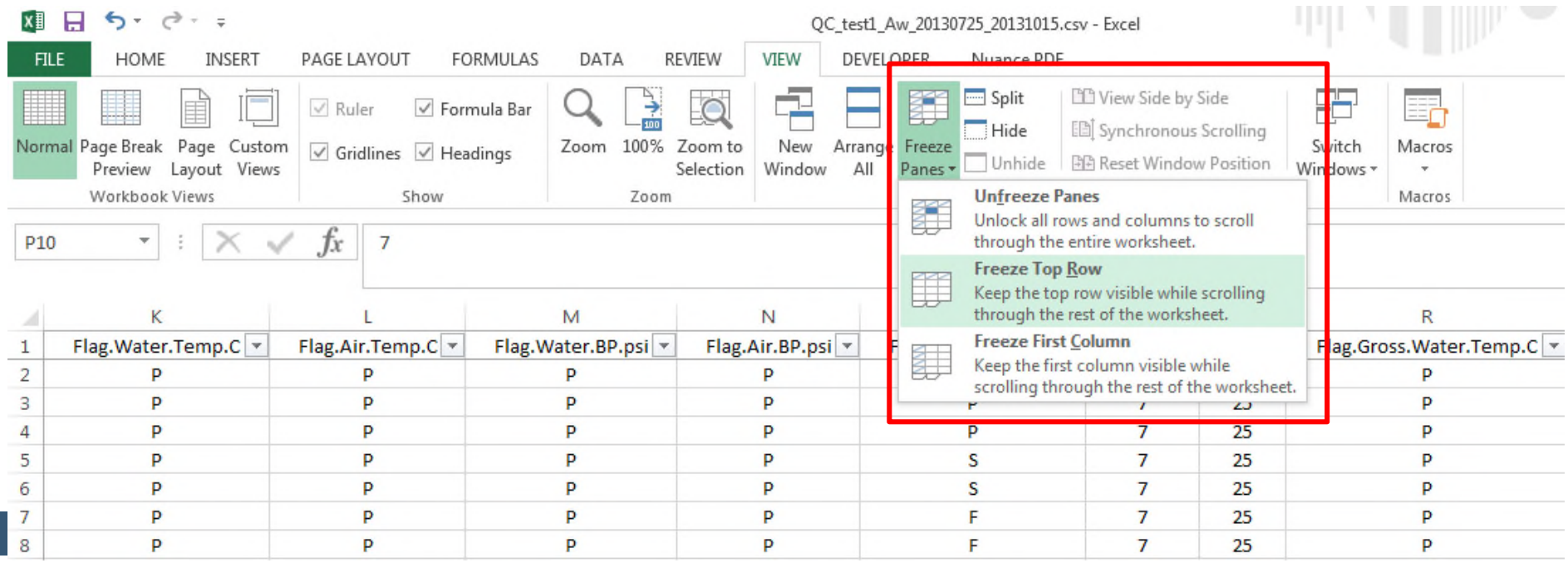
	A	B	C	D	E	F	G	H	I	J	K
1	SiteID	Date	Time	Date.Time	Water.Temp.C	Air.Temp.C	Water.BP.psi	Air.BP.psi	Water.Level.ft	Discrete.Water.Temp.C	Discrete.Air.Temp.C
2	ECO66G12	2013-04-26	10:30:00	2013-04-26 10:30	10.063	15.282	14.8024	14.3638	0.985	10.055	15.24
3	ECO66G12	2013-04-26	11:00:00	2013-04-26 11:00	10.357	14.134	14.7461	14.3644	0.854	NA	NA
4	ECO66G12	2013-04-26	11:30:00	2013-04-26 11:30	10.748	14.23	14.7428	14.3603	0.855	NA	NA
5	ECO66G12	2013-04-26	12:00:00	2013-04-26 12:00	11.139	15.473	14.7441	14.3627	0.853	NA	NA
6	ECO66G12	2013-04-26	12:30:00	2013-04-26 12:30	11.528	17.284	14.7431	14.3612	0.854	NA	NA
7	ECO66G12	2013-04-26	13:00:00	2013-04-26 13:00	11.916	18.806	14.7421	14.3628	0.848	NA	NA
8	ECO66G12	2013-04-26	13:30:00	2013-04-26 13:30	12.207	19.662	14.7453	14.3629	0.855	NA	NA
9	ECO66G12	2013-04-26	14:00:00	2013-04-26 14:00	12.594	20.424	14.7351	14.3537	0.853	NA	NA
10	ECO66G12	2013-04-26	14:30:00	2013-04-26 14:30	12.883	20.71	14.7315	14.3484	0.857	NA	NA
11	ECO66G12	2013-04-26	15:00:00	2013-04-26 15:00	13.173	20.901	14.7234	14.3426	0.852	NA	NA
12	ECO66G12	2013-04-26	15:30:00	2013-04-26 15:30	13.365	20.805	14.7171	14.3331	0.859	NA	NA
13	ECO66G12	2013-04-26	16:00:00	2013-04-26 16:00	13.558	21.091	14.7109	14.3302	0.852	NA	NA
14	ECO66G12	2013-04-26	16:30:00	2013-04-26 16:30	13.654	21.569	14.702	14.3215	0.851	NA	NA

Tips for easier navigation of the .csv files

Discuss only

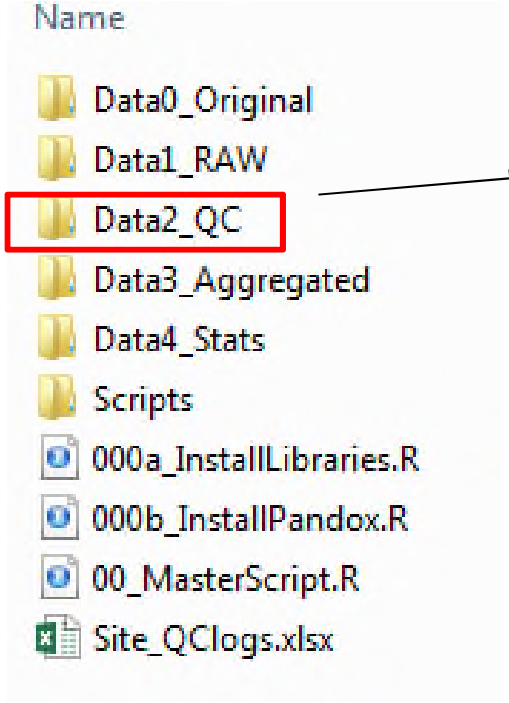
You can also **Freeze Panes – Freeze Top Row** to make viewing easier (this allows you to view the column headings while you scroll down).

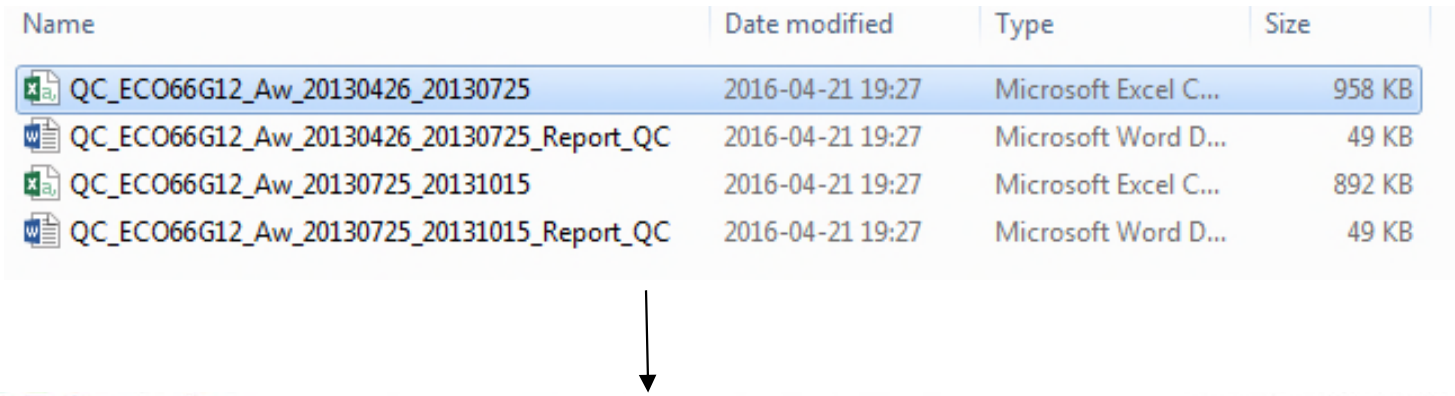
You can also use the **Split** function (this allows you to split the screen so that you can see the column headings while scrolling down, and the left-hand columns (e.g., date-time field) while scrolling to the right.

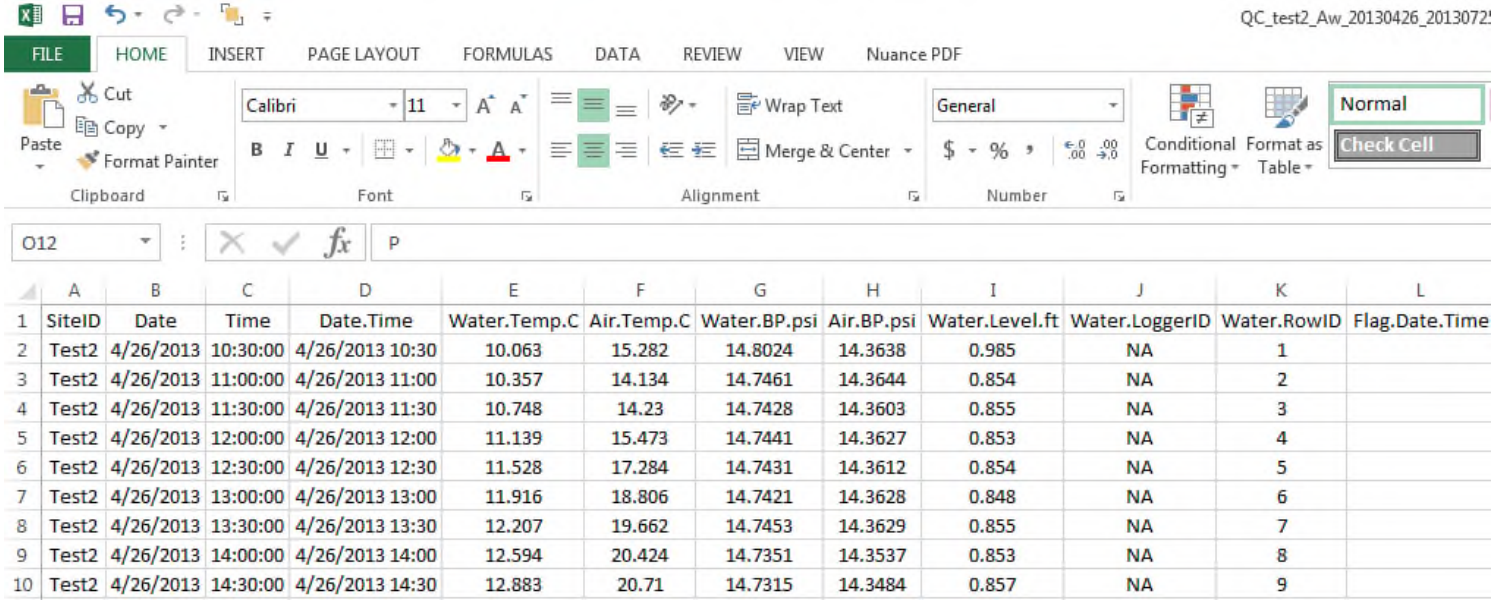


Your turn!
Open & scroll through a .csv file

Any questions? Comments?







QC_test2_Aw_20130426_20130725

FILE | HOME | INSERT | PAGE LAYOUT | FORMULAS | DATA | REVIEW | VIEW | Nuance PDF

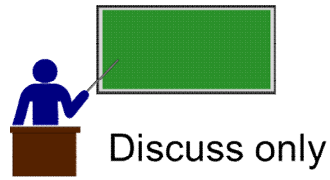
Cut, Copy, Format Painter, Paste, Calibri, 11, Wrap Text, General, Merge & Center, Conditional Formatting, Format as Table, Normal, Check Cell

Clipboard | Font | Alignment | Number

O12 | P

	A	B	C	D	E	F	G	H	I	J	K	L
1	SiteID	Date	Time	Date.Time	Water.Temp.C	Air.Temp.C	Water.BP.psi	Air.BP.psi	Water.Level.ft	Water.LoggerID	Water.RowID	Flag.Date.Time
2	Test2	4/26/2013	10:30:00	4/26/2013 10:30	10.063	15.282	14.8024	14.3638	0.985	NA	1	
3	Test2	4/26/2013	11:00:00	4/26/2013 11:00	10.357	14.134	14.7461	14.3644	0.854	NA	2	
4	Test2	4/26/2013	11:30:00	4/26/2013 11:30	10.748	14.23	14.7428	14.3603	0.855	NA	3	
5	Test2	4/26/2013	12:00:00	4/26/2013 12:00	11.139	15.473	14.7441	14.3627	0.853	NA	4	
6	Test2	4/26/2013	12:30:00	4/26/2013 12:30	11.528	17.284	14.7431	14.3612	0.854	NA	5	
7	Test2	4/26/2013	13:00:00	4/26/2013 13:00	11.916	18.806	14.7421	14.3628	0.848	NA	6	
8	Test2	4/26/2013	13:30:00	4/26/2013 13:30	12.207	19.662	14.7453	14.3629	0.855	NA	7	
9	Test2	4/26/2013	14:00:00	4/26/2013 14:00	12.594	20.424	14.7351	14.3537	0.853	NA	8	
10	Test2	4/26/2013	14:30:00	4/26/2013 14:30	12.883	20.71	14.7315	14.3484	0.857	NA	9	

Corrections

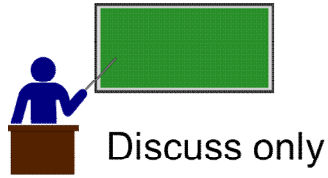


In general, 3 possible actions can be taken:

- Leave data as is
- Apply correction factor
- Remove data

Corrections should not be made unless the cause(s) of error(s) can be validated or explained in the field notes or by comparison with information from nearby stations. Accurate field notes and accuracy check logs are essential in the data correction process. Any discrepancies should be documented in your data file and any actions you take should also be carefully documented.

Corrections



Right now we **don't have any automated corrections programed into the scripts** (this is a future 'wish list' item).

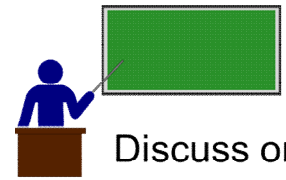
Any corrections you make need to be done **manually**.

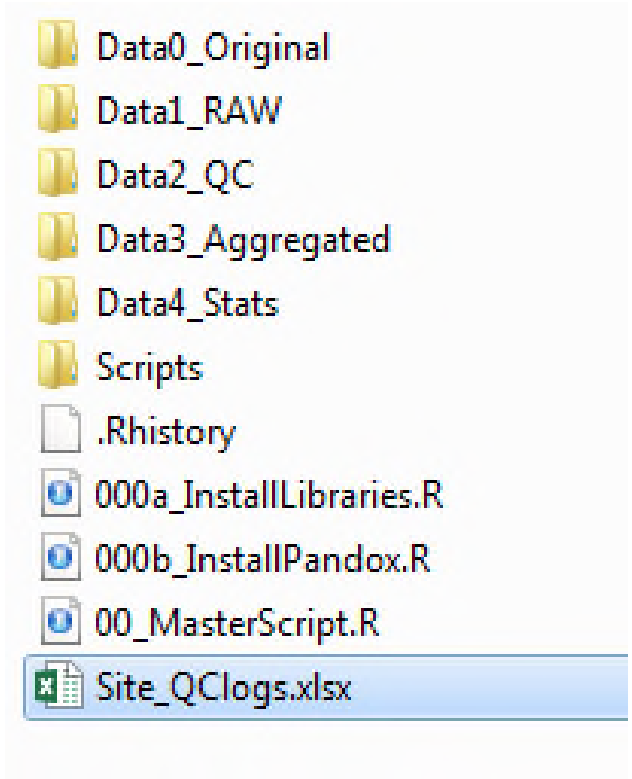
Some types of corrections are fairly straightforward and can be done in Excel (e.g., change values by same amount, or basic drift corrections); others are more complex.

Potential references –

- Wagner et al. 2006 (USGS) (more detailed)
- 'Best Practices' report (more general - has a table with a general summary of different types of problems that can occur (e.g., missing data, failed accuracy check) and recommended actions for addressing them)

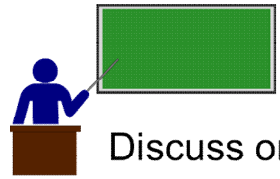
Site_QClogs Excel file

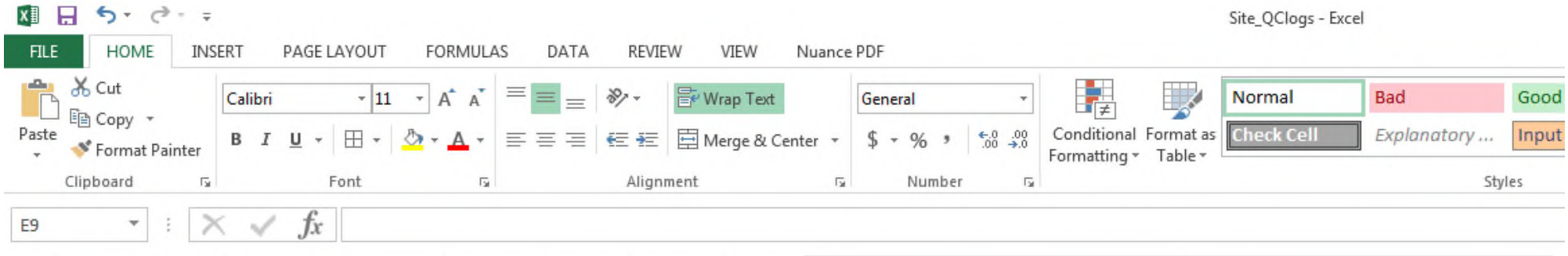




We created a QC log file – this is a place where you can log corrections, dates of deployment, sensor replacement, etc. (anything that can affect the quality of the data, or might explain unusual changes in the data)

Site_QClogs Excel file





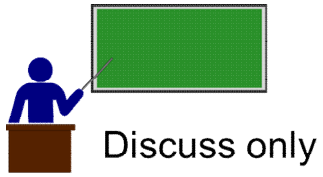
	A	B	C	D	E
1	**SiteID**	**Stream name**	**Person making entry**	**Date**	**Action**
2	ECO66G12	Sheeds Creek	JS	2013-04-26	deployed Hobo U20 non-vented pressure transducers (one on land, on in the water), water S/N 10229557, long -72.38454, lat 42.46471; air S/N 10229557, long -72.38450, lat 42.46470
3	ECO66G12	Sheeds Creek	JS	2013-07-25	site visit, downloaded data; maintenance check - everything looked good!
4	ECO66G12	Sheeds Creek	JS	2013-10-15	site visit, downloaded data; maintenance check - cleaned the sensor
5	ECO66G12	Sheeds Creek	JS	2013-11-15	ran QC checks on files from 4/26-10/15/2013; water temp - deleted 2 extreme outliers; aggregated

Example entries…this is still a work in progress…we welcome your comments and suggestions!

EPA
United States
Environmental Protection
Agency

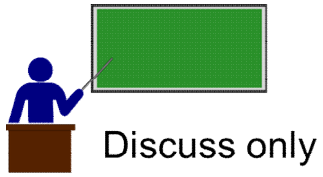
Running the Aggregate script

Aggregate process - overview



1. **Update entries in the MasterScript** (in this example, you only need to change 1 entry - Operation needs to be changed from [2] to [3])
2. **Run the script**, which searches the Data2_QC folder for the appropriate files and, in this example, combines the 2 QC'd files into a single file that spans the full date range (April 26 through October 15)
3. Open the **Data3_Aggregated folder**. There should be 2 outputs for each input file:
    - .**csv** file
    - **QC report** (Word document)

1. Update entries in the MasterScript

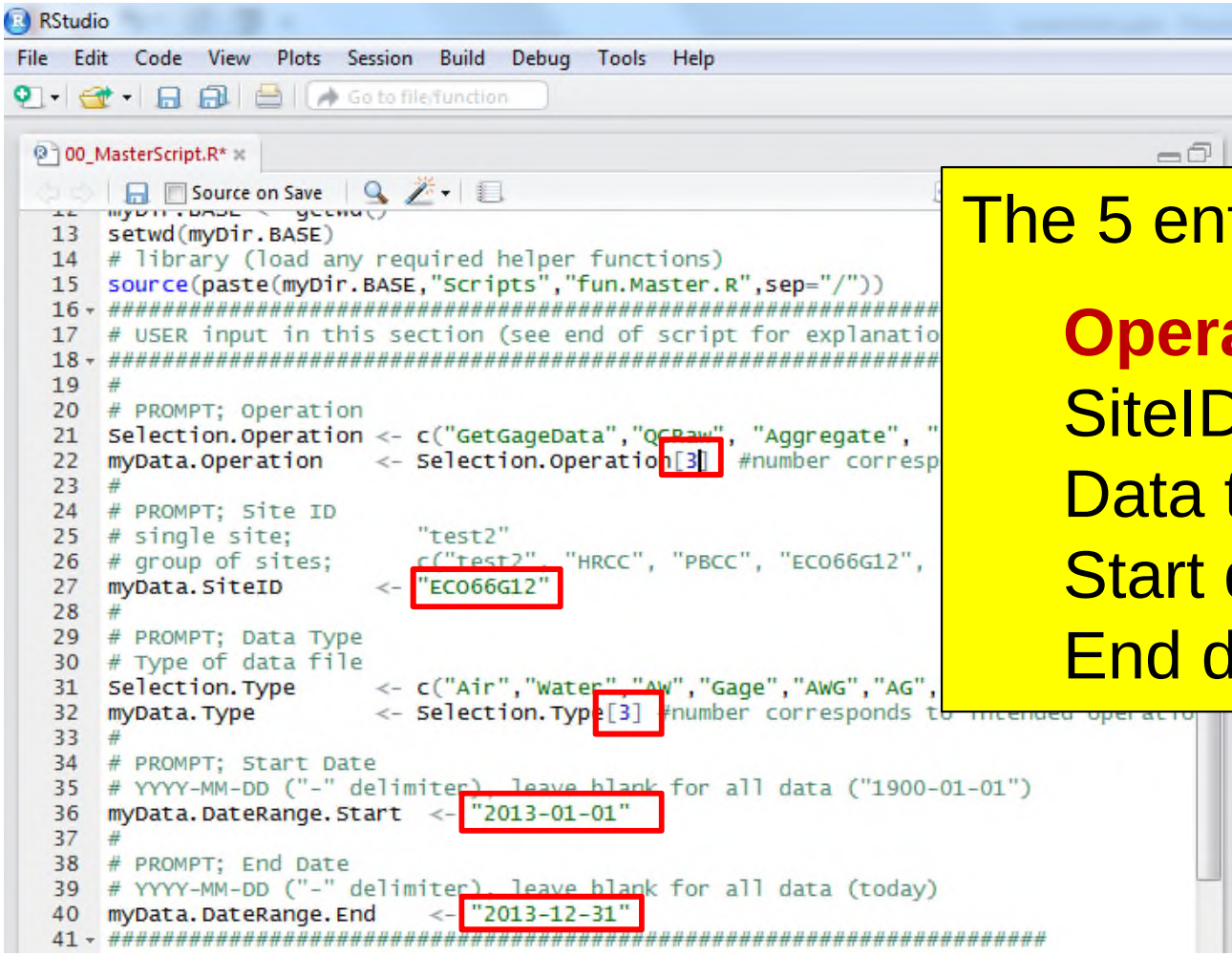


In this example, you only need to change 1 entry -

Operation needs to be changed from [2] to [3]

Selection.Operation
[1] = GetGageData
[2] = QCRaw
[3] = Aggregate
[4] = SummaryStats

1. Update entries in the MasterScript

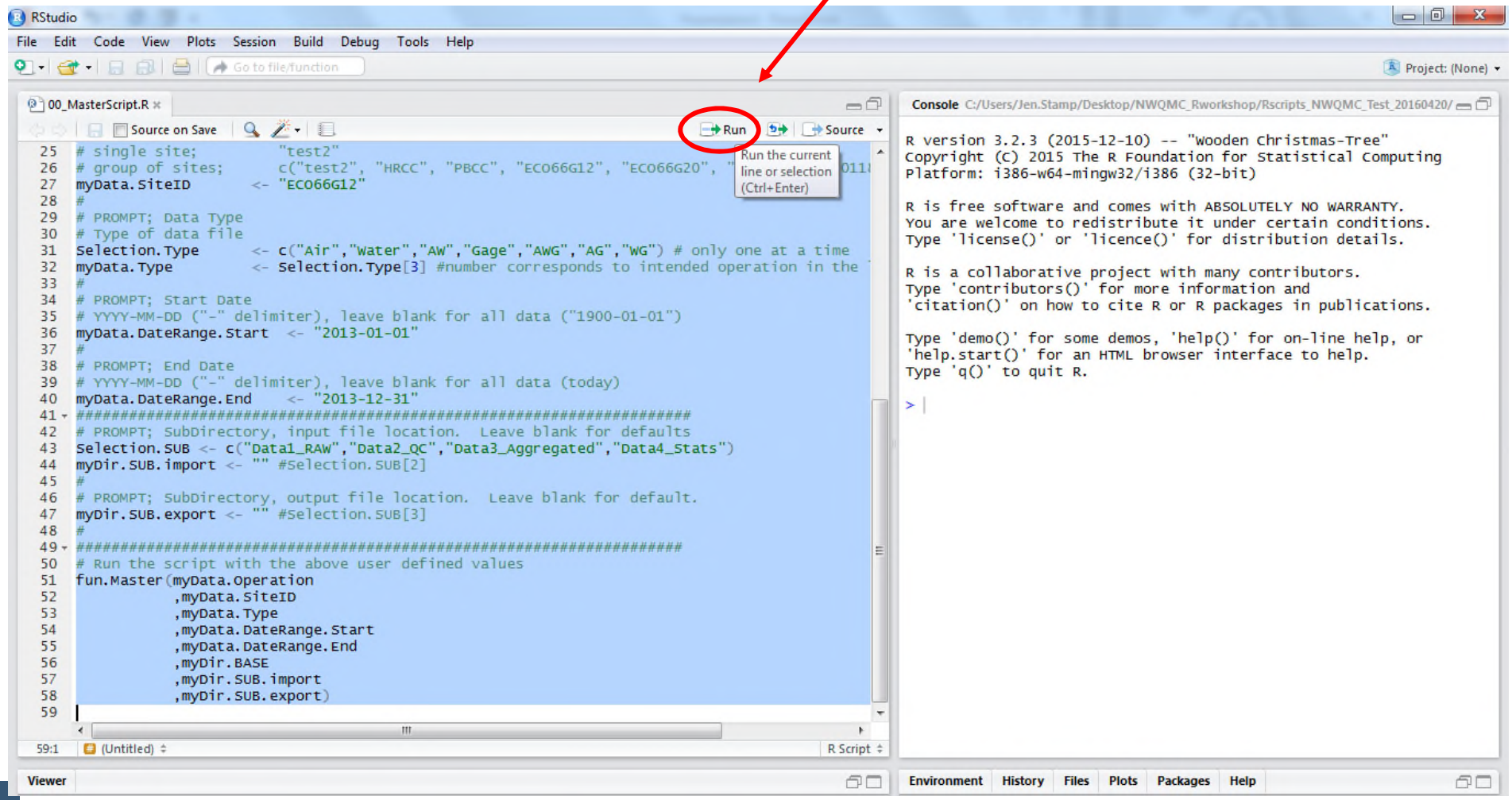


The 5 entries should read –

Operation [3]
SiteID “**ECO66G12**”
Data type [3]
Start date “**2013-01-01**”
End date “**2013-12-31**”

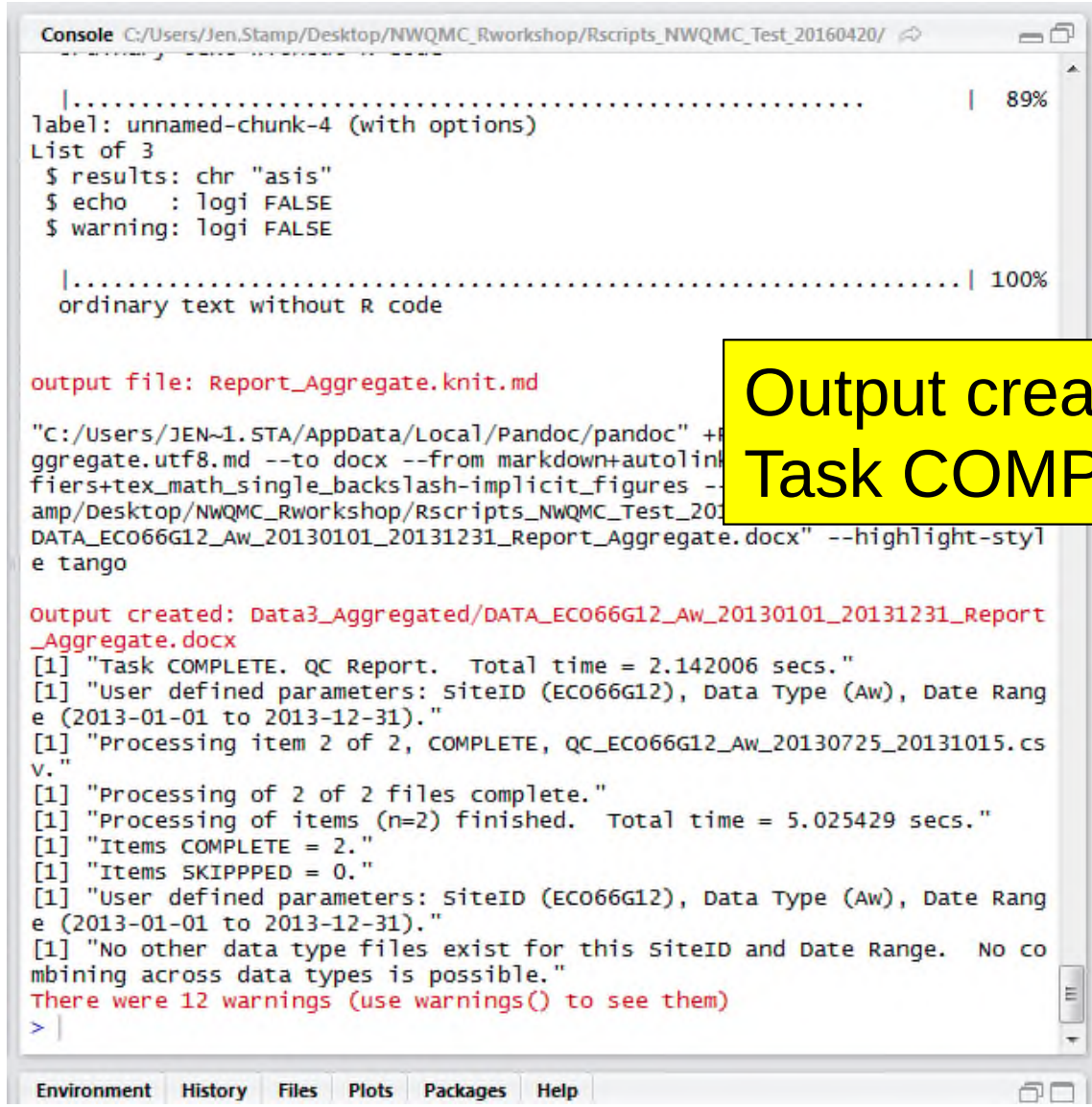
2. Run the script

Using your cursor, highlight the script and click RUN



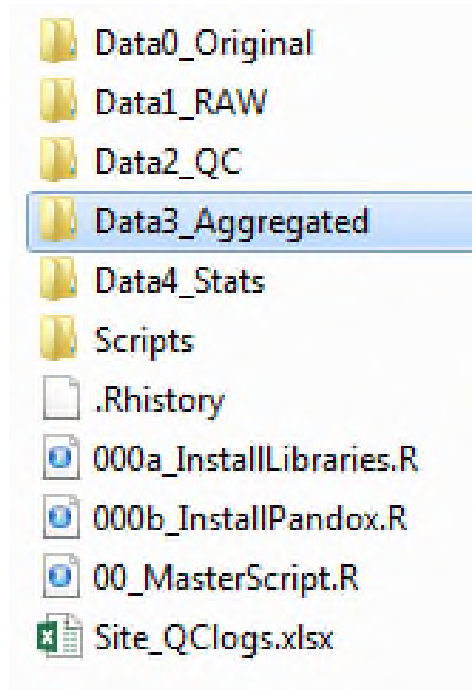
Check the Console (right) panel

If the script ran successfully, you should see a running log like this…



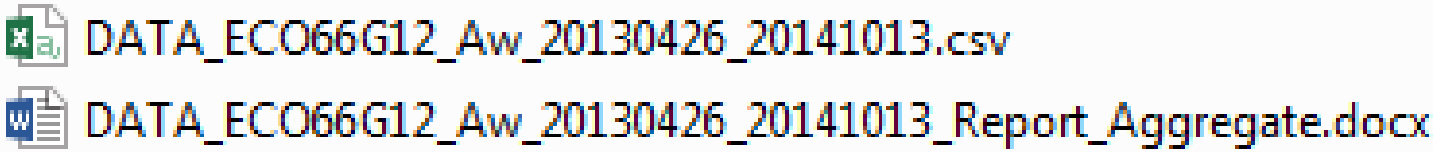
Output created…
Task COMPLETE

3. Open the Data3_Aggregated folder



There should be 2 files –

- **.csv** file (=merged, QC'd files for site and desired time period; all fields in the QC .csv file are carried over)
- **Aggregate report** (Word document) – contains time series plots



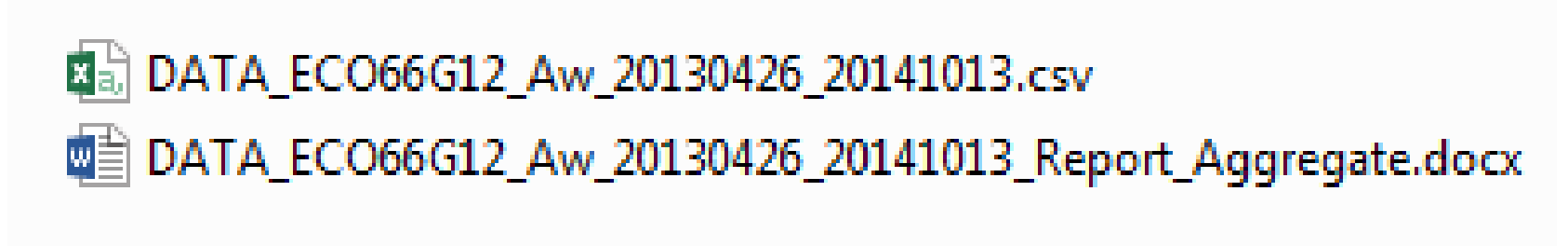
Check the files

Open the .csv file

- Does the file cover the full date range?
- Does everything else look correct?

Open the Aggregate report (Word document)

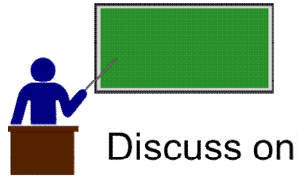
- Do the time series plots look correct?



EPA
United States
Environmental Protection
Agency

Running the SummaryStats script

SummaryStats script



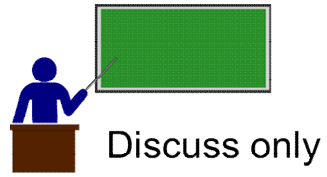
Setting expectations…

Right now the summary script is very basic (this is not going to ‘wow’ you).

The main intent is to get all the data (which come from different sources) QC’d and into the same format.

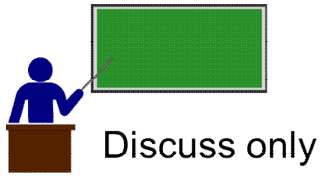
Once that happens, there are many possibilities for better visualization tools, more summary statistics, etc.

SummaryStats process - overview



1. **Update entries in the MasterScript** (in this example, you only need to change 1 entry - Operation needs to be changed from [3] to [4])
2. **Run the script**, which searches the Data3_Aggregated folder for the appropriate files
3. Open the **Data4_Stats folder**. There should be 3 outputs *for each parameter*:
    - **DV** .csv file (daily average)
    - **STATS** .csv file (summary statistics)
    - **PDF** files with various time series plots

1. Update entries in the MasterScript



In this example, you only need to change 1 entry -

Operation needs to be changed from [3] to [4]

Selection.Operation
[1] = GetGageData
[2] = QCRaw
[3] = Aggregate
[4] = SummaryStats

1. Update entries in the MasterScript

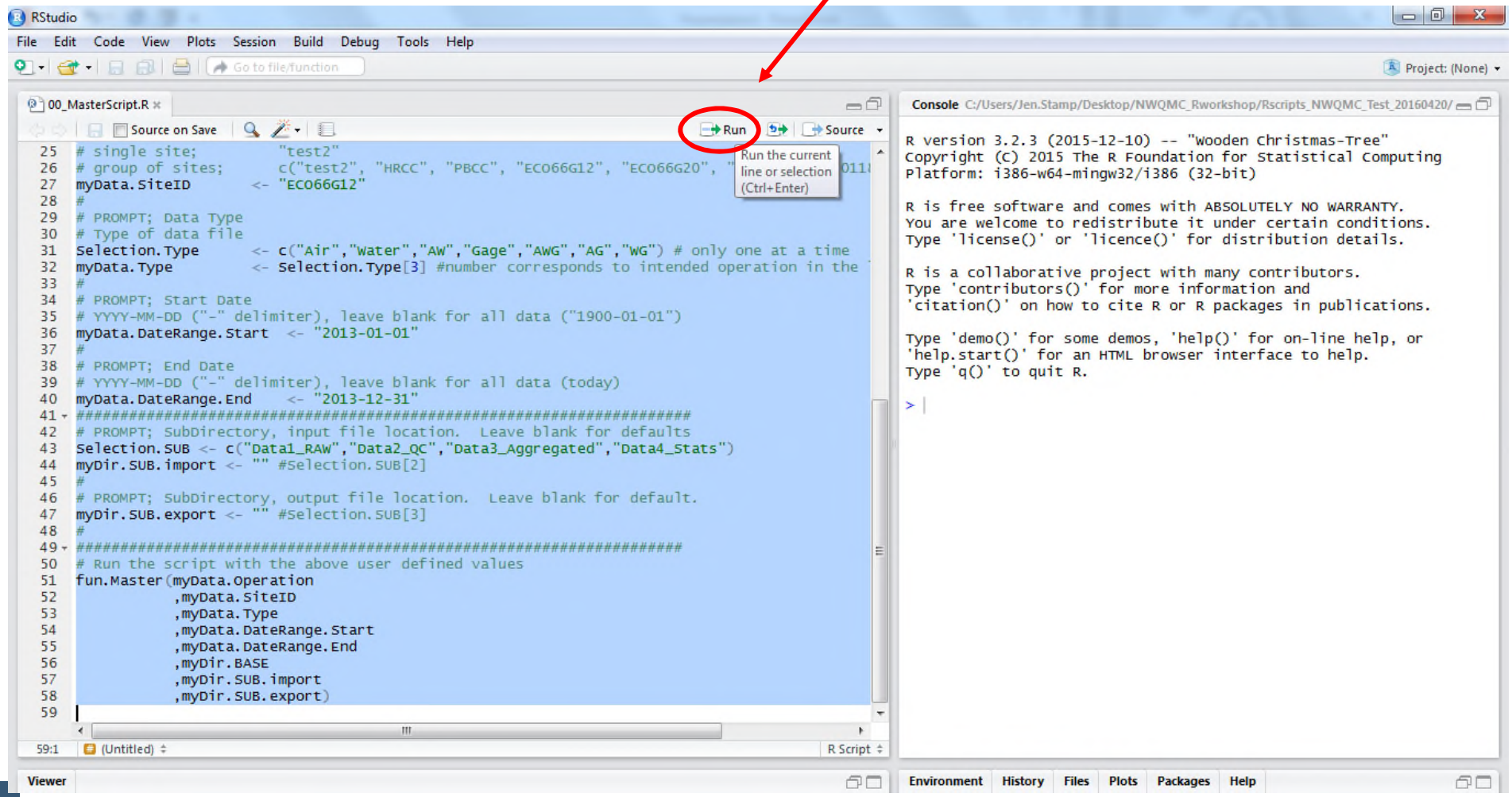
```
16  ##########################################################
17  # USER input in this section (see end of script for ex
18  ##########################################################
19  #
20  # PROMPT; Operation
21  Selection.Operation <- c("GetGageData","QCRaw", "Aggr
22  myData.Operation    <- Selection.Operation[4]  #numbe
23  #
24  # PROMPT; Site ID
25  # single site;          "test2"
26  # group of sites;       c("test2", "HRCC", "PBCC", "EC
27  myData.SiteID       <- 'ECO66G12'
28  #
29  # PROMPT; Data Type
30  # Type of data file
31  Selection.Type      <- c("Air","Water","AW","Gage","A
32  myData.Type         <- Selection.Type[3] #number corresponds to intended operatio
33  #
34  # PROMPT; Start Date
35  # YYYY-MM-DD ("-" delimiter), leave blank for all data ("1900-01-01")
36  myData.DateRange.Start  <- "2013-01-01"
37  #
38  # PROMPT; End Date
39  # YYYY-MM-DD ("-" delimiter), leave blank for all data (today)
40  myData.DateRange.End    <- "2013-12-31"
41  ##########################################################
```

The 5 entries should read –

- **Operation** [4]
- SiteID “ECO66G12”
- Data type [3]
- Start date “2013-01-01”
- End date “2013-12-31”

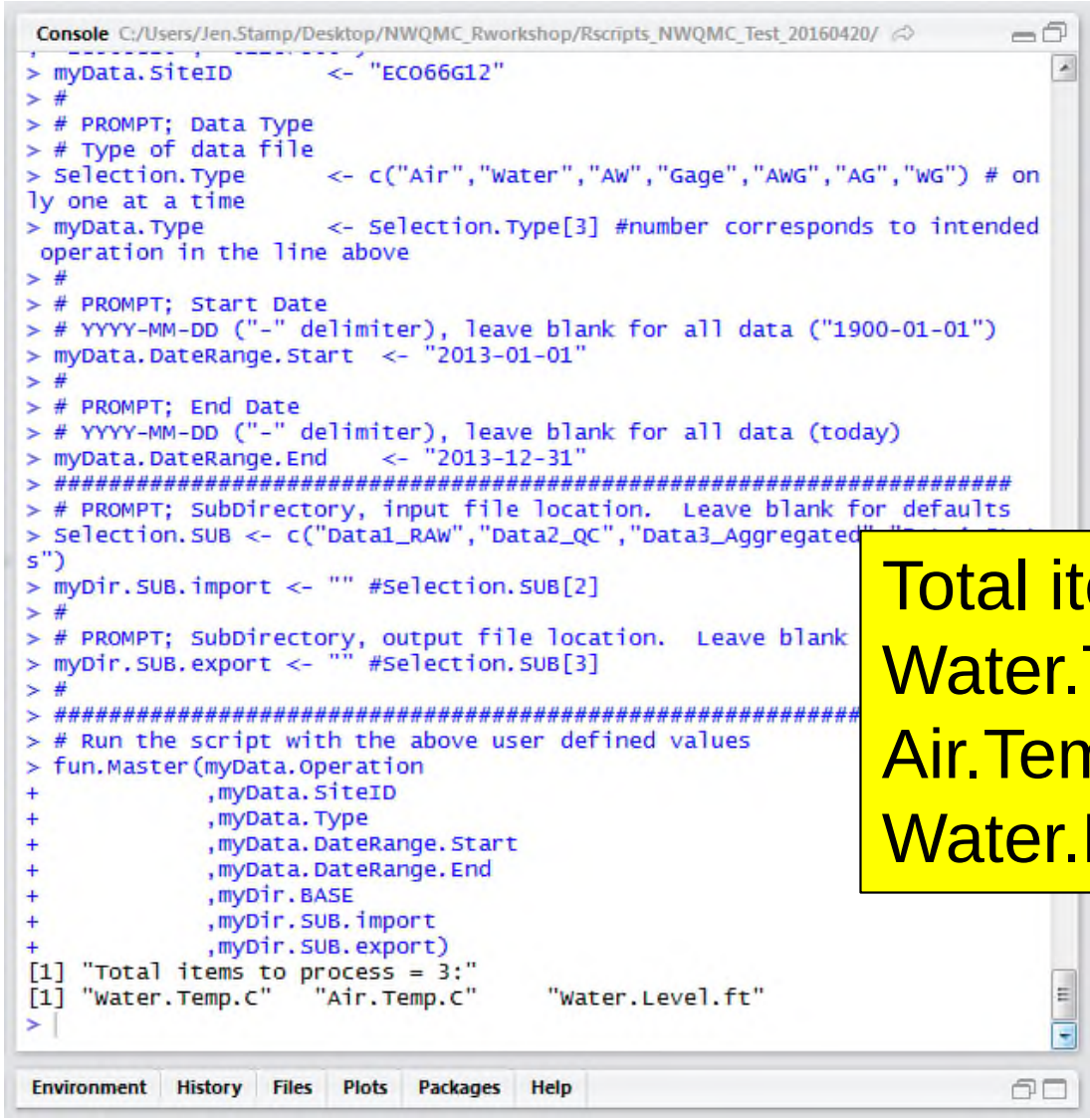
2. Run the script

Using your cursor, highlight the script and click RUN



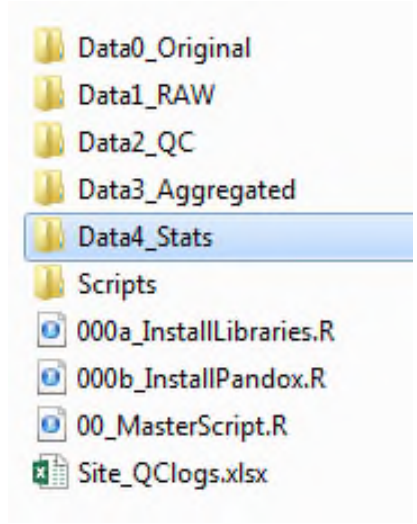
Check the Console (right) panel

If the script ran successfully, you should see a running log like this…



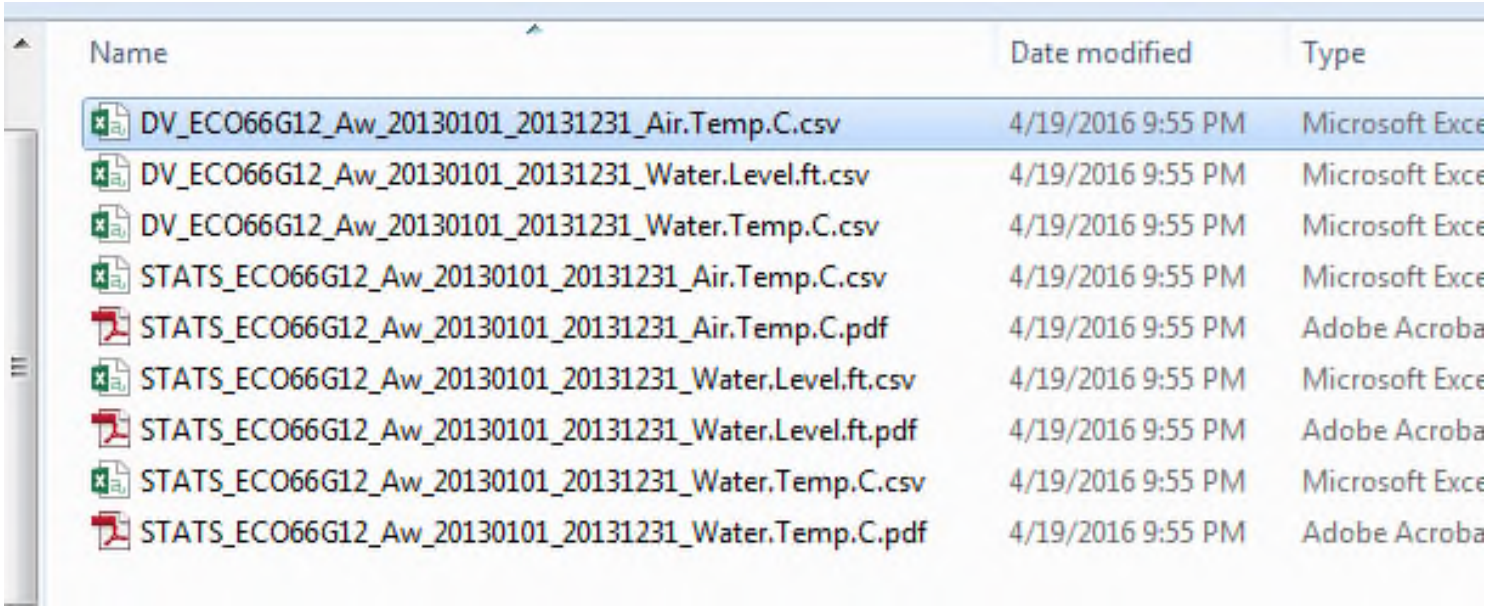
Total items to process = 3
Water.Temp.C
Air.Temp.C
Water.Level.ft

3. Open the Data4_Stats folder



There should be 3 files (each) for air temperature, water temperature and water level*

- **DV** .csv file (daily average)
- **STATS** .csv file (summary statistics for different time periods)
- **PDF** files with various time series plots



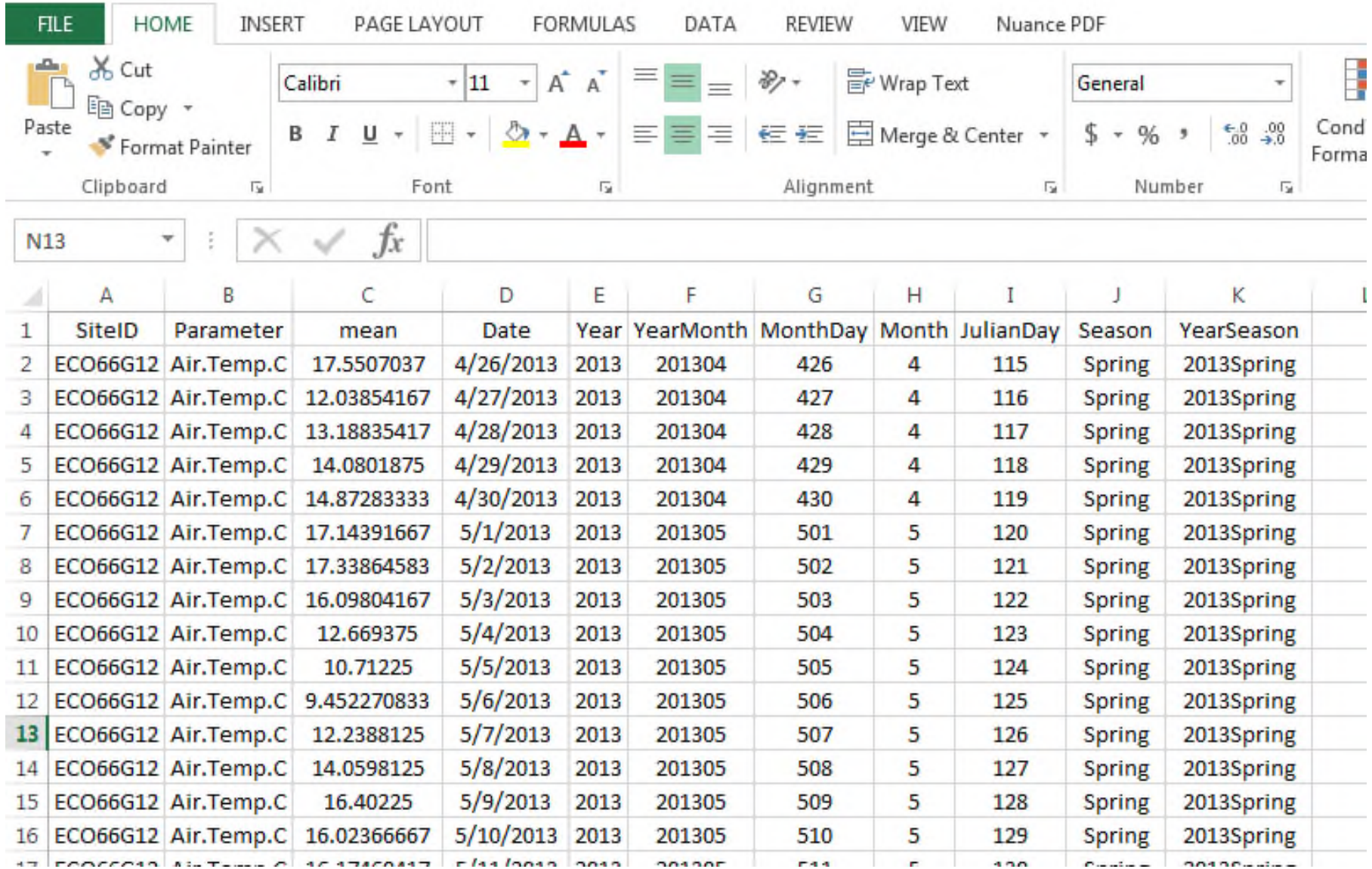
Name	Date modified	Type
DV_ECO66G12_Aw_20130101_20131231_Air.Temp.C.csv	4/19/2016 9:55 PM	Microsoft Exce
DV_ECO66G12_Aw_20130101_20131231_Water.Level.ft.csv	4/19/2016 9:55 PM	Microsoft Exce
DV_ECO66G12_Aw_20130101_20131231_Water.Temp.C.csv	4/19/2016 9:55 PM	Microsoft Exce
STATS_ECO66G12_Aw_20130101_20131231_Air.Temp.C.csv	4/19/2016 9:55 PM	Microsoft Exce
STATS_ECO66G12_Aw_20130101_20131231_Air.Temp.C.pdf	4/19/2016 9:55 PM	Adobe Acroba
STATS_ECO66G12_Aw_20130101_20131231_Water.Level.ft.csv	4/19/2016 9:55 PM	Microsoft Exce
STATS_ECO66G12_Aw_20130101_20131231_Water.Level.ft.pdf	4/19/2016 9:55 PM	Adobe Acroba
STATS_ECO66G12_Aw_20130101_20131231_Water.Temp.C.csv	4/19/2016 9:55 PM	Microsoft Exce
STATS_ECO66G12_Aw_20130101_20131231_Water.Temp.C.pdf	4/19/2016 9:55 PM	Adobe Acroba

*we're working on adding in discharge and potentially other parameters

SummaryStats - DV (daily) output

Example - air temperature

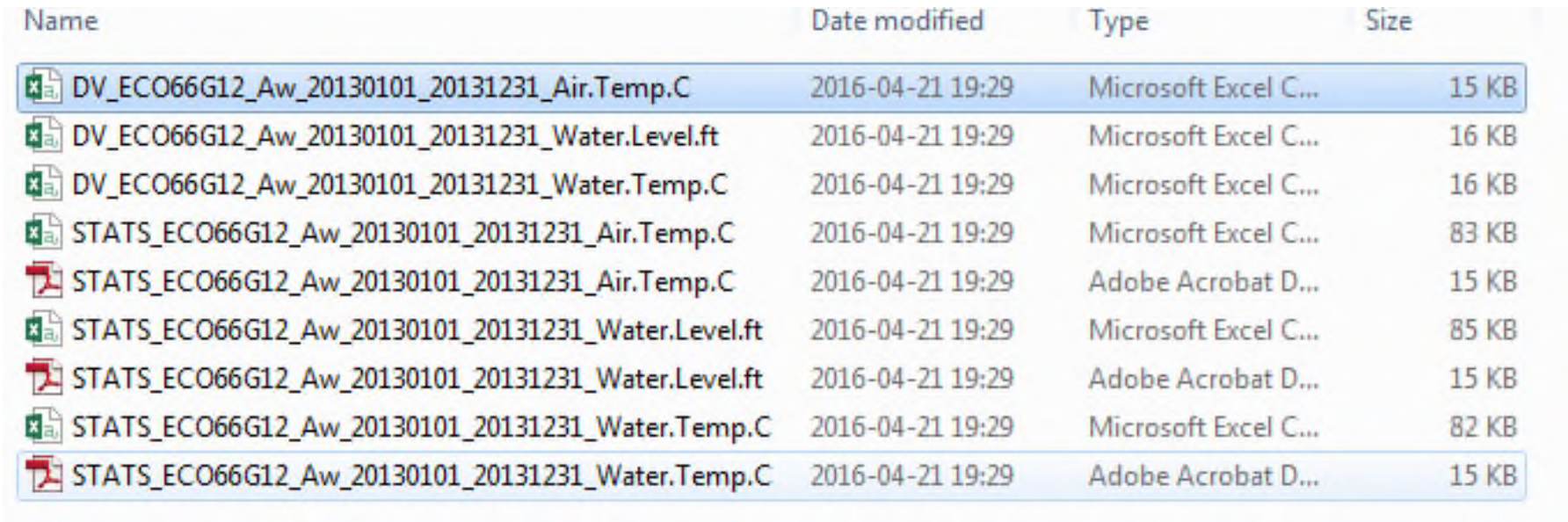
Discuss only



	A	B	C	D	E	F	G	H	I	J	K
1	SiteID	Parameter	mean	Date	Year	YearMonth	MonthDay	Month	JulianDay	Season	YearSeason
2	ECO66G12	Air.Temp.C	17.5507037	4/26/2013	2013	201304	426	4	115	Spring	2013Spring
3	ECO66G12	Air.Temp.C	12.03854167	4/27/2013	2013	201304	427	4	116	Spring	2013Spring
4	ECO66G12	Air.Temp.C	13.18835417	4/28/2013	2013	201304	428	4	117	Spring	2013Spring
5	ECO66G12	Air.Temp.C	14.0801875	4/29/2013	2013	201304	429	4	118	Spring	2013Spring
6	ECO66G12	Air.Temp.C	14.87283333	4/30/2013	2013	201304	430	4	119	Spring	2013Spring
7	ECO66G12	Air.Temp.C	17.14391667	5/1/2013	2013	201305	501	5	120	Spring	2013Spring
8	ECO66G12	Air.Temp.C	17.33864583	5/2/2013	2013	201305	502	5	121	Spring	2013Spring
9	ECO66G12	Air.Temp.C	16.09804167	5/3/2013	2013	201305	503	5	122	Spring	2013Spring
10	ECO66G12	Air.Temp.C	12.669375	5/4/2013	2013	201305	504	5	123	Spring	2013Spring
11	ECO66G12	Air.Temp.C	10.71225	5/5/2013	2013	201305	505	5	124	Spring	2013Spring
12	ECO66G12	Air.Temp.C	9.452270833	5/6/2013	2013	201305	506	5	125	Spring	2013Spring
13	ECO66G12	Air.Temp.C	12.2388125	5/7/2013	2013	201305	507	5	126	Spring	2013Spring
14	ECO66G12	Air.Temp.C	14.0598125	5/8/2013	2013	201305	508	5	127	Spring	2013Spring
15	ECO66G12	Air.Temp.C	16.40225	5/9/2013	2013	201305	509	5	128	Spring	2013Spring
16	ECO66G12	Air.Temp.C	16.02366667	5/10/2013	2013	201305	510	5	129	Spring	2013Spring

Check a DV file

Example - air temperature

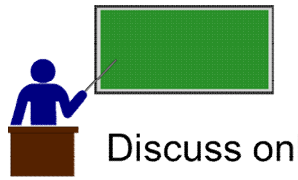


Name	Date modified	Type	Size
DV_ECO66G12_Aw_20130101_20131231_Air.Temp.C	2016-04-21 19:29	Microsoft Excel C...	15 KB
DV_ECO66G12_Aw_20130101_20131231_Water.Level.ft	2016-04-21 19:29	Microsoft Excel C...	16 KB
DV_ECO66G12_Aw_20130101_20131231_Water.Temp.C	2016-04-21 19:29	Microsoft Excel C...	16 KB
STATS_ECO66G12_Aw_20130101_20131231_Air.Temp.C	2016-04-21 19:29	Microsoft Excel C...	83 KB
STATS_ECO66G12_Aw_20130101_20131231_Air.Temp.C	2016-04-21 19:29	Adobe Acrobat D...	15 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Level.ft	2016-04-21 19:29	Microsoft Excel C...	85 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Level.ft	2016-04-21 19:29	Adobe Acrobat D...	15 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Temp.C	2016-04-21 19:29	Microsoft Excel C...	82 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Temp.C	2016-04-21 19:29	Adobe Acrobat D...	15 KB

Any comments? Questions?

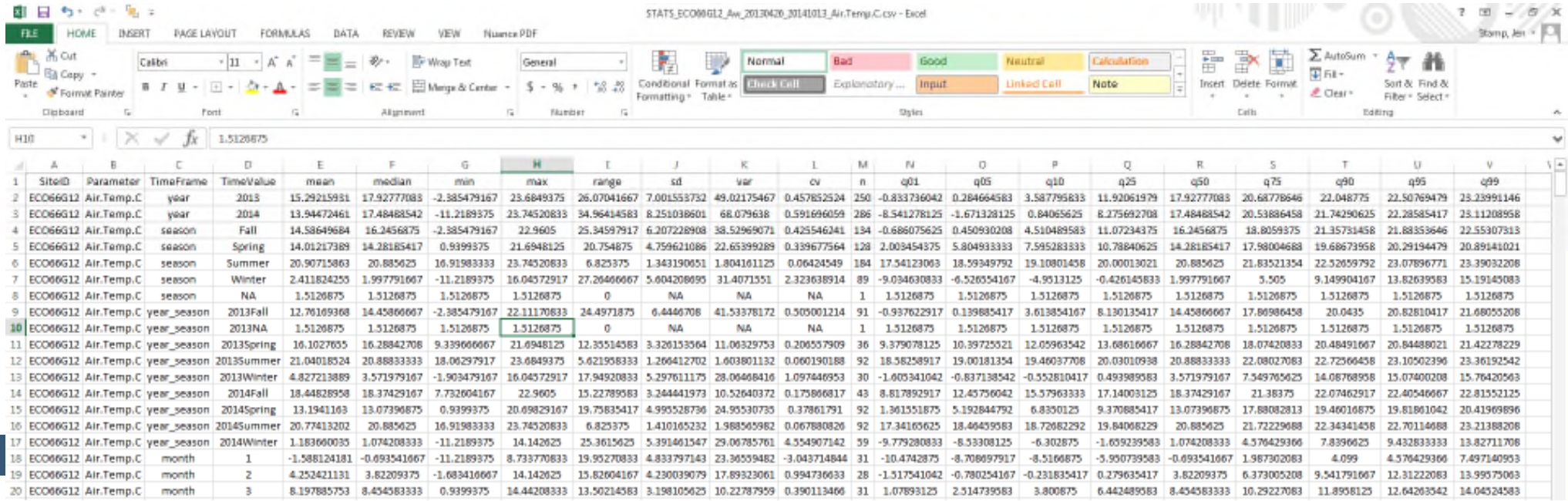
SummaryStats - STATs output

Example - air temperature



Statistics (for various timeframes) include –

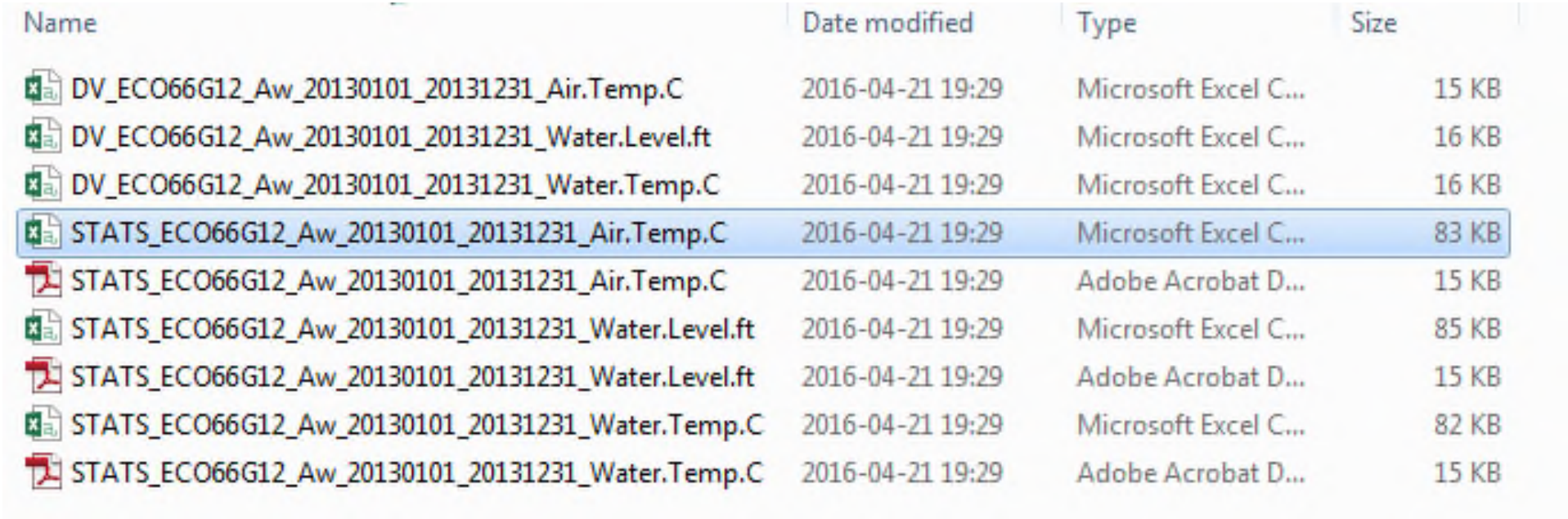
- Mean
- Median
- Min
- Max
- Range
- St dev
- Variance
- Count (n)
- Percentiles (01, 05, 10, 25, 50, 75, 90, 95, 99)



SiteID	Parameter	TimeFrame	TimeValue	mean	median	min	max	range	sd	var	cv	n	q01	q05	q10	q25	q50	q75	q90	q95	q99
ECO66G12	Air.Temp.C	year	2013	15.29215931	17.92777083	-2.385479167	23.6849375	26.07041667	7.001553732	49.02175467	0.457852524	250	-0.833736042	0.284664583	3.587795833	11.92061979	17.92777083	20.68778646	22.048775	22.50769479	23.23991146
ECO66G12	Air.Temp.C	year	2014	13.94472461	17.48488542	-11.2189375	23.74520833	34.96414583	8.251038601	68.079638	0.591696059	286	-8.541278125	-1.671328125	0.84065625	8.275692708	17.48488542	20.53886458	21.74290625	22.28585417	23.11208958
ECO66G12	Air.Temp.C	season	Fall	14.58649684	16.2456875	-2.385479167	22.9605	25.34597917	6.207228908	38.52969071	0.425546241	134	-0.686075625	0.450930208	4.510489583	11.07234375	16.2456875	18.8059375	21.35731458	21.88353646	22.55307313
ECO66G12	Air.Temp.C	season	Spring	14.01217389	14.28185417	0.9399375	21.6948125	20.754875	4.759621086	22.65399289	0.339677564	128	2.003454375	5.804933333	7.595283333	10.78840625	14.28185417	17.98004688	19.68673958	20.29194479	20.89141021
ECO66G12	Air.Temp.C	season	Summer	20.90715863	20.885625	16.91983333	23.74520833	6.825375	1.343190651	1.804161125	0.06424549	184	17.54123063	18.59349792	19.10801458	20.00013021	20.885625	21.83521354	22.52659792	23.07896771	23.39032208
ECO66G12	Air.Temp.C	season	Winter	2.411824255	1.997791667	-11.2189375	16.04572917	27.26466667	5.604208695	31.4071551	2.323638914	89	-9.034630833	-6.526554167	-4.9513125	-0.426145833	1.997791667	5.505	9.149904167	13.82639583	15.19145083
ECO66G12	Air.Temp.C	season	NA	1.5126875	1.5126875	1.5126875	1.5126875	0	NA	NA	NA	1	1.5126875	1.5126875	1.5126875	1.5126875	1.5126875	1.5126875	1.5126875	1.5126875	1.5126875
ECO66G12	Air.Temp.C	year_season	2013Fall	12.76169368	14.45866667	-2.385479167	22.11170833	24.4971875	6.4446708	41.53378172	0.505001214	91	-0.937622917	0.139885417	3.613854167	8.130135417	14.45866667	17.86986458	20.0435	20.82810417	21.68055208
ECO66G12	Air.Temp.C	year_season	2013NA	1.5126875	1.5126875	1.5126875	1.5126875	0	NA	NA	NA	1	1.5126875	1.5126875	1.5126875	1.5126875	1.5126875	1.5126875	1.5126875	1.5126875	1.5126875
ECO66G12	Air.Temp.C	year_season	2013Spring	16.1027655	16.28842708	9.339666667	21.6948125	12.35514583	3.326153564	11.06329753	0.206557909	36	9.379078125	10.39725521	12.05963542	13.68616667	16.28842708	18.07420833	20.48491667	20.84488021	21.42278229
ECO66G12	Air.Temp.C	year_season	2013Summer	21.04018524	20.88833333	18.06297917	23.6849375	5.621958333	1.266412702	1.603801132	0.060190188	92	18.58258917	19.00181354	19.46037708	20.03010938	20.88833333	22.08027083	22.72566458	23.10502396	23.36192542
ECO66G12	Air.Temp.C	year_season	2013Winter	4.827213889	3.571979167	-1.903479167	16.04572917	17.94920833	5.297611175	28.06468416	1.097446953	30	-1.605341042	-0.837138542	-0.552810417	0.493989583	3.571979167	7.549765625	14.08768958	15.07400208	15.76420563
ECO66G12	Air.Temp.C	year_season	2014Fall	18.44828958	18.37429167	7.732604167	22.9605	15.22789583	3.244441973	10.52640372	0.175866817	43	8.817892917	12.45756042	15.57963333	17.14003125	18.37429167	21.38375	22.07462917	22.40546667	22.81552125
ECO66G12	Air.Temp.C	year_season	2014Spring	13.1941163	13.07396875	0.9399375	20.69829167	19.75835417	4.995528736	24.95530735	0.37861791	92	1.361551875	5.192844792	6.8350125	9.370885417	13.07396875	17.88082813	19.46016875	19.81861042	20.41969896
ECO66G12	Air.Temp.C	year_season	2014Summer	20.77413202	20.885625	16.91983333	23.74520833	6.825375	1.410165232	1.988565982	0.067880826	92	17.34165625	18.46459583	18.72682292	19.84068229	20.885625	21.72229688	22.34341458	22.70114688	23.21388208
ECO66G12	Air.Temp.C	year_season	2014Winter	1.183660035	1.074208333	-11.2189375	14.142625	25.3615625	5.391461547	29.06785761	4.554907142	59	-9.779280833	-8.53308125	-6.302875	-1.659239583	1.074208333	4.576429366	7.8396625	9.432833333	13.82711708
ECO66G12	Air.Temp.C	month	1	-1.588124181	-0.693541667	-11.2189375	8.733770833	19.95270833	4.833797143	23.36559482	-3.043714844	31	-10.4742875	-8.708697917	-8.5166875	-5.950739583	-0.693541667	1.987302083	4.099	4.576429366	7.497140953
ECO66G12	Air.Temp.C	month	2	4.252421131	3.82209375	-1.683416667	14.142625	15.82604167	4.230039079	17.89323061	0.994736633	28	-1.517541042	-0.780254167	-0.231835417	0.279635417	3.82209375	6.373005208	9.541791667	12.31222083	13.99575063
ECO66G12	Air.Temp.C	month	3	8.197885753	8.454583333	0.9399375	14.44208333	13.50214583	3.198105625	10.22787959	0.390113466	31	1.07893125	2.514739583	3.800875	6.442489583	8.454583333	10.29227083	11.8958125	12.64263542	14.04524583

Check a STATS file

Example - air temperature



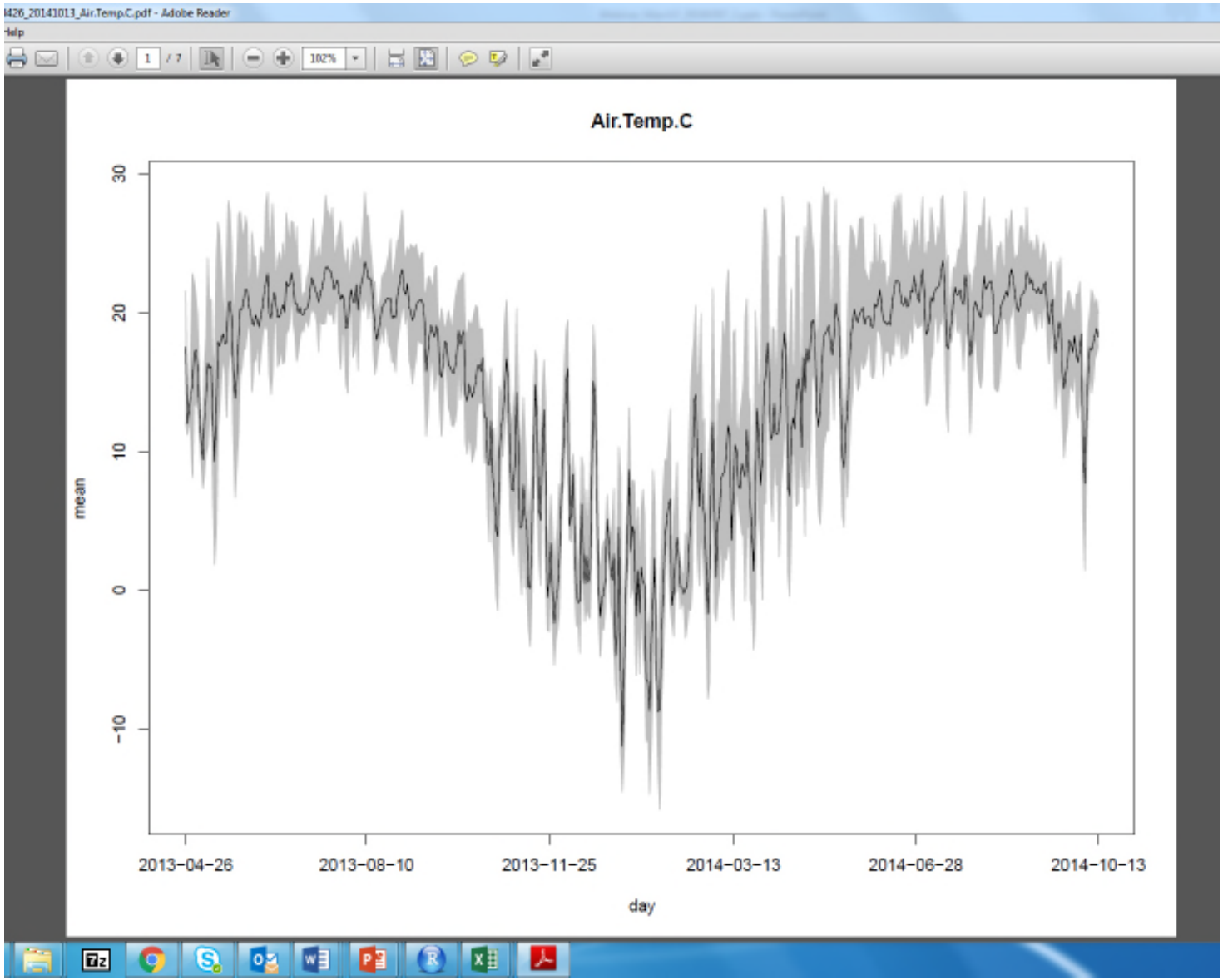
Name	Date modified	Type	Size
DV_ECO66G12_Aw_20130101_20131231_Air.Temp.C	2016-04-21 19:29	Microsoft Excel C...	15 KB
DV_ECO66G12_Aw_20130101_20131231_Water.Level.ft	2016-04-21 19:29	Microsoft Excel C...	16 KB
DV_ECO66G12_Aw_20130101_20131231_Water.Temp.C	2016-04-21 19:29	Microsoft Excel C...	16 KB
STATS_ECO66G12_Aw_20130101_20131231_Air.Temp.C	2016-04-21 19:29	Microsoft Excel C...	83 KB
STATS_ECO66G12_Aw_20130101_20131231_Air.Temp.C	2016-04-21 19:29	Adobe Acrobat D...	15 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Level.ft	2016-04-21 19:29	Microsoft Excel C...	85 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Level.ft	2016-04-21 19:29	Adobe Acrobat D...	15 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Temp.C	2016-04-21 19:29	Microsoft Excel C...	82 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Temp.C	2016-04-21 19:29	Adobe Acrobat D...	15 KB

Any comments? Questions?

SummaryStats - PDF output

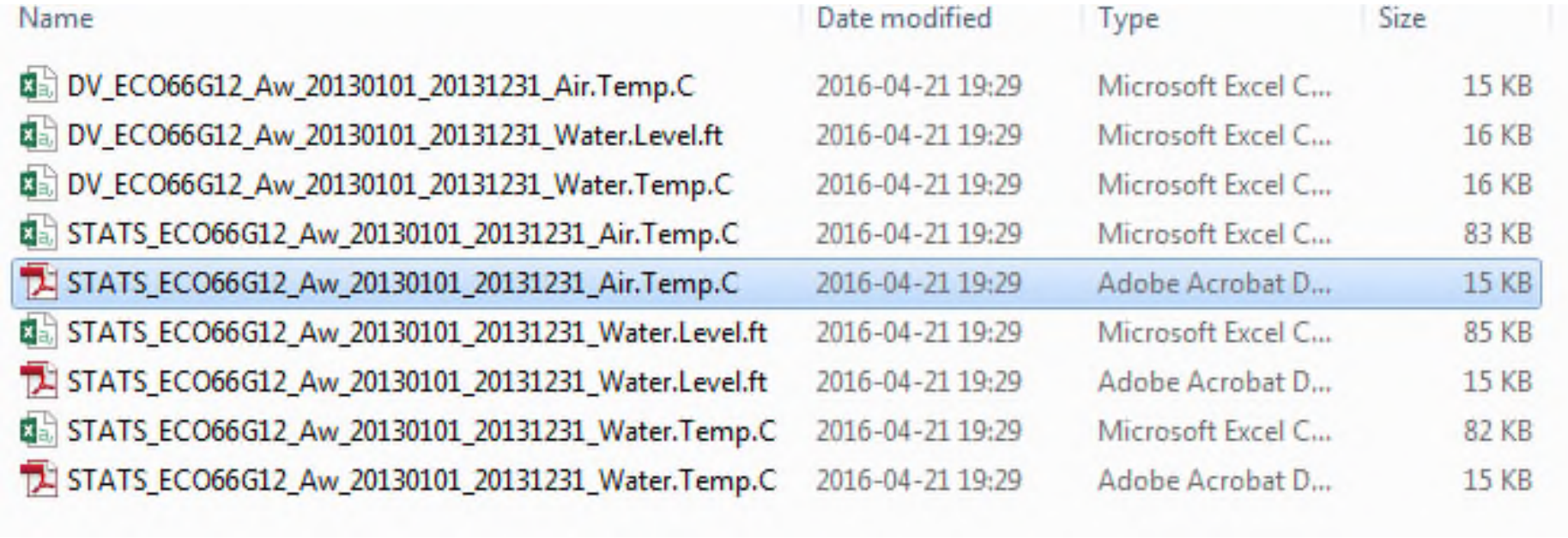
Example - air temperature, mean daily

Discuss only



Check a PDF file

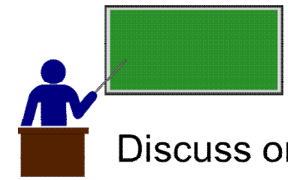
Example - air temperature



Name	Date modified	Type	Size
DV_ECO66G12_Aw_20130101_20131231_Air.Temp.C	2016-04-21 19:29	Microsoft Excel C...	15 KB
DV_ECO66G12_Aw_20130101_20131231_Water.Level.ft	2016-04-21 19:29	Microsoft Excel C...	16 KB
DV_ECO66G12_Aw_20130101_20131231_Water.Temp.C	2016-04-21 19:29	Microsoft Excel C...	16 KB
STATS_ECO66G12_Aw_20130101_20131231_Air.Temp.C	2016-04-21 19:29	Microsoft Excel C...	83 KB
STATS_ECO66G12_Aw_20130101_20131231_Air.Temp.C	2016-04-21 19:29	Adobe Acrobat D...	15 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Level.ft	2016-04-21 19:29	Microsoft Excel C...	85 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Level.ft	2016-04-21 19:29	Adobe Acrobat D...	15 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Temp.C	2016-04-21 19:29	Microsoft Excel C...	82 KB
STATS_ECO66G12_Aw_20130101_20131231_Water.Temp.C	2016-04-21 19:29	Adobe Acrobat D...	15 KB

Any comments? Questions?

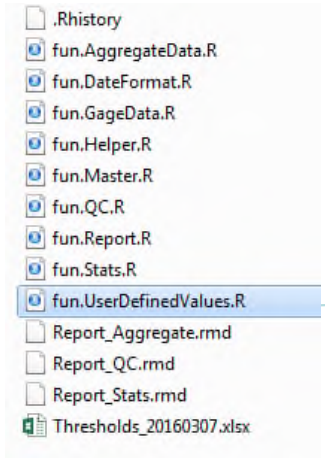
What if data are flagged as fail (F)? Are they included in the SummaryStat calculations?

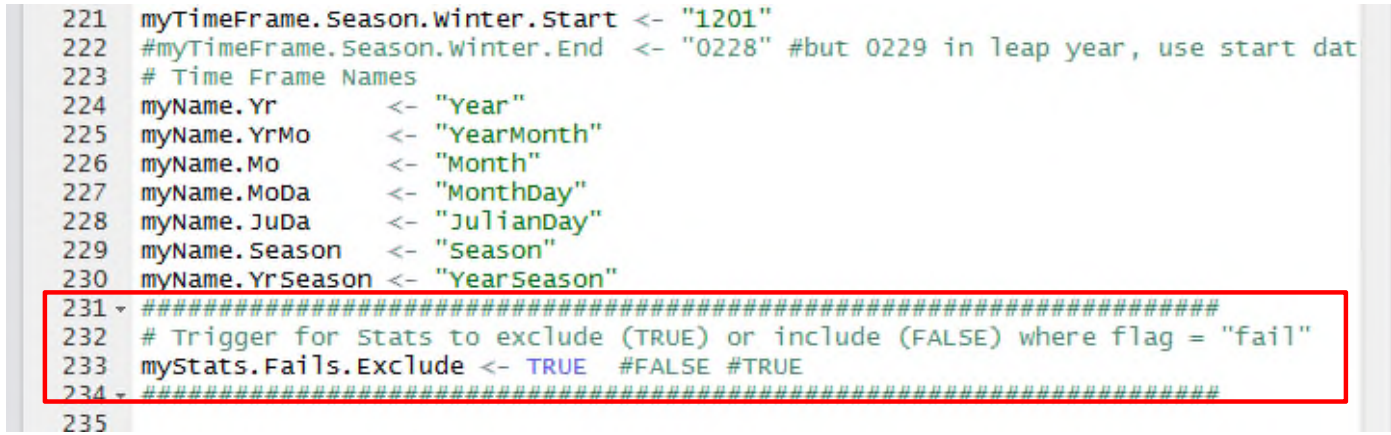


The scripts are written in a way that gives you flexibility on this. In the last line (line 233) of the **fun.UserDefinedValues.R file** in the **Scripts folder**, if you enter –

- TRUE – the script will exclude values flagged as 'F'
- FALSE – the script will include values flagged as 'F'

Scripts folder





Right now the script you are working with is set to 'TRUE'

EPA
United States
Environmental Protection
Agency

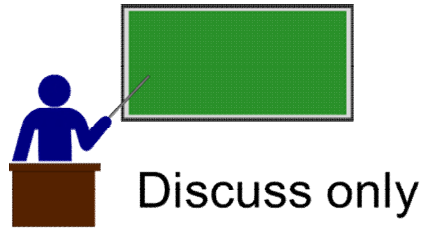
Congratulations!

You have successfully run
through test dataset #1!

EPA
United States
Environmental Protection
Agency

Test Dataset # 2

Test Dataset #2



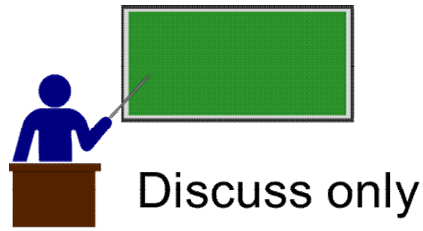
Standalone temperature sensor data

- Water temperature
- Air temperature

2 Files –

- Same site (WSR01)
- Different parameters
- Same date range
  - April 24, 2014 – July 10, 2014

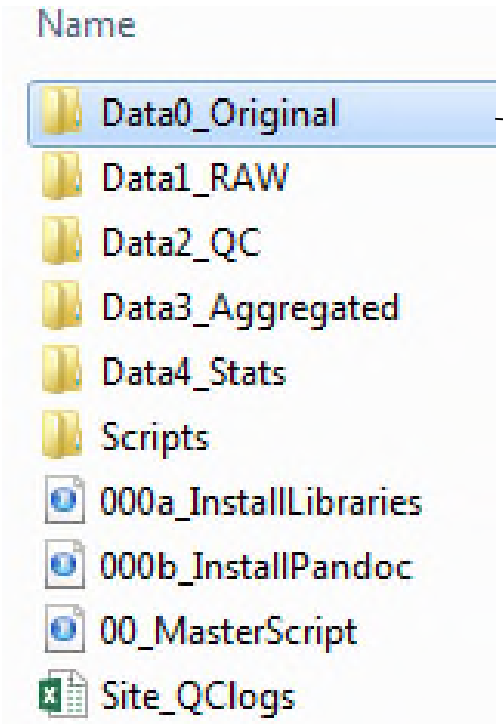
Work flow for Test Dataset #2

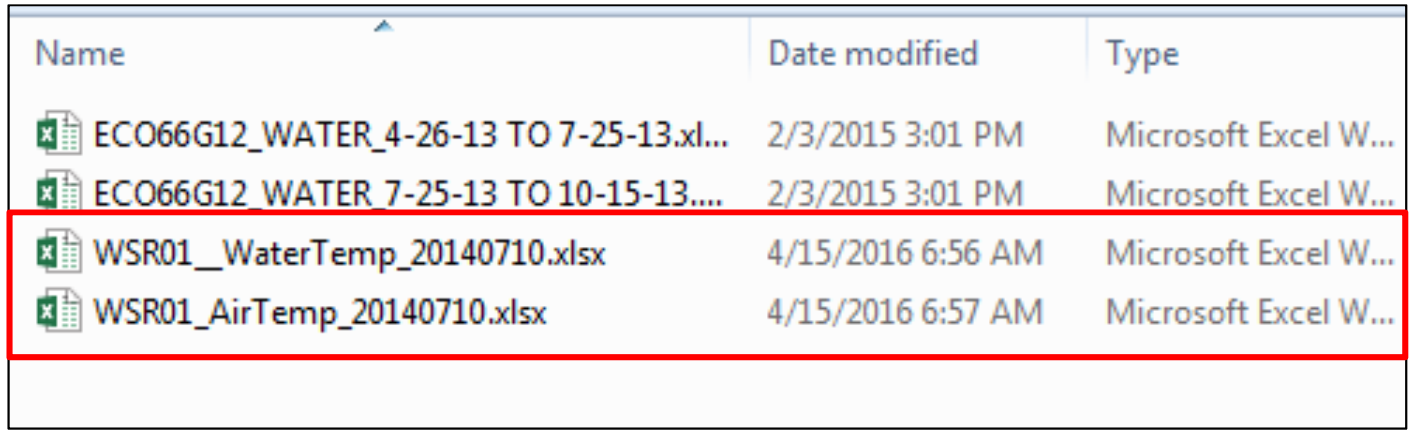


1. **Prepare your data** (review only*)
   - Move it into the correct folders
   - Organize, format and name the files
2. **Run the QC portion** of the MasterScript (on both files) – we'll run air temperature first, then water temperature
   - Check the data, correct if needed
3. **Run the Aggregate** portion of the MasterScript to get both parameters into the same file
   – we'll run air temperature first, then water temperature, then will aggregate them as an AW (air-water) file
4. **Run the SumStats** portion of the MasterScript to generate the summary statistics

*We prepared the data files in advance in order to maximize the amount of time we can spend running the actual scripts

Open the Data0_Original folder

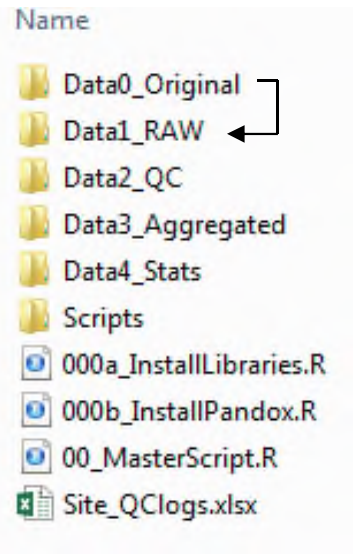




To save time, we've already put the WSR01 files into the Data0_Original folder.

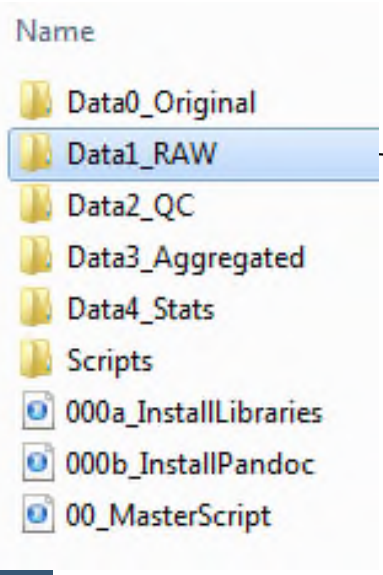
If you were working with your own files, you would put the originals into this folder (and would not alter them).

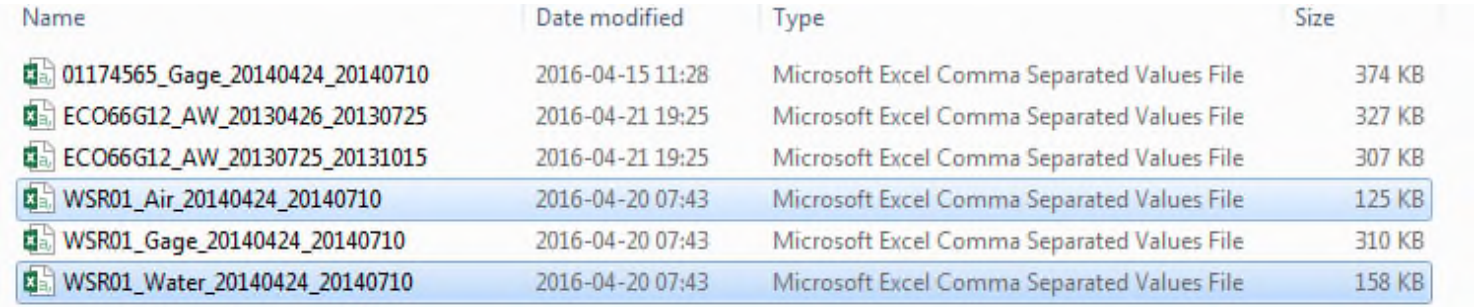
Open the Data1_RAW folder



To save time, we've already copied and pasted the WSR01 files from the Data0_Original into the Data1_RAW folder, and prepared them (reformatted, renamed, saved as .csv).

For a quick refresher, open the 2 files and review them.





Note: If you were working with your own files, you would have to perform these steps manually.

QC

1. Update entries in the MasterScript for the air temperature file

```
16 ##################################################################
17 # USER input in this section (see end of script for explanations)
18 ##################################################################
19 #
20 # PROMPT; Operation
21 Selection.Operation <- c("GetGageData","QCRaw", "Aggregate", "SummaryStats")
22 myData.Operation    <- Selection.Operation[2] #number corresponds to intended op
23 #
24 # PROMPT; Site ID
25 # single site;          "test2"
26 # group of sites;       c("test2", "HRCC", "PBCC", "ECO66G12",
27 myData.SiteID         <- "WSR01"
28 #
29 # PROMPT; Data Type
30 # Type of data file
31 Selection.Type       <- c("Air","Water","AW","Gage","AWG","AG"
32 myData.Type          <- Selection.Type[1] #number corresponds
33 #
34 # PROMPT; Start Date
35 # YYYY-MM-DD ("-" delimiter), leave blank for all data ("1900
36 myData.DateRange.Start  <- "2014-01-01"
37 #
38 # PROMPT; End Date
39 # YYYY-MM-DD ("-" delimiter), leave blank for all data (today
40 myData.DateRange.End    <- "2014-12-31"
41 ##################################################################
42 # PROMPT; SubDirectory, input file location.  Leave blank for defaults
43 Selection.SUB <- c("Data1_RAW","Data2_QC","Data3_Aggregated","Data4_Stats")
44 myDir.SUB.import <- "" #Selection.SUB[2]
45 #
46 # PROMPT; SubDirectory, output file location.  Leave blank for default.
47 myDir.SUB.export <- "" #Selection.SUB[3]
48
40:40  (Untitled)                                                R Script
```

The 5 entries should read –

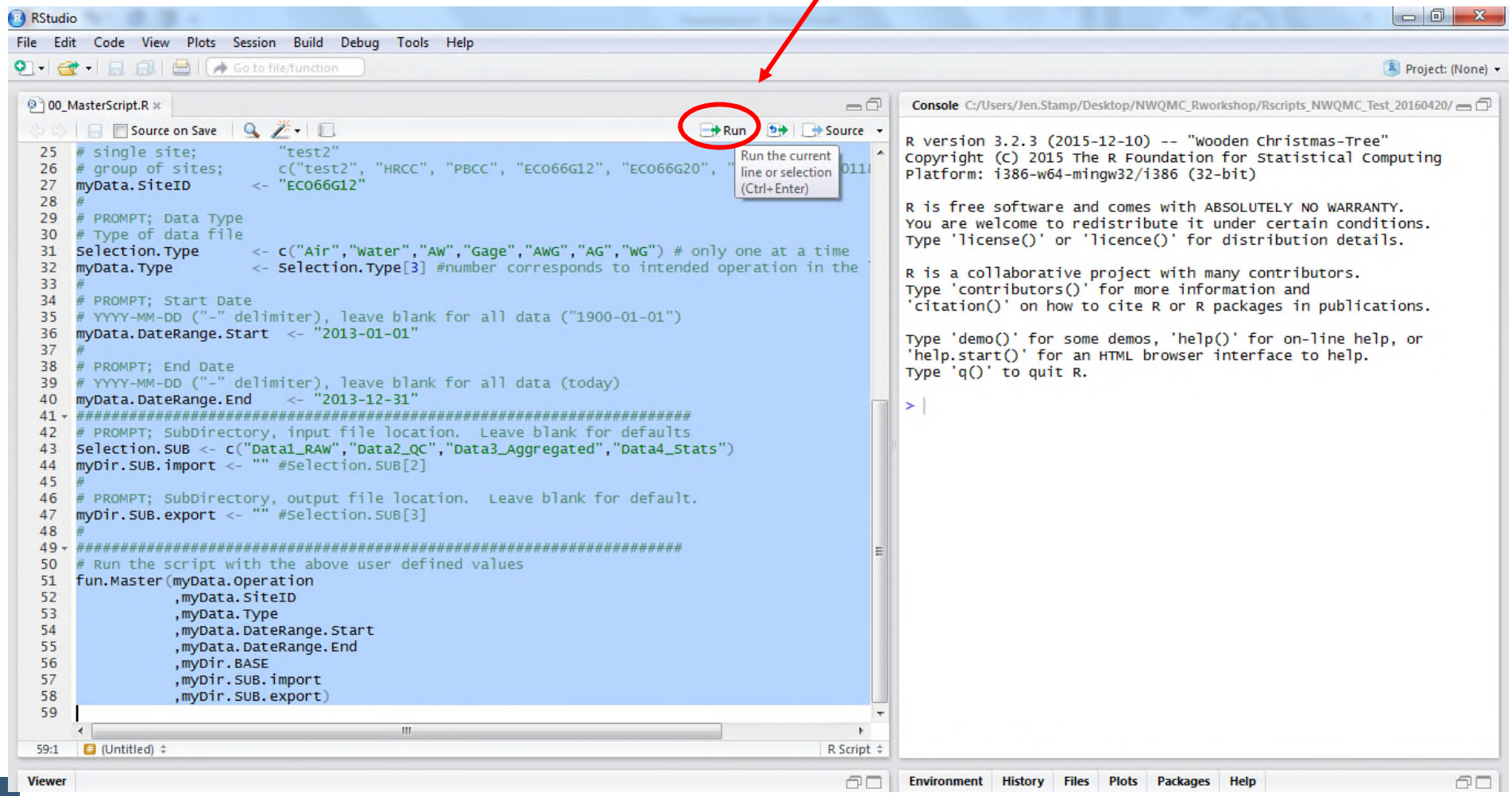
- Operation [2] = QCRaw
- SiteID “WSR01”
- Data type [1] = Air
- Start date “2014-01-01”
- End date “2014-12-31”

There are numerous possibilities for date entries…in this example, we used calendar year 2014. Feel free to experiment with other entries.

QC

2. Run the script

Using your cursor, highlight the script and click RUN



QC

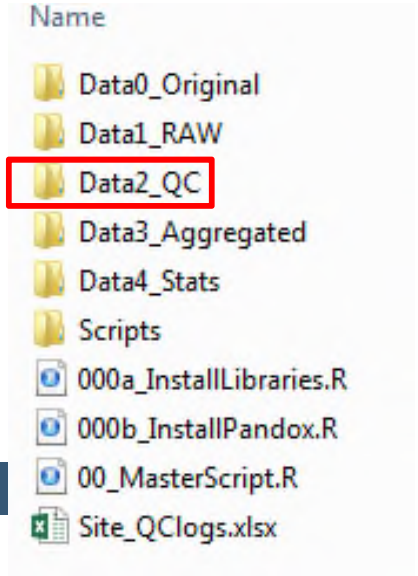
3. Verify that the script ran properly – air temperature

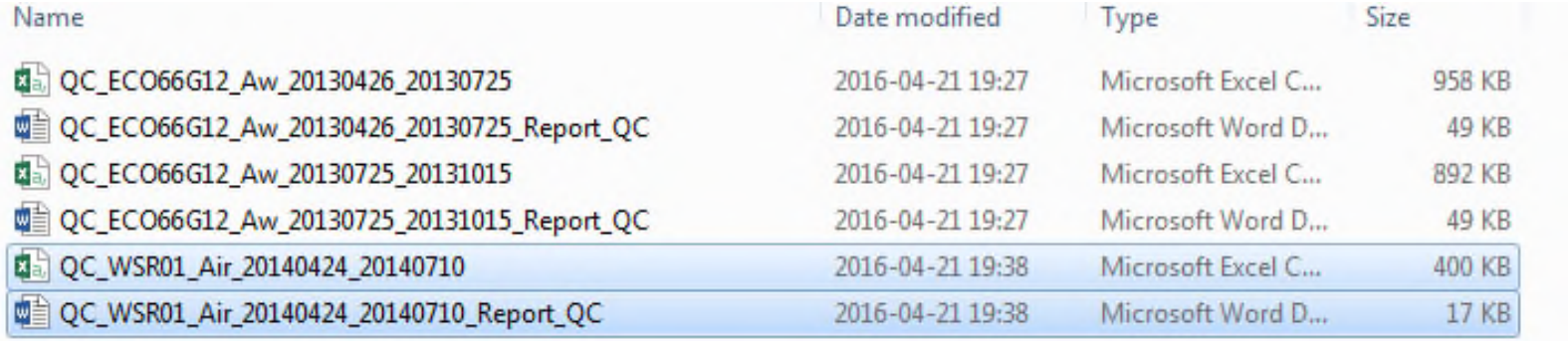
Check your Console (right) panel –

should end with **'Task COMPLETE'**

Open the Data2_QC folder – verify that there are 2 outputs (.csv & Word) for *air temperature*

Open & check the files (if time permits) – missing data, flagged values, time series plots





QC

4. Update entries in the MasterScript for the water temperature file

```
# USER input in this section (see end of script for explanations)
#
# PROMPT; Operation
Selection.Operation <- c("GetGageData","QCRaw", "Aggregate", "SummaryStats")
myData.Operation    <- Selection.Operation[2]  #number corresponds to intended op
#
# PROMPT; Site ID
# single site;          "test2"
# group of sites;       c("test2", "HRCC", "PBCC", "ECO66G12", "ECO66G20", "ECO68C
myData.SiteID       <- "WSR01"
#
# PROMPT; Data Type
# Type of data file
Selection.Type      <- c("Air","Water","AW","Gage","AW
myData.Type         <- Selection.Type[2] #number corre
#
# PROMPT; Start Date
# YYYY-MM-DD ("-" delimiter), leave blank for all data
myData.DateRange.Start  <- "2014-01-01"
#
# PROMPT; End Date
# YYYY-MM-DD ("-" delimiter), leave blank for all data
myData.DateRange.End    <- "2014-12-31"
#
# PROMPT; SubDirectory, input file location.  Leave bl
Selection.SUB <- c("Data1_RAW","Data2_QC","Data3_Aggre
myDir.SUB.import <- "" #Selection.SUB[2]
#
# PROMPT; SubDirectory, output file location.  Leave b
myDir.SUB.export <- "" #Selection.SUB[3]
```

(Untitled) R Script

The only entry that you need to change is Data type (from [1] to [2])

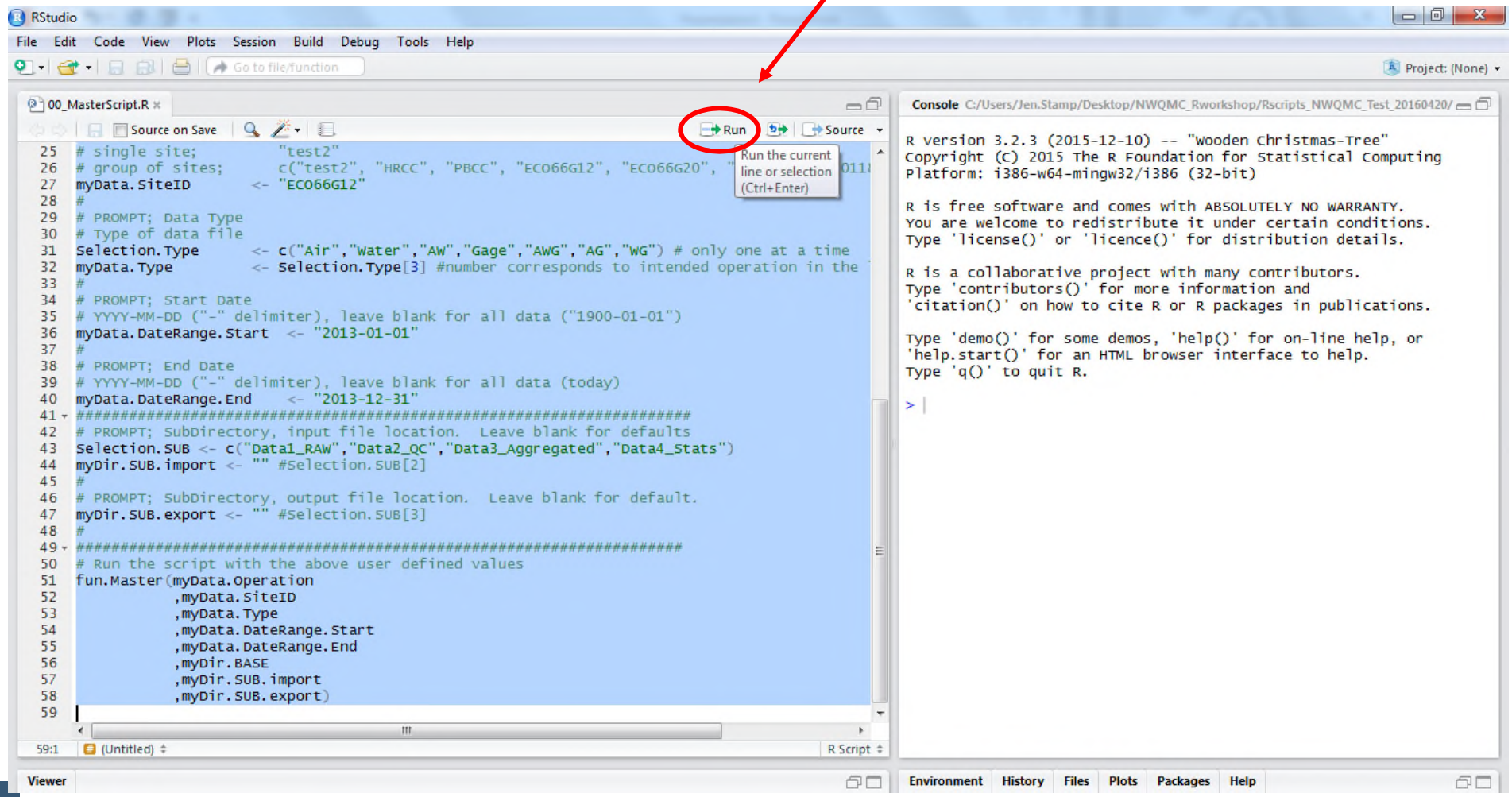
The 5 entries should read –

- Operation [2] = QCRaw
- SiteID "WSR01"
- **Data type** [2] = Water
- Start date "2014-01-01"
- End date "2014-12-31"

QC

5. Run the script

Using your cursor, highlight the script and click RUN



QC

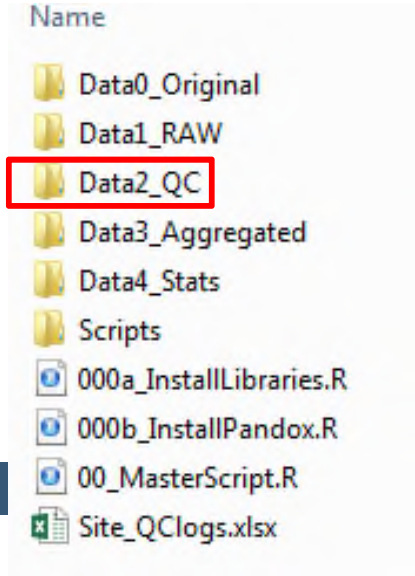
6. Verify that the script ran properly – water temperature

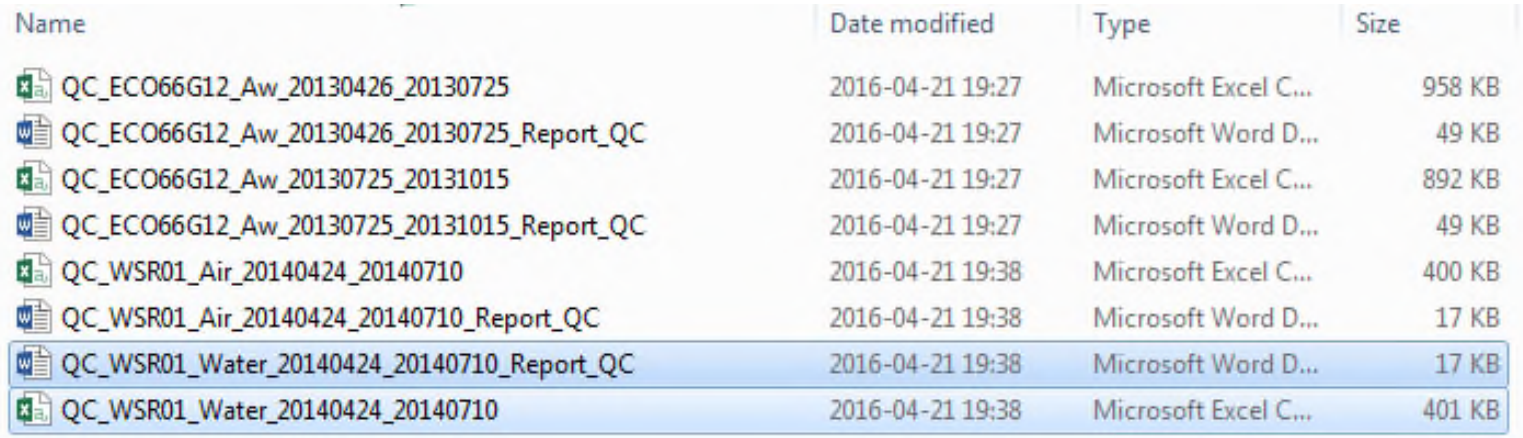
Check your Console (right) panel –

should end with **'Task COMPLETE'**

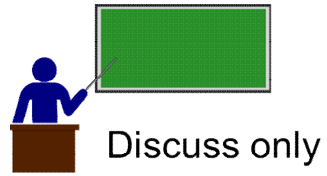
Open the Data2_QC folder – verify that there are 2 outputs (.csv & Word) for *water temperature*

Open & check the files (if time permits) – missing data, flagged values, time series plots





Aggregate – review steps



For each file (let's do air temperature first, then water temperature) -

1. **Update entries** in the MasterScript
2. **Run the script**
3. **Verify that the script ran properly**
4. **Open and check the files**

Then **change the Data type in the MasterScript to AW** (=air-water) & follow the same steps to generate an output that has both parameters (air + water temperature) in one file

Note: in this example, the individual air and water temperature files have to be run through the Aggregate script before the parameters can be combined into one 'AW' file. Ideally we could skip steps 1 & 2 and go right to the AW file, but the scripts are not currently written in a way that allows this (this is a future "wish list" item).

Aggregate

1. Update entries in the MasterScript for the air temperature file

```
16  ######################################################################
17  # USER input in this section (see end of script for explanations)
18  ######################################################################
19  #
20  # PROMPT; Operation
21  Selection.Operation <- c("GetGageData","QCRaw", "A
22  myData.Operation    <- Selection.Operation[3]  #nu
23  #
24  # PROMPT; Site ID
25  # single site;          "test2"
26  # group of sites;       c("test2", "HRCC", "PBCC",
27  myData.SiteID       <- "WSR01"
28  #
29  # PROMPT; Data Type
30  # Type of data file
31  Selection.Type      <- c("Air","Water","AW","Gage"
32  myData.Type         <- Selection.Type[1] #number c
33  #
34  # PROMPT; Start Date
35  # YYYY-MM-DD ("-" delimiter), leave blank for all
36  myData.DateRange.Start  <- "2014-01-01"
37  #
38  # PROMPT; End Date
39  # YYYY-MM-DD ("-" delimiter), leave blank for all data (today)
40  myData.DateRange.End    <- "2014-12-31"
41  ######################################################################
```

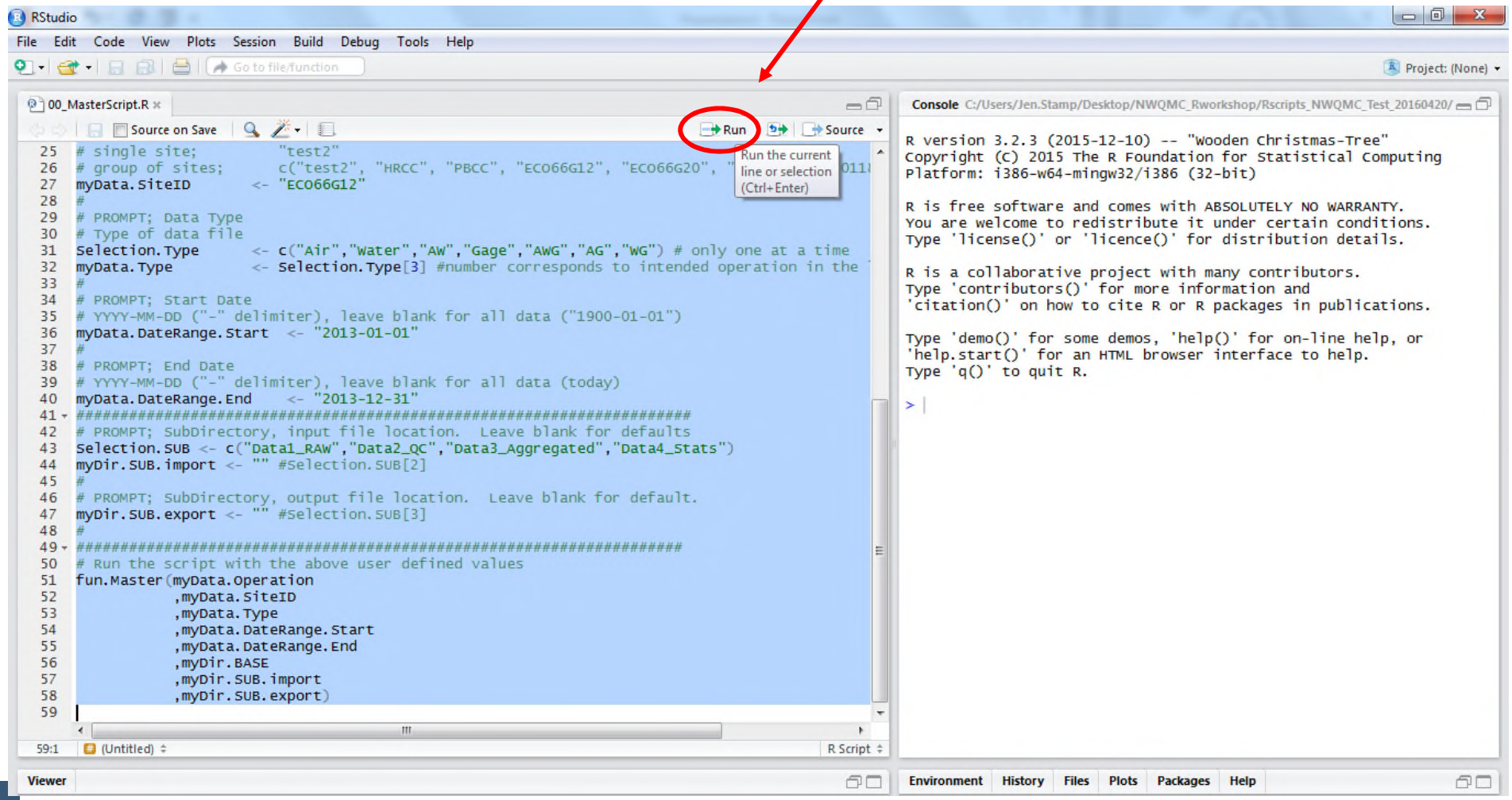
The 5 entries should read –

- Operation [3] = Aggregate
- SiteID “WSR01”
- Data type [1] = Air
- Start date “2014-01-01”
- End date “2014-12-31”

Aggregate

2. Run the script

Using your cursor, highlight the script and click RUN



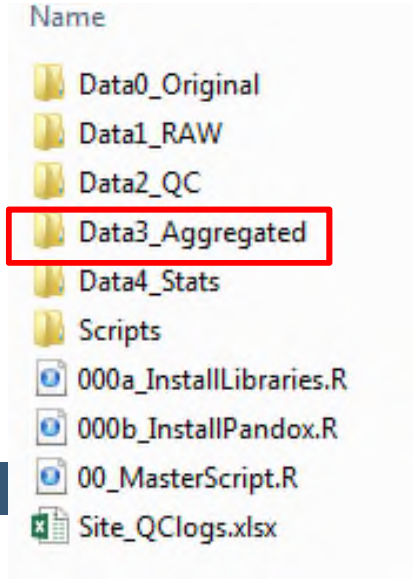
Aggregate

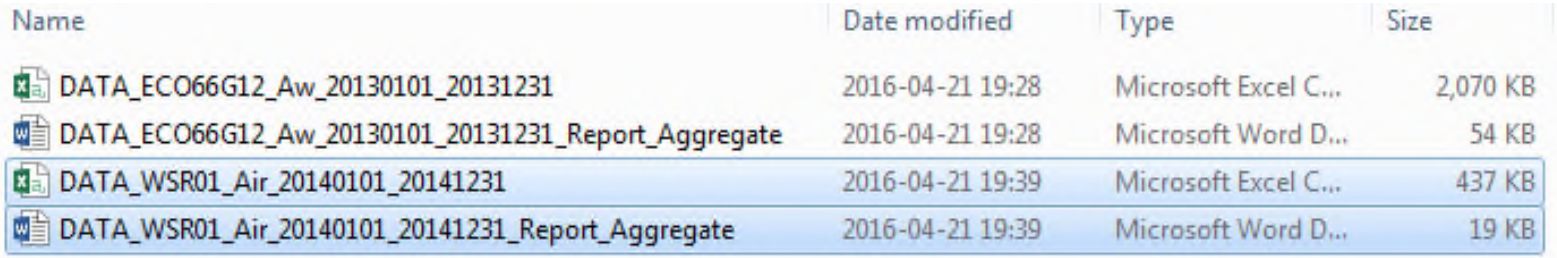
3. Verify that the script ran properly – air temperature

Check your Console (right) panel – log should include "Output created" and "Task COMPLETE"

Open the Data3_Aggregated folder – verify that there are 2 outputs for the WSR01 *air temperature* file

- .csv
- Aggregate report (Word)





Aggregate

4. Update entries in the MasterScript for the water temperature file

```
#####################################################################
# USER input in this section (see end of script for explanations)
#####################################################################
#
# PROMPT; Operation
Selection.Operation <- c("GetGageData","QCRaw", "Aggregate", "SummaryStats")
myData.Operation    <- Selection.Operation[3] #number corresponds to intended op
#
# PROMPT; Site ID
# single site;          "test2"
# group of sites;       c("test2", "HRCC", "PBCC", "ECO66G12", "ECO66G20", "ECO68C
myData.SiteID       <- "WSR01"
#
# PROMPT; Data Type
# Type of data file
Selection.Type      <- c("Air","Water","AW","Gage","AWG"
myData.Type         <- Selection.Type[2] #number corresp
#
# PROMPT; Start Date
# YYYY-MM-DD ("-" delimiter), leave blank for all data (
myData.DateRange.Start  <- "2014-01-01"
#
# PROMPT; End Date
# YYYY-MM-DD ("-" delimiter), leave blank for all data (
myData.DateRange.End    <- "2014-12-31"
#####################################################################
# PROMPT; SubDirectory, input file location.  Leave blan
Selection.SUB <- c("Data1_RAW","Data2_QC","Data3_Aggrega
myDir.SUB.import <- "" #Selection.SUB[2]
#
# PROMPT; SubDirectory, output file location.  Leave blank for default.
myDir.SUB.export <- "" #Selection.SUB[3]
```

(Untitled) R Script

The only entry that you need to change is Data type (from [1] to [2])

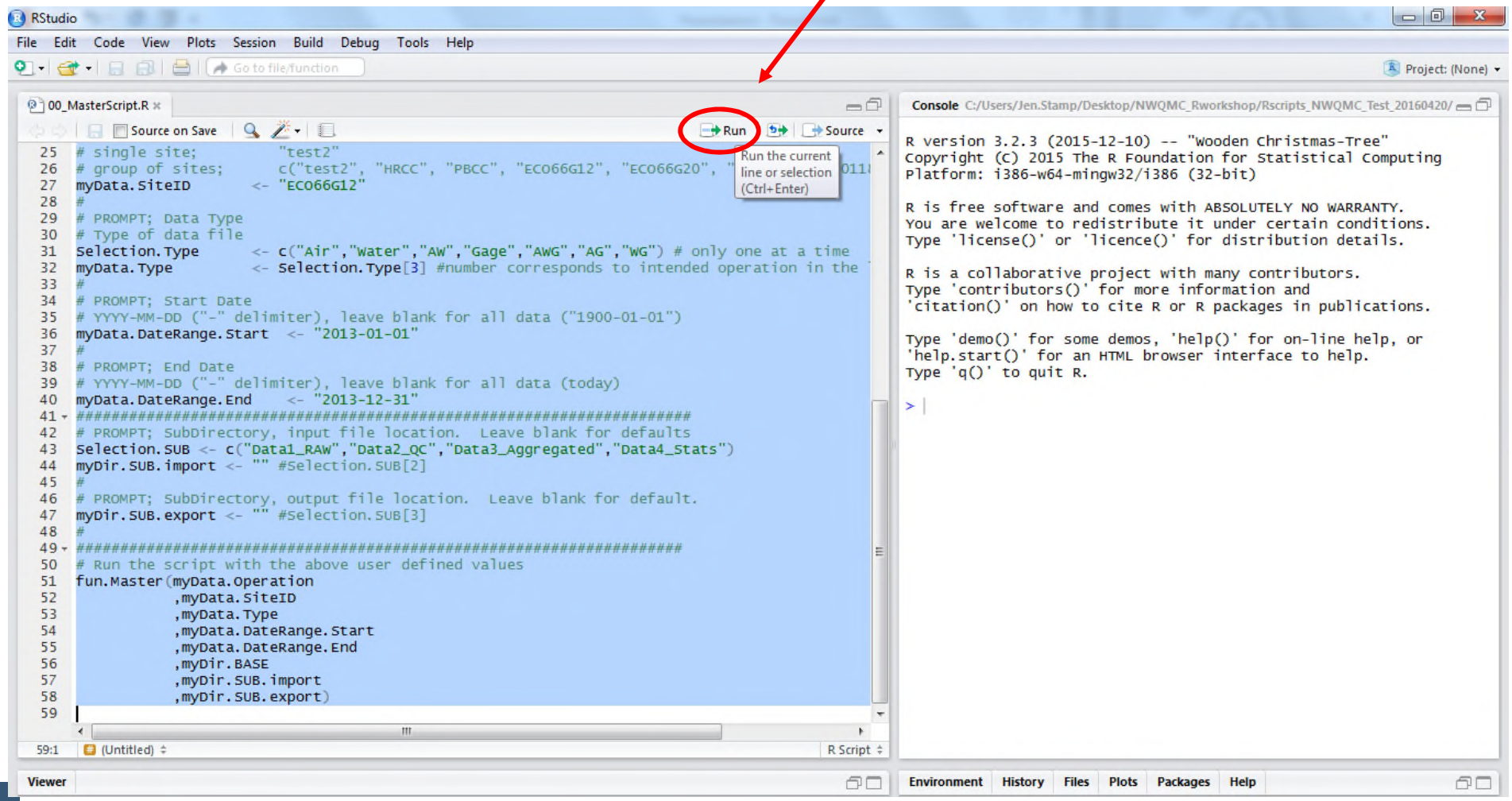
The 5 entries should read –

- Operation [3] = Aggregate
- SiteID “**WSR01**”
- **Data type** [2] = Water
- Start date “**2014-01-01**”
- End date “**2014-12-31**”

Aggregate

5. Run the script

Using your cursor, highlight the script and click RUN



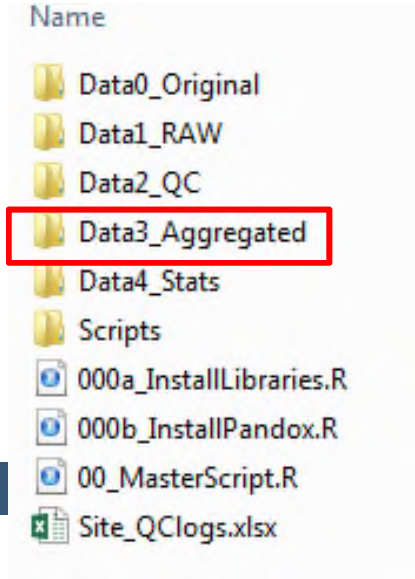
Aggregate

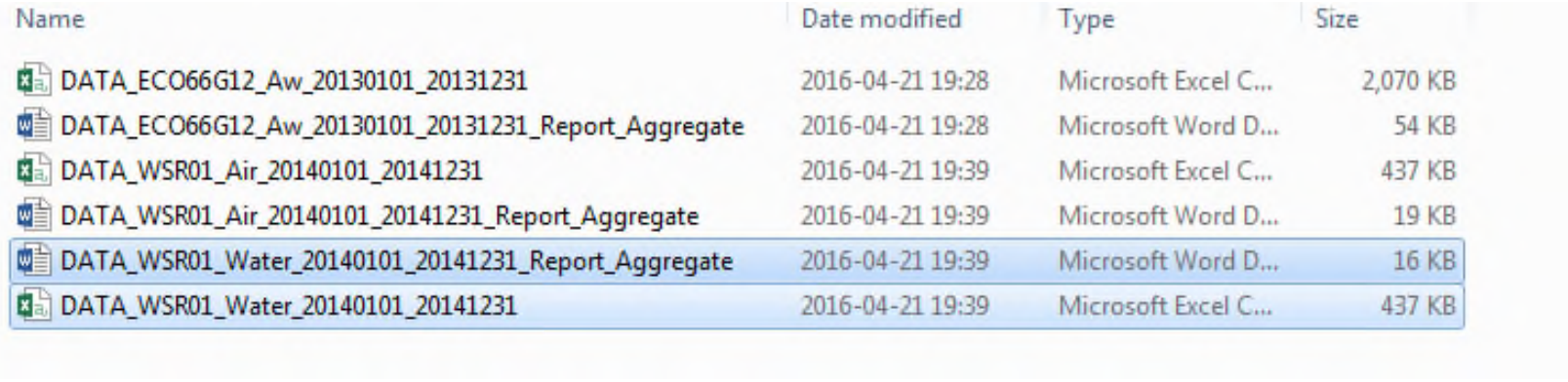
6. Verify that the script ran properly – water temperature

Check your Console (right) panel – log should include "Output created" and "Task COMPLETE"

Open the Data3_Aggregated folder – verify that there are 2 outputs for the WSR01 *water temperature* file

- .csv
- Aggregate report (Word)





Aggregate

7. Update entries in the MasterScript for the AW (air-water) file

```
16 ##################################################################
17 # USER input in this section (see end of script for explanations)
18 ##################################################################
19 #
20 # PROMPT; Operation
21 Selection.Operation <- c("GetGageData","QCRaw", "Aggregate", "SummaryStats")
22 myData.Operation    <- Selection.Operation[3]  #number corresponds to intended
23 #
24 # PROMPT; Site ID
25 # single site;          "test2"
26 # group of sites;       c("test2", "HRCC", "PBCC", "ECO66G12", "ECO66G20", "ECO
27 myData.SiteID       <- "WSR01"
28 #
29 # PROMPT; Data Type
30 # Type of data file
31 Selection.Type      <- c("Air","Water","AW","Gage",
32 myData.Type         <- Selection.Type[3] #number co
33 #
34 # PROMPT; Start Date
35 # YYYY-MM-DD ("-" delimiter), leave blank for all d
36 myData.DateRange.Start  <- "2014-01-01"
37 #
38 # PROMPT; End Date
39 # YYYY-MM-DD ("-" delimiter), leave blank for all d
40 myData.DateRange.End    <- "2014-12-31"
41 ##################################################
```

The only entry that you need to change is Data type (from [2] to [3])

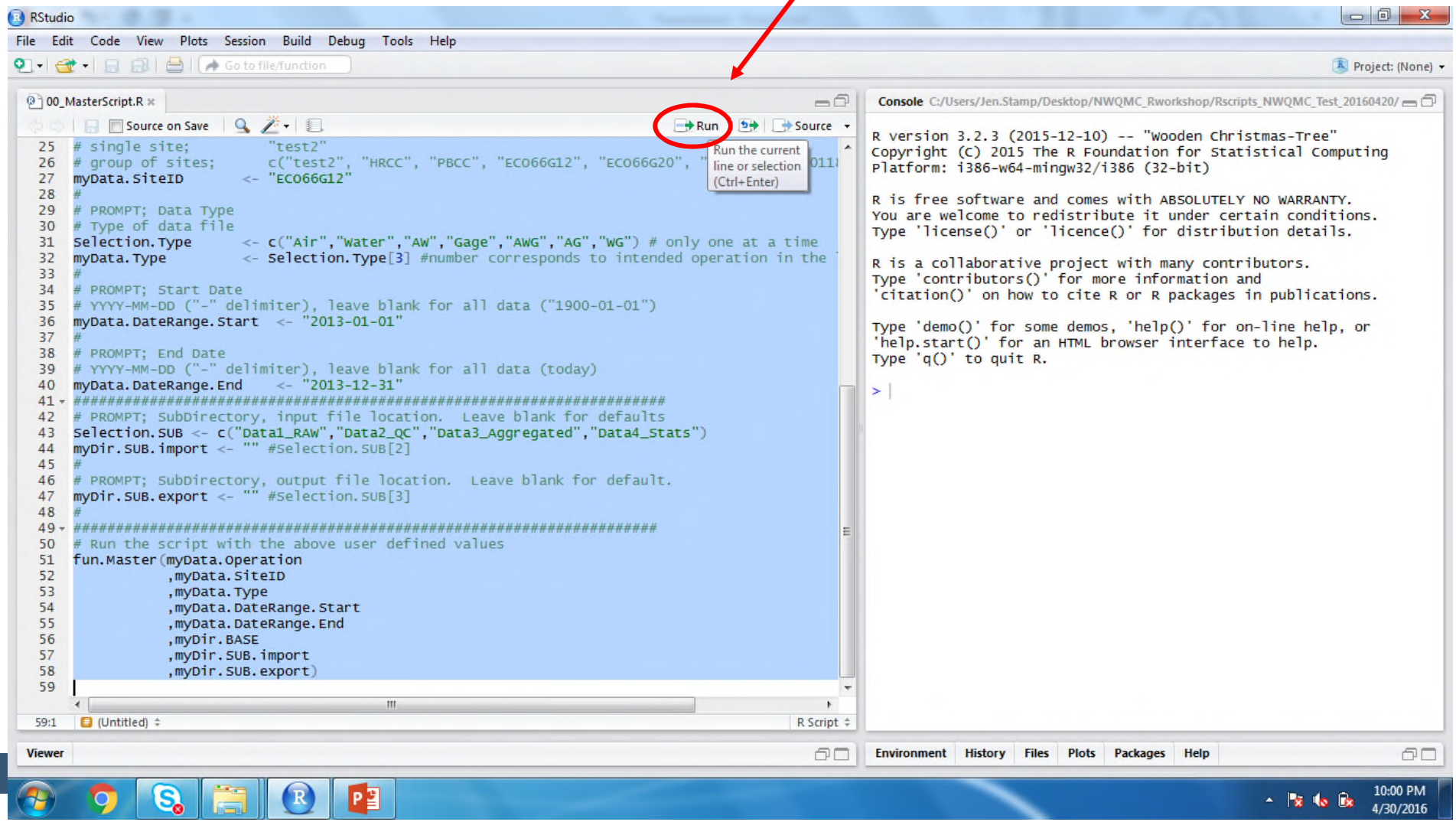
The 5 entries should read –

- Operation [3] = Aggregate
- SiteID “WSR01”
- **Data type** [3] = AW
- Start date “2014-01-01”
- End date “2014-12-31”

Aggregate

8. Run the script

Using your cursor, highlight the script and click RUN



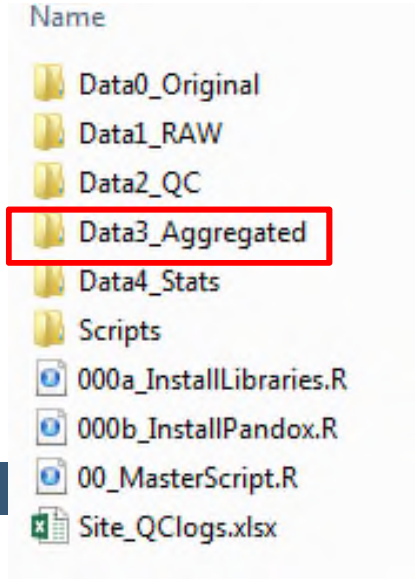
Aggregate

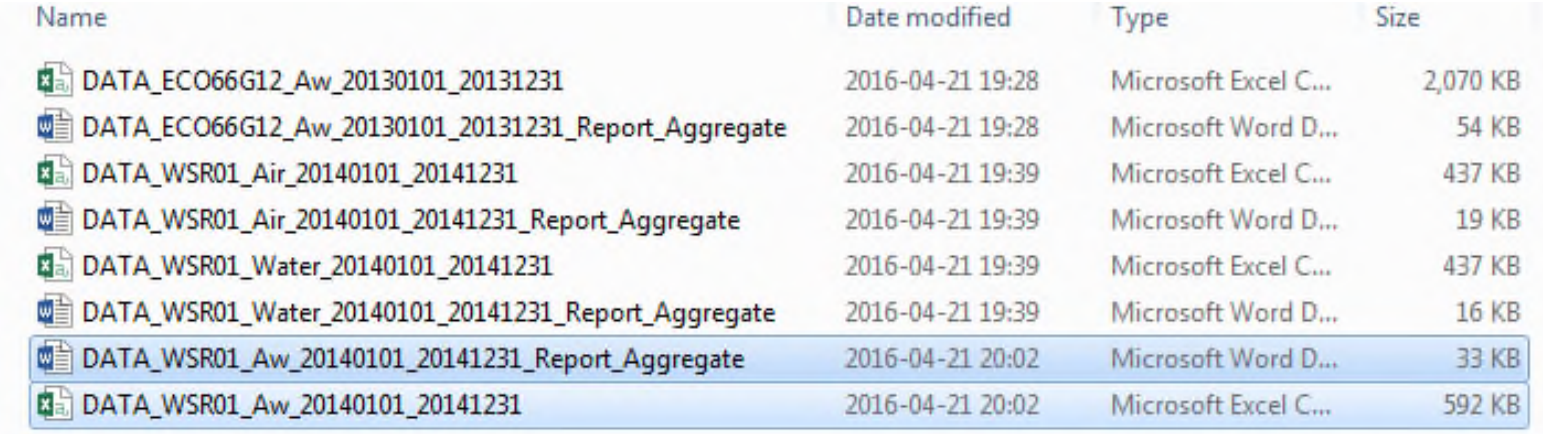
9. Verify that the script ran properly – AW

Check your Console (right) panel – log should include "Output created" and "Task COMPLETE"

Open the Data3_Aggregated folder – verify that there are 2 outputs for the WSR01 *AW* file

- .csv
- Aggregate report (Word)

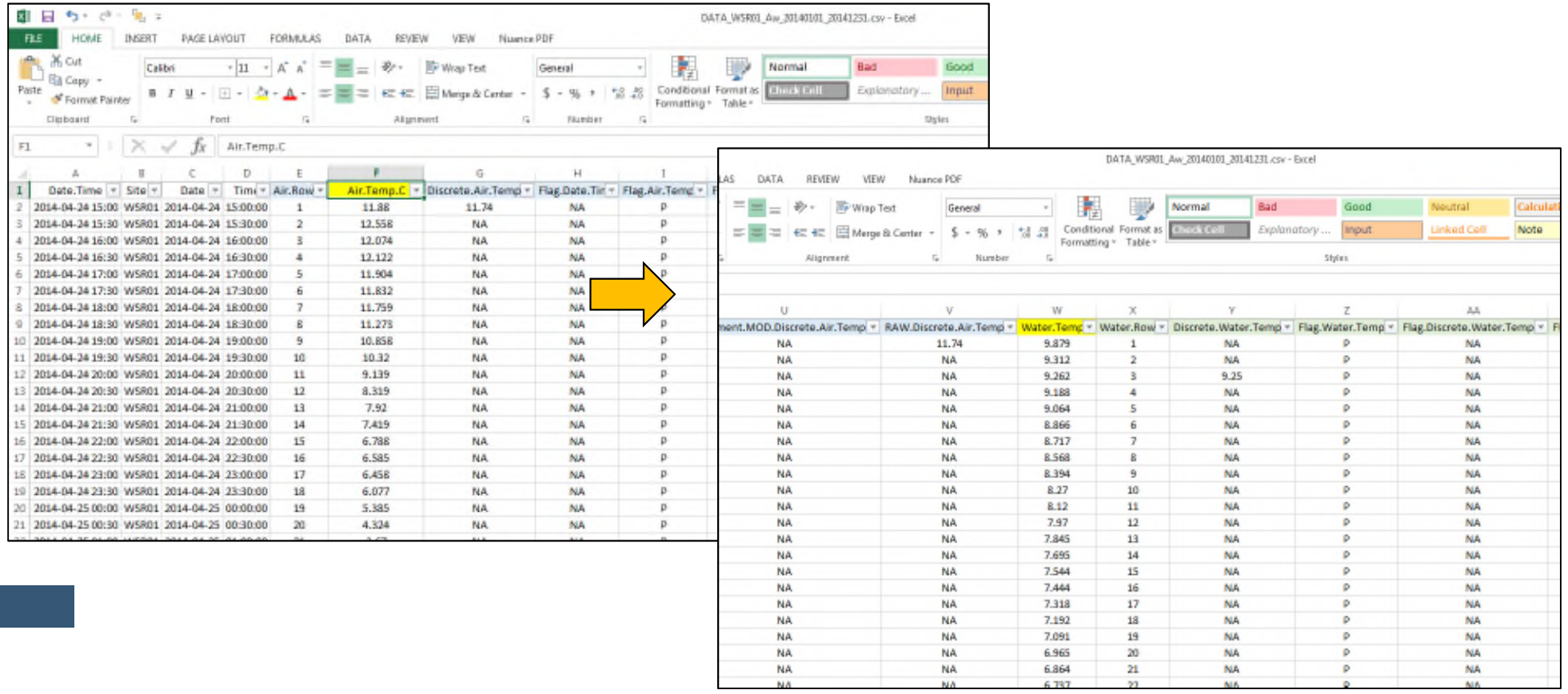




Aggregate

10. Open the AW files – give them a quick review (time-permitting)

Verify that both parameters are in the file.
Note: the air temperature fields come first (including flags, etc.); you'll need to scroll to the right to find the water temperature fields



SummaryStats

1. Update entries in the MasterScript

```
16  ##################################################################
17  # USER input in this section (see end of script for explanations)
18  ##################################################################
19  #
20  # PROMPT; Operation
21  Selection.Operation <- c("GetGageData","QCRaw", "Aggregate", "SummaryStats")
22  myData.Operation    <- Selection.Operation[4]  #number corresponds to intende
23  #
24  # PROMPT; Site ID
25  # single site;          "test2"
26  # group of sites;       c("test2", "HRCC", "PBCC", "ECO66G12", "ECO66G20", "E
27  myData.SiteID       <- "WSR01"
28  #
29  # PROMPT; Data Type
30  # Type of data file
31  Selection.Type      <- c("Air","Water","AW","Gage","AWG","AG","WG") # only o
32  myData.Type         <- Selection.Type[3] #number corresponds to intended oper
33  #
34  # PROMPT; Start Date
35  # YYYY-MM-DD ("-" delimiter), leave blank for
36  myData.DateRange.Start  <- "2014-01-01"
37  #
38  # PROMPT; End Date
39  # YYYY-MM-DD ("-" delimiter), leave blank for
40  myData.DateRange.End    <- "2014-12-31"
41  ##########################################
```

The only entry that you need to change is Operation from [3] to [4])

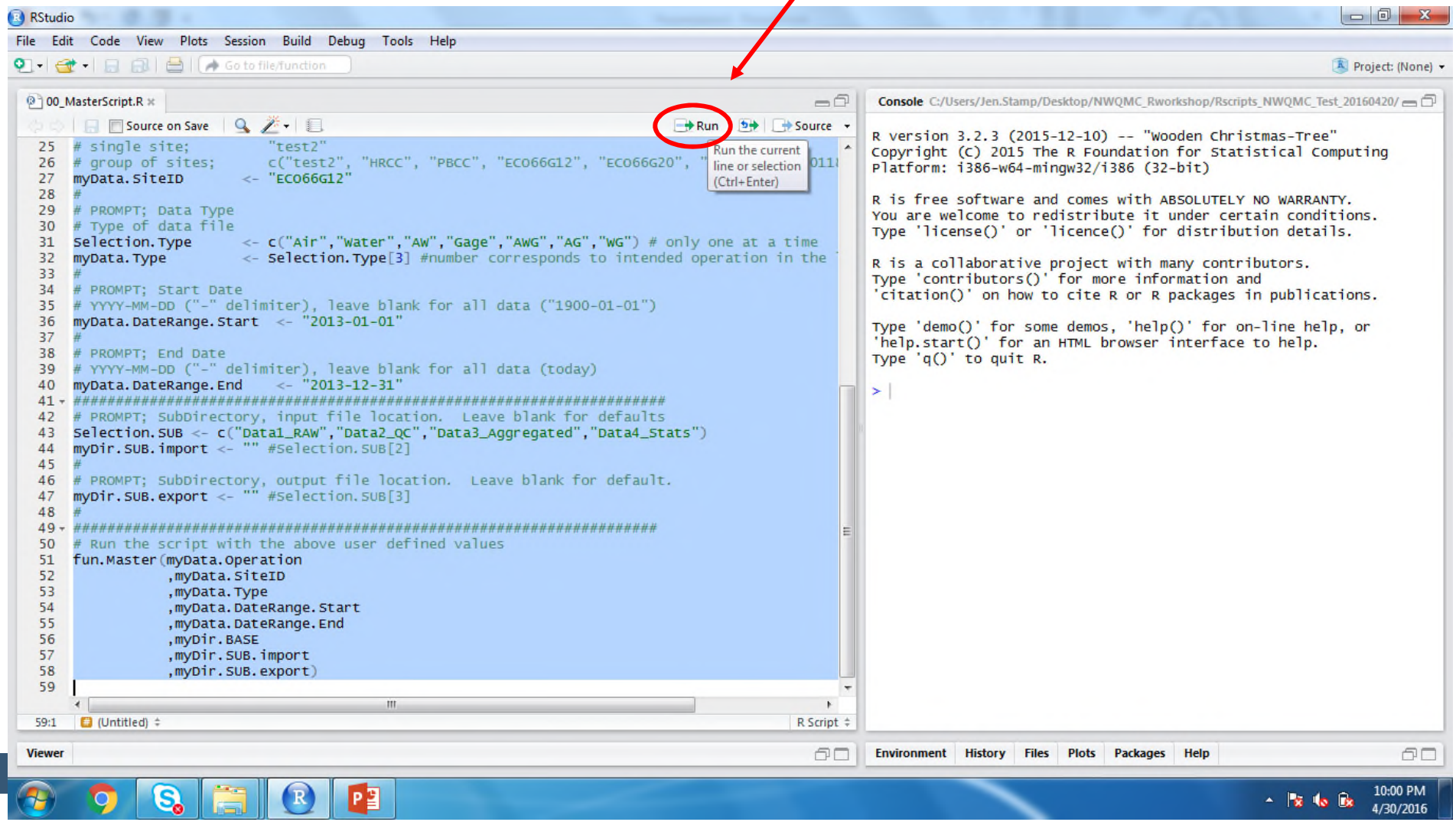
The 5 entries should read –

Operation [4] = SummaryStats
SiteID “**WSR01**”
Data type [3] = Air + water
Start date “**2014-01-01**”
End date “**2014-12-31**”

SummaryStats

2. Run the script

Using your cursor, highlight the script and click RUN



SummaryStats

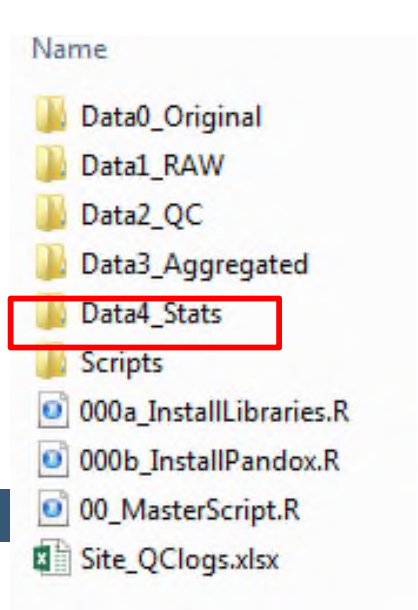
3. Verify that the script ran properly

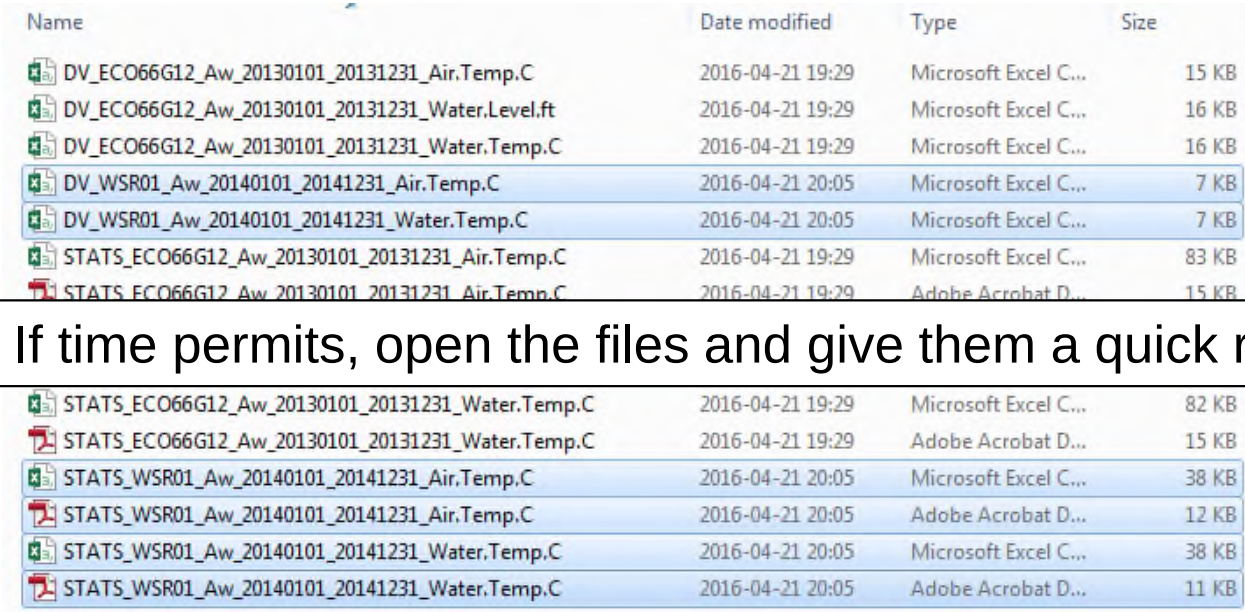
Check your Console (right) panel –

Total items to process = 2 (Water.Temp.C, Air.Temp.C)

Open the Data4_Stats folder – verify that there are 3 outputs for each parameter (*air temperature, water temperature)*

- **DV** .csv file (daily average)
- **STATS** .csv file (summary statistics)
- **PDF** files with various time series plots





If time permits, open the files and give them a quick review

EPA
United States
Environmental Protection
Agency

Congratulations!

You have successfully run through the temperature data.

Now let’s add in USGS gage data for this site

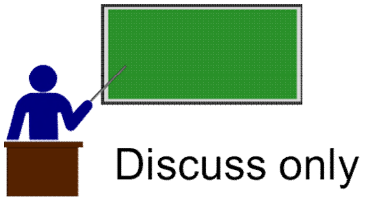
EPA
United States
Environmental Protection
Agency

Running the GetGageData script

Pulls data from the USGS NWIS website (thus requires an internet connection)

The USGS gage identifier for this site is **01174565.**

Work flow for GetGageData



1. Make the appropriate entries in the MasterScript
2. Run the GetGageData portion of the script
3. Verify that the script ran properly
   - Check Console panel (should see gage information, then Task COMPLETE)
   - Open the Data1_RAW folder – there should be a gage file (.csv)
4. Open the gage file and add a new column with the SiteID that you want to match the gage data with
5. Change the file name so that it references the appropriate SiteID

1. Make appropriate entries into the MasterScript

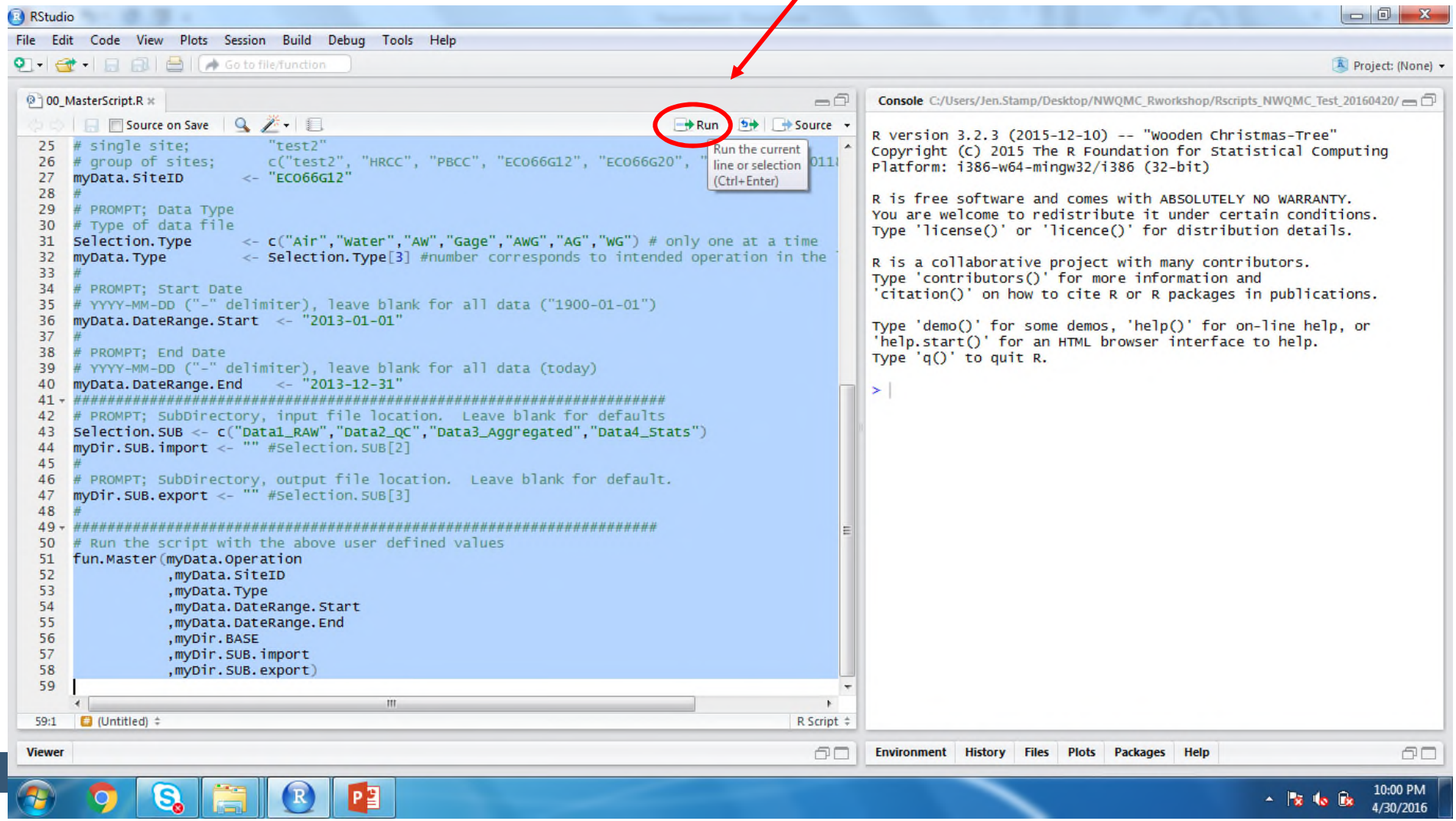
```
17  # USER input in this section (see end of script for explanations)
18  ###################################################################
19  #
20  # PROMPT; Operation
21  Selection.Operation <- c("GetGageData","QCRaw", "Aggregate", "SummaryStats")
22  myData.Operation    <- Selection.Operation[1]  #number corresponds to intended op
23  #
24  # PROMPT; Site ID
25  # single site;          "test2"
26  # group of sites;       c("test2", "HRCC", "PBCC"
27  myData.SiteID       <- "01174565"
28  #
29  # PROMPT; Data Type
30  # Type of data file
31  Selection.Type      <- c("Air","Water","AW","Gag
32  myData.Type         <- Selection.Type[4] #number
33  #
34  # PROMPT; Start Date
35  # YYYY-MM-DD ("-" delimiter), leave blank for al
36  myData.DateRange.Start  <- "2014-04-24"
37  #
38  # PROMPT; End Date
39  # YYYY-MM-DD ("-" delimiter), leave blank for all data (today)
40  myData.DateRange.End    <- "2014-07-10"
41  ###################################################################
```

The 5 entries should read –

- Operation [1] = GetGageData
- SiteID “**01174565**”
- Data type [4] = Gage
- Start date “**2014-04-24**”
- End date “**2014-07-10**”

2. Run the script

Using your cursor, highlight the script and click RUN

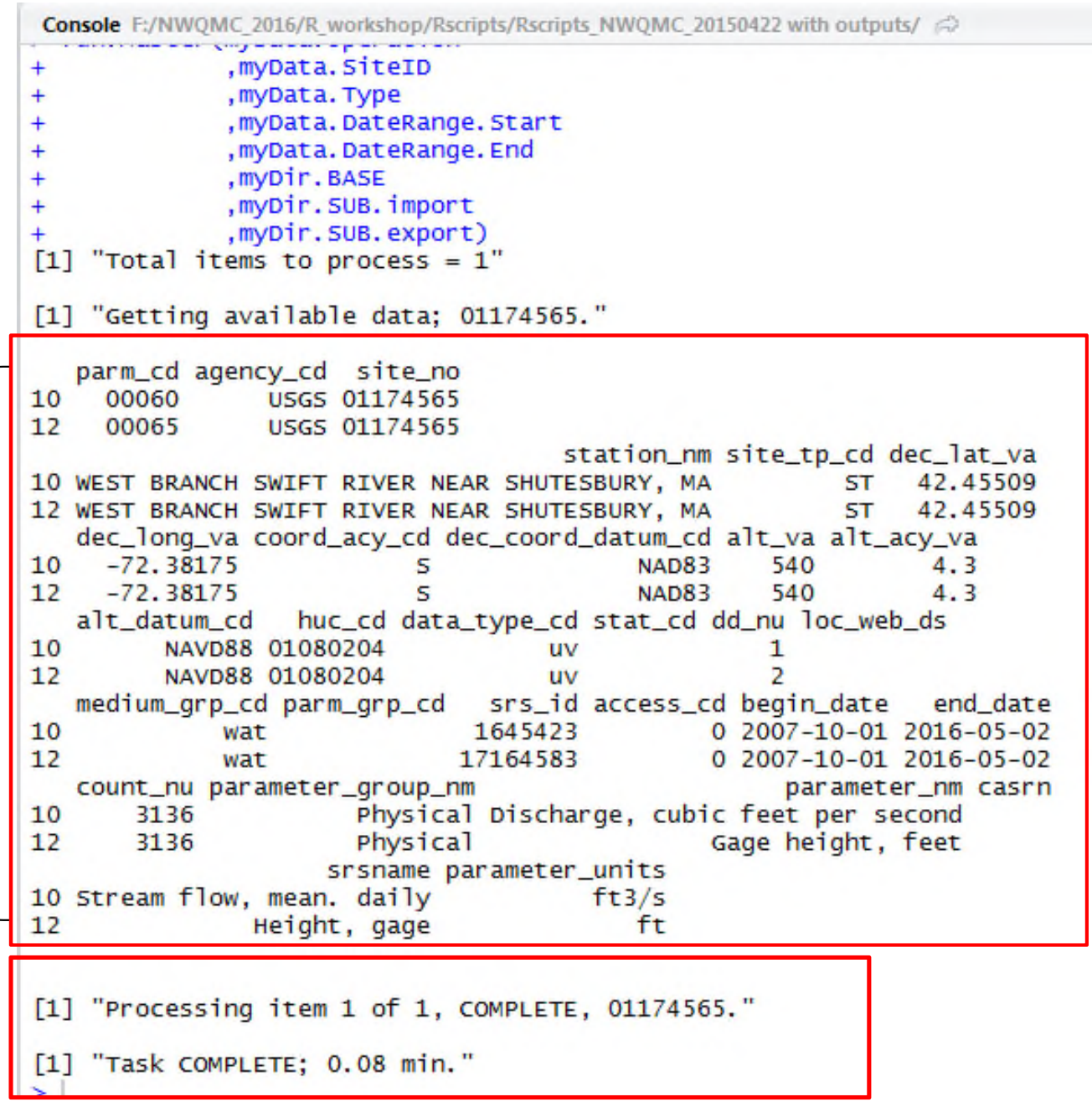


3. Verify that the script ran properly

Check the Console (right) panel

Gage information will appear immediately

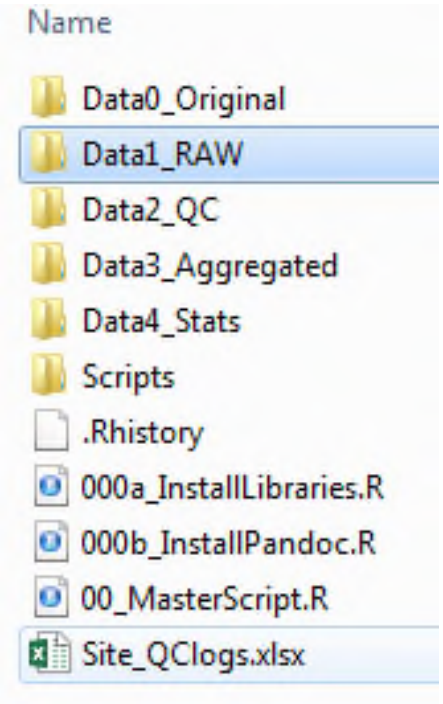
When it's finished running, it will say 'Task COMPLETE'

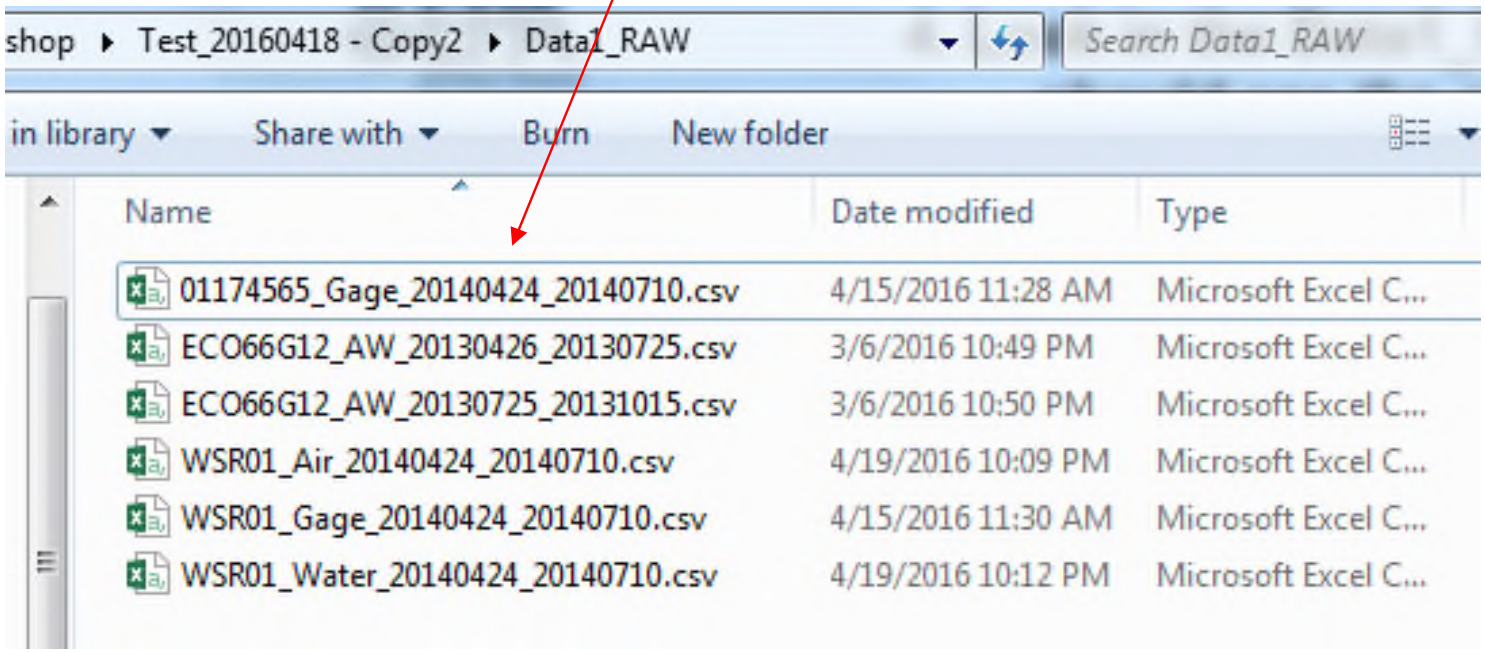


3. Verify that the script ran properly

Look in the Data1_RAW folder –
you should see the gage .csv file

File name: 01174565_Gage_20140424_20140710





4. Open the gage .csv file and add in a SiteID column, then save the updated file (still as a .csv)

Fill in the SiteID that you want to match the gage data with –
in this example, ***WSR01***

The script formats the column headings for you

	A	B	C	D	E	F	G
1	GageID	SiteID	Date.Time	Discharge.ft3/s	Water.Level.ft	Discharge.ft3/s_cd	Water.Level.ft_cd
2	1174565	WSR01	4/24/2014 0:00	38	1.77	A	A
3	1174565	WSR01	4/24/2014 0:15	37	1.75	A	A
4	1174565	WSR01	4/24/2014 0:30	38	1.76	A	A
5	1174565	WSR01	4/24/2014 0:45	38	1.76	A	A
6	1174565	WSR01	4/24/2014 1:00	38	1.76	A	A
7	1174565	WSR01	4/24/2014 1:15	38	1.77	A	A
8	1174565	WSR01	4/24/2014 1:30	37	1.75	A	A
9	1174565	WSR01	4/24/2014 1:45	37	1.75	A	A

The output includes USGS codes
(P=provisional; A=approved)

5. Change the file name so that it references the appropriate SiteID (=the SiteID that you want to match the temperature data files with)

Change the file name to 'WSR01_Gage_20140424_20140710'

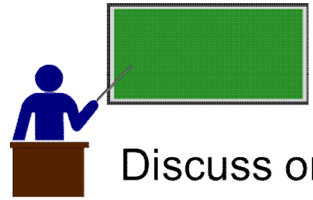
Name	Date modified	Type
01174565_Gage_20140424_20140710.csv	4/15/2016 11:28 AM	Microsoft Excel C...
ECO66G12_AW_20130426_20130725.csv	4/21/2016 7:25 PM	Microsoft Excel C...
ECO66G12_AW_20130725_20131015.csv	4/21/2016 7:25 PM	Microsoft Excel C...
WSR01_Air_20140424_20140710.csv	4/20/2016 7:43 AM	Microsoft Excel C...
WSR01_Gage_20140424_20140710.csv	4/20/2016 7:43 AM	Microsoft Excel C...
WSR01_Water_20140424_20140710.csv	4/20/2016 7:43 AM	Microsoft Excel C...

Congrats! You have successfully brought in USGS gage data.

EPA
United States
Environmental Protection
Agency

Aggregating gage data with the WSR01 temperature data

Work flow for aggregating the gage data with the temperature data



1. **Run the QC** portion of the script on the gage data
2. **Verify that the QC script ran properly**, make corrections if needed
3. **Run the Aggregate** portion of the script on the **gage file (alone)**
4. **Run the Aggregate** portion of the script on **Data type AWG (air-water-gage)**
5. **Verify that the Aggregate script ran properly**

1. Run the QC portion of the script on the gage file

Make appropriate entries into the MasterScript

```
17  # USER input in this section (see end of script for explanations)
18  ###################################################################
19  #
20  # PROMPT; Operation
21  Selection.Operation <- c("GetGageData","QCRaw", "Aggregate", "SummaryStats")
22  myData.Operation    <- Selection.Operation[2]  #numb
23  #
24  # PROMPT; Site ID
25  # single site;          "test2"
26  # group of sites;       c("test2", "HRCC", "PBCC", "
27  myData.SiteID         <- "WSR01"
28  #
29  # PROMPT; Data Type
30  # Type of data file
31  Selection.Type        <- c("Air","Water","AW","Gage","
32  myData.Type           <- Selection.Type[4] #number co
33  #
34  # PROMPT; Start Date
35  # YYYY-MM-DD ("-" delimiter), leave blank for all da
36  myData.DateRange.Start  <- "2014-04-24"
37  #
38  # PROMPT; End Date
39  # YYYY-MM-DD ("-" delimiter), leave blank for all data (today)
40  myData.DateRange.End    <- "2014-07-10"
41  ###################################################################
```

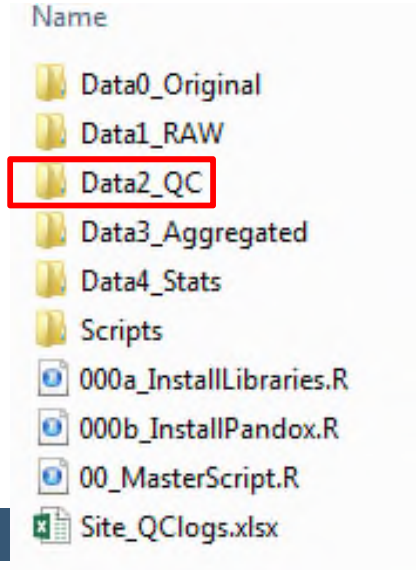
The 5 entries should read –

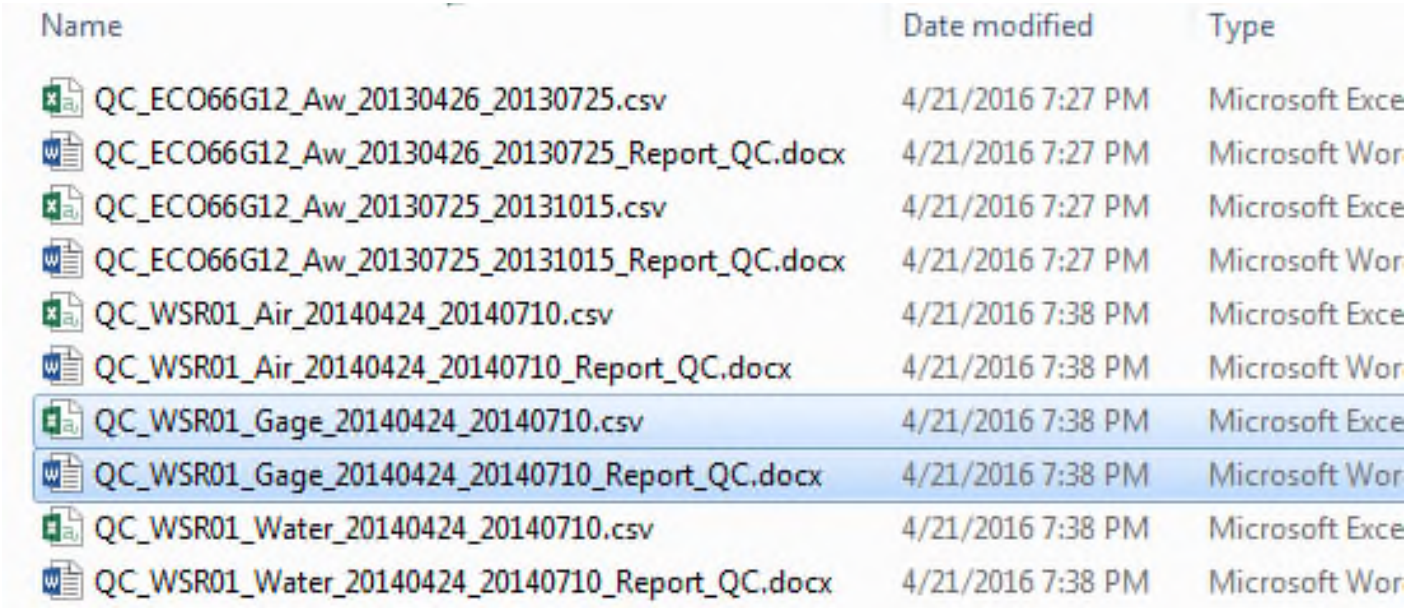
- Operation [2] = QCRaw
- SiteID “**WSR01**”
- Data type [4] = Gage
- Start date “**2014-04-24**”
- End date “**2014-07-10**”

Highlight the script in the Source panel and click RUN

2. Verify that the QC script ran properly

- **Check the Console** (right) panel –
  - should see 'Task COMPLETE'
- **Open the Data2_QC folder**
  - Verify that there are 2 outputs for the WSR01 *gage* file (.**csv** and Word files)





3. Run the Aggregate portion of the script on the gage file

Make appropriate entries into the MasterScript

```
16  ##################################################################
17  # USER input in this section (see end of script for explanations)
18  ##################################################################
19  #
20  # PROMPT; Operation
21  Selection.Operation <- c("GetGageData","QCRaw", "Aggregate", "SummaryStats")
22  myData.Operation    <- Selection.Operation[3] #numbe
23  #
24  # PROMPT; Site ID
25  # single site;          "test2"
26  # group of sites;       c("test2", "HRCC", "PBCC", "EC
27  myData.SiteID       <- "WSR01"
28  #
29  # PROMPT; Data Type
30  # Type of data file
31  Selection.Type      <- c("Air","Water","AW","Gage","A
32  myData.Type         <- Selection.Type[4] #number corr
33  #
34  # PROMPT; Start Date
35  # YYYY-MM-DD ("-" delimiter), leave blank for all dat
36  myData.DateRange.Start  <- "2014-04-24"
37  #
38  # PROMPT; End Date
39  # YYYY-MM-DD ("-" delimiter), leave blank for all data (today)
40  myData.DateRange.End    <- "2014-07-10"
41  ##################################################################
```

The 5 entries should read –

- Operation [3] = Aggregate
- SiteID “**WSR01**”
- Data type [4] = Gage
- Start date “**2014-04-24**”
- End date “**2014-07-10**”

Highlight the script in the Source panel and click RUN

4. Run the Aggregate portion of the script for AWG

(AWG= air/water/gage)

Make appropriate entries into the MasterScript

```
#
# PROMPT; Operation
Selection.Operation <- c("GetGageData","QCRaw", "Aggre
myData.Operation    <- Selection.Operation[3]  #number
#
# PROMPT; Site ID
# single site;          "test2"
# group of sites;       c("test2", "HRCC", "PBCC", "ECO
myData.SiteID       <- "WSR01"
#
# PROMPT; Data Type
# Type of data file
Selection.Type      <- c("Air","Water","AW","Gage","AWG","AG","WG") # only one at
myData.Type         <- Selection.Type[5] #number corresponds to intended operatio
#
# PROMPT; Start Date
# YYYY-MM-DD ("-" delimiter), leave blank for all data ("1900-01-01")
myData.DateRange.Start  <- "2014-04-24"
#
# PROMPT; End Date
# YYYY-MM-DD ("-" delimiter), leave blank for all data (today)
myData.DateRange.End    <- "2014-07-10"
##################################################################
```

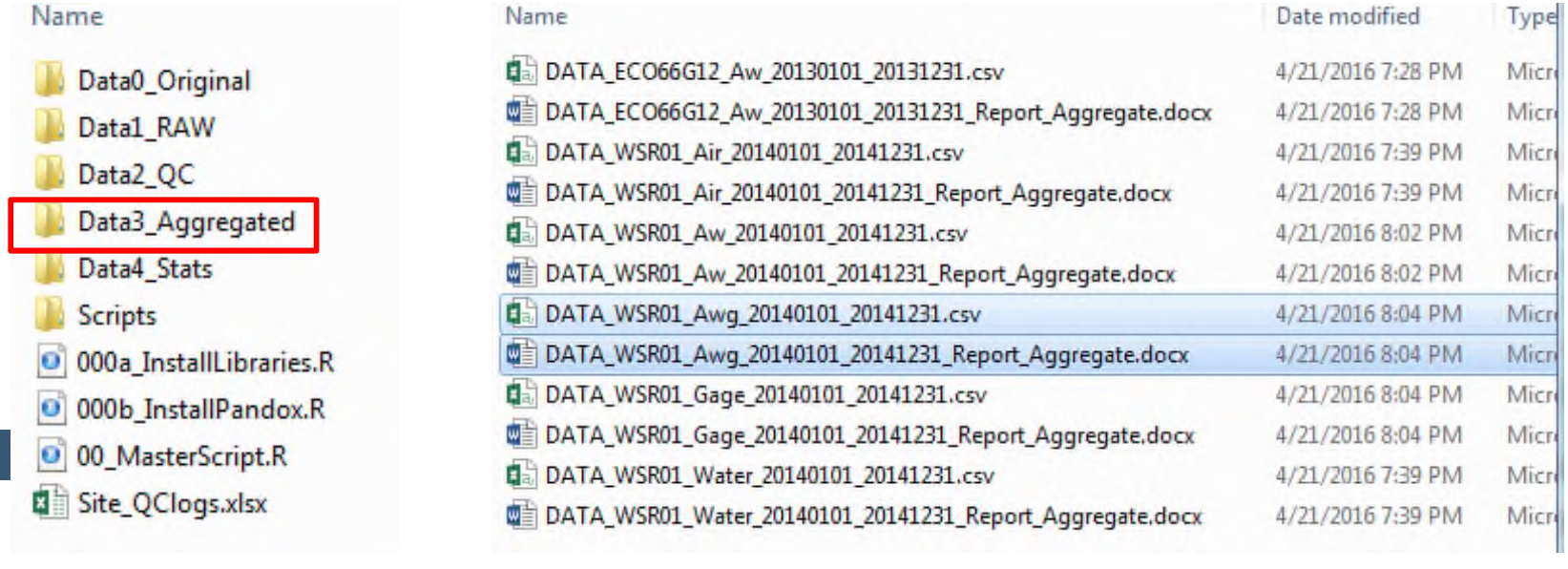
The 5 entries should read –

- Operation [3] = Aggregate
- SiteID “WSR01”
- Data type [5] = AWG
- Start date “2014-04-24”
- End date “2014-07-10”

Highlight the script in the Source panel and click RUN

5. Verify that the Aggregate script ran properly

- **Check the Console** (right) panel –
  - log should include “Output created” and “Task COMPLETE”
- **Open the Data3_Aggregated folder**
  - Verify that there are 2 outputs for the WSR01 *AWG* file (**.csv** and Word files)
- **Open & check the AWG files**
  - Verify that all the 3 parameters are in the file (scroll to the right!)
  - Note the different recording intervals (gage 15-minute, temperature 30-minute)



EPA
United States
Environmental Protection
Agency

Congratulations!

You have successfully aggregated the USGS gage data with the temperature data

EPA United States Environmental Protection Agency

Potential places to modify the R scripts

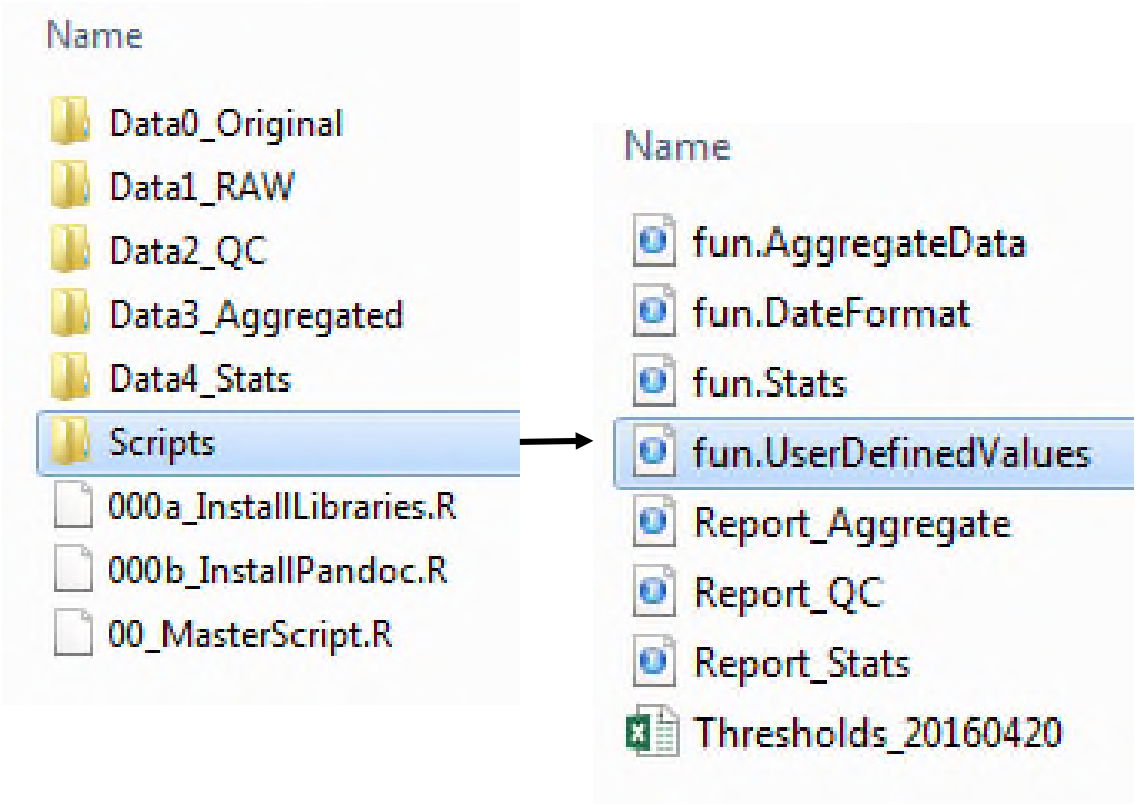
UserDefinedValues

- Thresholds
- Including or excluding flagged values
- Adding more variables (e.g., conductivity, other data from gages)

Changing QC thresholds

The thresholds for the tests (gross, spikes, rate of change (RoC) and flat line) are in the **fun.UserDefinedValues.R file** in the **Scripts folder**.

You can go into this file and change thresholds if desired (we're still trying to figure out appropriate thresholds for each parameter).



Thresholds for Gross test

United States Environmental Protection Agency

Gross Min/Max - Fail flag

- Extreme, unrealistic, seemingly impossible values
- Example - water temp values flagged ‘F’ if ≥ 30 or ≤ -2

Gross Min/Max - Suspect flag

- Highly unlikely, but not as extreme as above
- Example - water temp values flagged ‘S’ if ≥ 25 or ≤ -1

```
100 ##############################################
101 # QC Tests and Calculations
102 #http://stackoverflow.com/questions/16143700/pa
103 #myNames.QCTests.Calcs.combo <- as.vector(t(out
104 # combine so can check for and remove later.
105 #myNames.DataFields.QCTests.Calcs.combo <- as.v
106 # Data Quality Flag Thresholds
107 ## Gross Min/Max, Fail (equipment)
108 myThresh.Gross.Fail.Hi.WaterTemp   <- 30
109 myThresh.Gross.Fail.Lo.WaterTemp   <- -2
110 myThresh.Gross.Fail.Hi.AirTemp     <- 35
111 myThresh.Gross.Fail.Lo.AirTemp     <- -25
112 myThresh.Gross.Fail.Hi.WaterBP    <- 17
113 myThresh.Gross.Fail.Lo.WaterBP    <- 11
114 myThresh.Gross.Fail.Hi.AirBP      <- 17
115 myThresh.Gross.Fail.Lo.AirBP      <- 11
116 myThresh.Gross.Fail.Hi.WaterLevel <- 6
117 myThresh.Gross.Fail.Lo.WaterLevel <- -1
118 ## Gross Min/Max, Suspect (extreme)
119 myThresh.Gross.Suspect.Hi.WaterTemp   <- 25
120 myThresh.Gross.Suspect.Lo.WaterTemp   <- -1
121 myThresh.Gross.Suspect.Hi.AirTemp     <- 30
122 myThresh.Gross.Suspect.Lo.AirTemp     <- -20
123 myThresh.Gross.Suspect.Hi.WaterBP    <- 16
124 myThresh.Gross.Suspect.Lo.WaterBP    <- 12
125 myThresh.Gross.Suspect.Hi.AirBP      <- 16
126 myThresh.Gross.Suspect.Lo.AirBP      <- 12
127 myThresh.Gross.Suspect.Hi.WaterLevel <- 5
128 myThresh.Gross.Suspect.Lo.WaterLevel <- 0
129 ## Spike thresholds (absolute change)
```

Screenshot showing where you can adjust thresholds for the Gross test in the **fun.UserDefinedValues.R** file (in the Scripts folder)

Thresholds for Spike test (absolute change)

- Difference from one measurement to the next
- Example – water temp values flagged ‘F’ if absolute change is ≥10 and flagged ‘S’ if absolute change is ≥ 5

```
129  ## Spike thresholds (absolute change)
130  myThresh.Spike.Hi.WaterTemp   <- 10
131  myThresh.Spike.Lo.WaterTemp   <- 5
132  myThresh.Spike.Hi.AirTemp     <- 10
133  myThresh.Spike.Lo.AirTemp     <- 5
134  myThresh.Spike.Hi.WaterBP     <- 5
135  myThresh.Spike.Lo.WaterBP     <- 3
136  myThresh.Spike.Hi.AirBP       <- 5
137  myThresh.Spike.Lo.AirBP       <- 3
138  myThresh.Spike.Hi.WaterLevel  <- 5
139  myThresh.Spike.Lo.WaterLevel  <- 3
```

Screenshot showing where you can adjust thresholds for the Spike test in the **fun.UserDefinedValues.R** file (in the Scripts folder)

Thresholds for Rate of Change (RoC) (relative change) test

- Calculates standard deviation (st dev) over 25-hour period, if values differ by ≥ 3 st dev over this period, values flagged ‘S’

```
140 ## Rate of Change (relative change)
141 myDefault.RoC.SD.number    <- 3
142 myDefault.RoC.SD.period    <- 25 #hours
143 myThresh.RoC.SD.number.WaterTemp  <- myDefault.RoC.SD.number
144 myThresh.Roc.SD.period.WaterTemp  <- myDefault.RoC.SD.period
145 myThresh.RoC.SD.number.AirTemp    <- myDefault.RoC.SD.number
146 myThresh.Roc.SD.period.AirTemp    <- myDefault.RoC.SD.period
147 myThresh.RoC.SD.number.WaterBP    <- myDefault.RoC.SD.number
148 myThresh.Roc.SD.period.WaterBP    <- myDefault.RoC.SD.period
149 myThresh.RoC.SD.number.AirBP      <- myDefault.RoC.SD.number
150 myThresh.Roc.SD.period.AirBP      <- myDefault.RoC.SD.period
151 myThresh.RoC.SD.number.WaterLevel <- myDefault.RoC.SD.number
152 myThresh.Roc.SD.period.WaterLevel <- myDefault.RoC.SD.period
153 ## No Change (flat-line)
```

Screenshot showing where you can adjust thresholds for the RoC test in the **fun.UserDefinedValues.R** file (in the Scripts folder)

Thresholds for Flat-line (no change) test

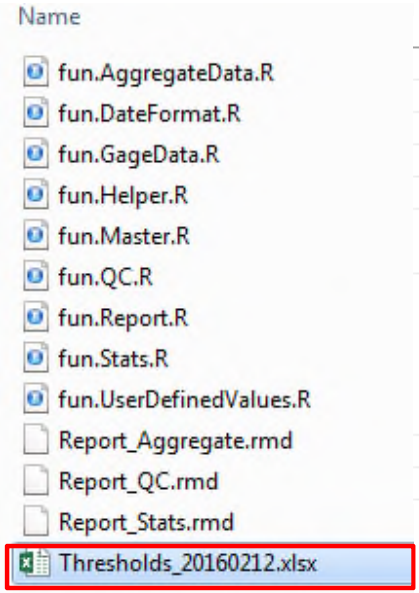
- Looks across **consecutive measurements**
- If 30 measurements in a row are within ≤ 0.01 or 0.001 of one another (depending on the parameter), values are flagged ‘F’
- If 10 measurements in a row are within ≤ 0.01 or 0.001 of one another (depending on the parameter), values are flagged ‘S’

```
152 myThresh.RoC.SD.number.AirBP       <- myDefault.RoC.SD.number
153 myThresh.Roc.SD.period.AirBP       <- myDefault.RoC.SD.period
154 myThresh.RoC.SD.number.WaterLevel <- myDefault.RoC.SD.number
155 myThresh.Roc.SD.period.WaterLevel <- myDefault.RoC.SD.period
156 ## No Change (flat-line)
157 myDefault.Flat.Hi          <- 30   # maximum is myThresh.Flat.Max
158 myDefault.Flat.Lo          <- 10
159 myDefault.Flat.Tolerance   <- 0.01 # set to one sigdig less than
160 myThresh.Flat.Hi.WaterTemp          <- myDefault.Flat.Hi
161 myThresh.Flat.Lo.WaterTemp          <- myDefault.Flat.Lo
162 myThresh.Flat.Tolerance.WaterTemp   <- 0.01
163 myThresh.Flat.Hi.AirTemp           <- myDefault.Flat.Hi
164 myThresh.Flat.Lo.AirTemp           <- myDefault.Flat.Lo
165 myThresh.Flat.Tolerance.AirTemp    <- 0.01
166 myThresh.Flat.Hi.WaterP           <- myDefault.Flat.Hi
167 myThresh.Flat.Lo.WaterP           <- myDefault.Flat.Lo
168 myThresh.Flat.Tolerance.WaterP    <- 0.001
169 myThresh.Flat.Hi.AirBP            <- myDefault.Flat.Hi
170 myThresh.Flat.Lo.AirBP            <- myDefault.Flat.Lo
171 myThresh.Flat.Tolerance.AirBP     <- 0.001
172 myThresh.Flat.Hi.WaterLevel       <- myDefault.Flat.Hi * 2
173 myThresh.Flat.Lo.WaterLevel       <- myDefault.Flat.Lo * 2
174 myThresh.Flat.Tolerance.WaterLevel <- 0.0
175 myThresh.Flat.MaxComp     <- max(myThresh.Flat.Hi.WaterTemp
176                                 ,myThresh.Flat.Hi.AirTemp
177                                 ,myThresh.Flat.Hi.WaterP
178                                 ,myThresh.Flat.Hi.AirBP
179                                 ,myThresh.Flat.Hi.WaterLevel)
```

Screenshot showing where you can adjust thresholds for the flat-line test in the **fun.UserDefinedValues.R** file (in the Scripts folder)

Thresholds 'translator' file

We added an Excel file into the Scripts folder to help make it easier to interpret the settings in the fun.UserDefinedValues.R script.



1	**Adjusted Table uses formulas on the next page**						
2	So can modify the units and the table auto updates						
3		**Water Temp**		**Air Temp**		**Water Level**	
4		**Fail**	**Suspect**	**Fail**	**Suspect**	**Fail**	**Suspect**
5	Gross	>30°C or <-2°C	>30°C or <-2°C	>35°C or <-25°C	>35°C or <-25°C	>6 ft or <-1 ft	>6 ft or <-1 ft
6	Spike	≥10°C (±)	≥5°C (±)	≥10°C (±)	≥5°C (±)	≥5 ft (±)	≥3 ft (±)
7	Rate of Change	NA	≥3 stdev within 25 hrs	NA	≥3 stdev within 25 hrs	NA	≥3 stdev within 25 hrs
8	Flat Line	>30 consecutive measurements within 0.01 units of one another	>10 consecutive measurements within 0.01 units of one another	>30 consecutive measurements within 0.01 units of one another	>10 consecutive measurements within 0.01 units of one another	>60 consecutive measurements within 0 units of one another	>20 consecutive measurements within 0 units of one another
9							
10	*gross test for water level would need to be adjusted for each site						

This file has 2 worksheets. Shown here is the Table worksheet, which 'translates' the thresholds. If you make changes in the fun.UserDefinedValues.R file, enter the changes in the 'ScriptValues' worksheet, and it will automatically update the 'Table' worksheet with the new 'translated' thresholds.

United States Environmental Protection Agency

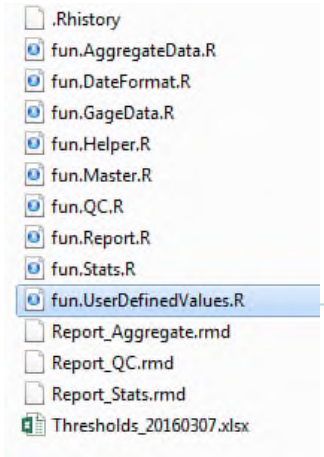
How to handle flagged data?

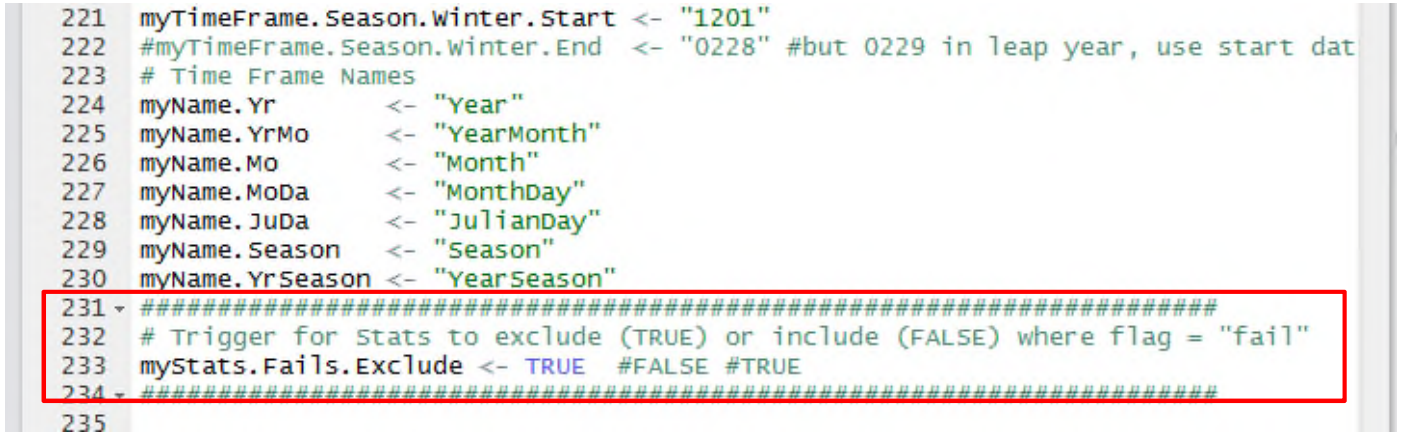
Are data that are flagged as fail (F) included in the SummaryStat calculations?

The scripts are written in a way that gives you flexibility on this. In the last line (line 233) of the **fun.UserDefinedValues.R file** in the **Scripts folder**, if you enter –

- TRUE – the script will exclude values flagged as ‘F’
- FALSE – the script will include values flagged as ‘F’

Scripts folder





Right now the script you are working with is set to ‘TRUE’

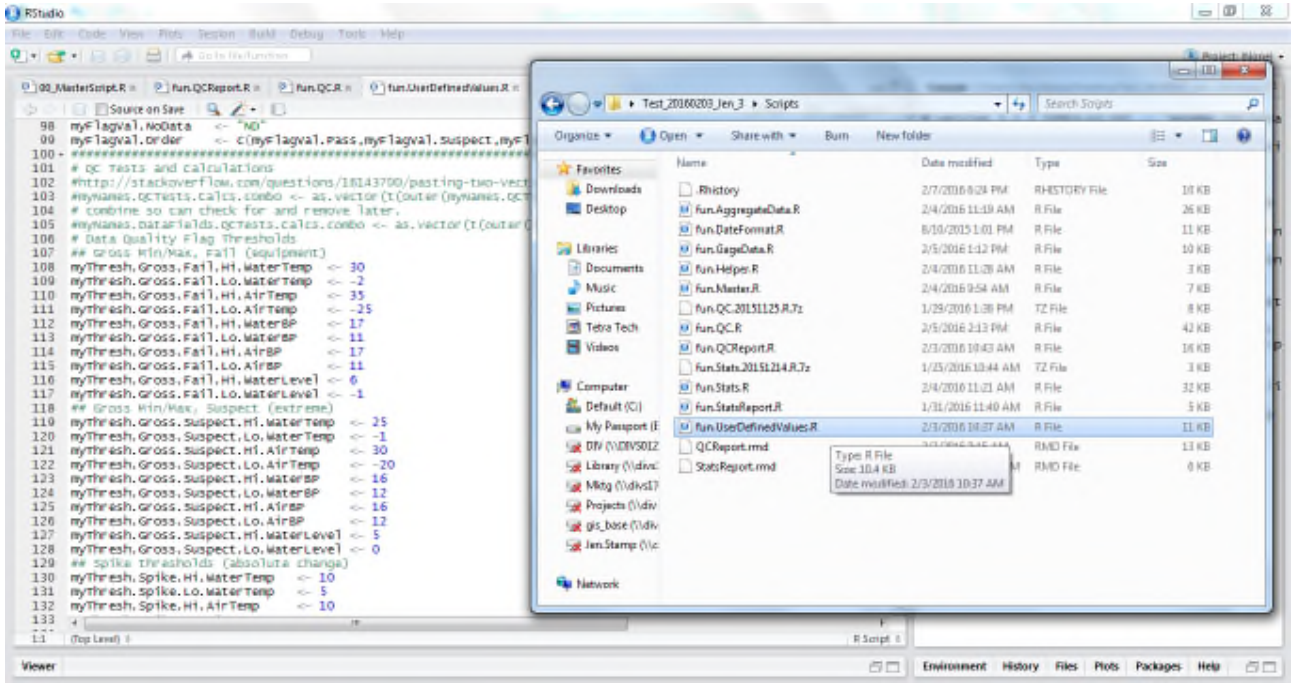
EPA
United States
Environmental Protection
Agency

Adding new variables

Adding variables

If collecting data for parameters not accounted for in the scripts the user can add them to the “fun.UserDefinedValues.R” and “fun.QC.R” scripts and the new parameters will be assessed by the QC, Aggregate, and Stats scripts.

Open fun.UserDefinedValues.R script



Adding variables

Example with “discharge”; text will reference “xyz”.

United States Environmental Protection Agency

Names

Need units (myUnits.xyz), name (myName.xyz), and label (myLab.xyz). Field names ,including units, cannot contain special characters (e.g., %, space, or /).

```
18 ##########################################################################
19 # Acceptable column names for the data
20 #(special characters (e.g., %, space, or /) are converted to "." by R, "°" converted to "
21 myUnits.AirTemp      <- "C" # C or F
22 myUnits.WaterTemp    <- myUnits.AirTemp
23 myUnits.AirBP        <- "psi"
24 myUnits.WaterP       <- myUnits.AirBP
25 myUnits.WaterLevel   <- "ft"
26 myUnits.Discharge    <- "ft3.s"
27 ## Air and Water
28 myName.SiteID          <- "SiteID"
29 myName.Date            <- "Date"
30 myName.Time            <- "Time"
31 myName.DateTime        <- "Date.Time"
32 ## Water
33 myName.RowID.Water     <- "Water.RowID"
34 myName.LoggerID.Water  <- "Water.LoggerID"
35 myName.WaterTemp       <- paste("Water.Temp.",myUnits.WaterTemp,sep="")  # "°" from HoboWa
36 ## Air
37 myName.RowID.Air       <- "Air.RowID"
38 myName.LoggerID.Air    <- "Air.LoggerID"
39 myName.AirTemp         <- paste("Air.Temp.",myUnits.AirTemp,sep="")    # "°" from HoboWare
40 myName.WaterP          <- paste("Water.BP.",myUnits.WaterP,sep="")
41 myName.AirBP           <- paste("Air.BP.",myUnits.WaterP,sep="")
42 myName.WaterLevel      <- paste("Water.Level.",myUnits.WaterLevel,sep="")
43 myName.Discharge       <- paste("Discharge.",myUnits.Discharge,sep="")
44 ## plot labels
45 myLab.WaterTemp        <- paste("Temperature, Water (°",myUnits.WaterTemp,")",sep="")
46 myLab.AirTemp          <- paste("Temperature, Air (°",myUnits.AirTemp,")",sep="")
47 myLab.Date             <- "Date"
48 myLab.DateTime         <- "Date"
49 myLab.WaterP           <- paste("Pressure, Water (",myUnits.AirBP,")",sep="")
50 myLab.AirBP            <- paste("Barometric Pressure, Air (",myUnits.WaterP,")",sep="")
51 myLab.WaterLevel       <- paste("Water Level (",myUnits.WaterLevel,")",sep="")
52 myLab.Temp.BOTH        <- paste("Temperature (°",myUnits.WaterTemp,")",sep="")
53 myLab.Discharge        <- paste("Discharge (ft³/s)")
54 ##########################################################################
```

Adding variables

United States Environmental Protection Agency

Example with “discharge”; text will reference “xyz”.

Discrete Measurements

Need to add name (myName.Discrete.xyz) and label (myLab.Discrete.xyz).

```
54  ##################################################################
55  # Discrete Measurements
56  myPrefix.Discrete           <- "Discrete"
57  myName.Discrete.WaterTemp   <- paste(myPrefix.Discrete,myName.WaterTemp,sep=".")
58  myName.Discrete.AirTemp     <- paste(myPrefix.Discrete,myName.AirTemp,sep=".")
59  myName.Discrete.WaterP      <- paste(myPrefix.Discrete,myName.WaterP,sep=".")
60  myName.Discrete.AirBP       <- paste(myPrefix.Discrete,myName.AirBP,sep=".")
61  myName.Discrete.WaterLevel  <- paste(myPrefix.Discrete,myName.WaterLevel,sep=".")
62  myName.Discrete.Discharge   <- paste(myPrefix.Discrete,myName.Discharge,sep=".")
63  myLab.Discrete.WaterTemp    <- paste(myLab.WaterTemp,"(Discrete)",sep=" ")
64  myLab.Discrete.AirTemp      <- paste(myLab.AirTemp,"(Discrete)",sep=" ")
65  myLab.Discrete.WaterP       <- paste(myLab.WaterP,"(Discrete)",sep=" ")
66  myLab.Discrete.AirBP        <- paste(myLab.AirBP,"(Discrete)",sep=" ")
67  myLab.Discrete.WaterLevel   <- paste(myLab.WaterLevel,"(Discrete)",sep=" ")
68  myLab.Discrete.Discharge    <- paste(myLab.Discharge,"(Discrete)",sep=" ")
69  ##################################################################
```

Data Fields

Add in “myName.xyz”, “myName.Discrete.xyz”, and a new color (in the right order).

```
83  ##################################################################
84  # Data Fields
85  myNames.DataFields <- c(myName.WaterTemp,myName.AirTemp,myName.WaterP,myName.AirBP,myName.WaterLevel,myName.Discharge
86                          ,myName.Discrete.WaterTemp,myName.Discrete.AirTemp,myName.Discrete.WaterP,myName.Discrete.AirBP,myName.Discrete.WaterLevel,myName.Discrete.Discharge)
87  myNames.DataFields.Lab <- c(myLab.WaterTemp,myLab.AirTemp,myLab.WaterP,myLab.AirBP,myLab.WaterLevel,myLab.Discharge
88                              ,myLab.Discrete.WaterTemp,myLab.Discrete.AirTemp,myLab.Discrete.WaterP,myLab.Discrete.AirBP,myLab.Discrete.WaterLevel,myLab.Discrete.Discharge)
89  myNames.DataFields.Col <- c("blue","green","gray","gray","black","brown")
90  #
91  ## Name Order (change order below to change order in output file)
92  myNames.Order <- c(myName.SiteID,myName.Date,myName.Time,myName.DateTime,myName.WaterTemp,myName.LoggerID.Air,myName.RowID.Air
93                     ,myName.AirTemp,myName.WaterP,myName.AirBP,myName.WaterLevel,myName.LoggerID.Water,myName.RowID.Water,myName.Discharge
94                     ,myName.Discrete.WaterTemp,myName.Discrete.AirTemp,myName.Discrete.WaterP,myName.Discrete.AirBP,myName.Discrete.WaterLevel,myName.Discrete.Discharge)
95  ##################################################################
```

Adding variables

Example with “discharge”; text will reference “xyz”.

United States Environmental Protection Agency

QC Test

Add thresholds for each of the QC checks. Each of the test has a Fail and Suspect and High and Low variable that is used for flagging during the QC check.

- Gross
  - myThresh.Gross.Fail.Hi.xyz
  - myThresh.Gross.Fail.Lo.xyz
  - myThresh.Gross.Suspect.Hi.xyz
  - myThresh.Gross.Suspect.Lo.xyz

Do this for each test – Gross, spike, rate of change, flatline

```
110  ## Gross Min/Max, Fail (equipment)
111  myThresh.Gross.Fail.Hi.WaterTemp  <- 30
112  myThresh.Gross.Fail.Lo.WaterTemp  <- -2
113  myThresh.Gross.Fail.Hi.AirTemp    <- 35
114  myThresh.Gross.Fail.Lo.AirTemp    <- -25
115  myThresh.Gross.Fail.Hi.WaterP     <- 17
116  myThresh.Gross.Fail.Lo.WaterP     <- 11
117  myThresh.Gross.Fail.Hi.AirBP      <- 17
118  myThresh.Gross.Fail.Lo.AirBP      <- 11
119  myThresh.Gross.Fail.Hi.WaterLevel <- 6    # no longer used (only check for negative values for WaterLevel)
120  myThresh.Gross.Fail.Lo.WaterLevel <- -1   # no longer used (only check for negative values for WaterLevel)
121  myThresh.Gross.Fail.Hi.Discharge <-  10^5 #dependant upon stream size
122  myThresh.Gross.Fail.Lo.Discharge <-  -1   #dependant upon stream size
123  ## Gross Min/Max, Suspect (extreme)
124  myThresh.Gross.Suspect.Hi.WaterTemp  <- 25
125  myThresh.Gross.Suspect.Lo.WaterTemp  <- -1
126  myThresh.Gross.Suspect.Hi.AirTemp    <- 30
127  myThresh.Gross.Suspect.Lo.AirTemp    <- -20
128  myThresh.Gross.Suspect.Hi.WaterP     <- 16
129  myThresh.Gross.Suspect.Lo.WaterP     <- 12
130  myThresh.Gross.Suspect.Hi.AirBP      <- 16
131  myThresh.Gross.Suspect.Lo.AirBP      <- 12
132  myThresh.Gross.Suspect.Hi.WaterLevel <- 5    # no longer used (only check for negative values for WaterLevel)
133  myThresh.Gross.Suspect.Lo.WaterLevel <- 0    # no longer used (only check for negative values for WaterLevel)
134  myThresh.Gross.Suspect.Hi.Discharge <-  10^3 #dependant upon stream size
135  myThresh.Gross.Suspect.Lo.Discharge <-  -1   #dependant upon stream size
```

Adding variables

Example with “discharge”; text will reference “xyz”.

Flags

myName.Flag.xyz

```
182 ##################################################################
183 # Data Fields with Flags
184 myName.Flag <- "Flag" # flag prefix
185 myNames.Cols4Flags <- c(myName.DateTime,myNames.DataFields)
186 myNames.Flags <- paste(myName.Flag,myNames.Cols4Flags,sep=".")  # define ones using in the calling script
187 ## flag labels
188 myName.Flag.DateTime     <- paste(myName.Flag,myName.DateTime,sep=".")
189 myName.Flag.WaterTemp    <- paste(myName.Flag,myName.WaterTemp,sep=".")
190 myName.Flag.AirTemp      <- paste(myName.Flag,myName.AirTemp,sep=".")
191 myName.Flag.WaterP       <- paste(myName.Flag,myName.WaterP,sep=".")
192 myName.Flag.AirBP        <- paste(myName.Flag,myName.AirBP,sep=".")
193 myName.Flag.WaterLevel   <- paste(myName.Flag,myName.WaterLevel,sep=".")
194 myName.Flag.Discharge    <- paste(myName.Flag,myName.Discharge,sep=".")
195 # Data Quality Test Names
196 myNames.QCTests <- c("Gross","Spike","RoC","Flat")
197 myNames.QCCalcs <- c("SD.Time","SD","SDxN",paste("n",1:myThresh.Flat.MaxComp,sep="."),"flat.Lo","flat.Hi")
198 ##################################################################
```

Adding variables

Example with “discharge”; text will reference “xyz”.

Fun.QC.R

Need to add in section to perform the QC on the new variable. Copy another section and change the names.

```
529     # 6.5. Discharge
530     myField <- myName.Discharge
531     myMsg.data <- "Discharge"
532     myMsg <- paste("WORKING (QC Tests and Flags - ",myMsg.data,")",sep="")
533     myItems.Complete <- myItems.Complete + 1
534     myItems.Log[intCounter,2] <- myMsg
535     fun.Msg.Status(myMsg, intCounter, intItems.Total, strFile)
536     flush.console()
537     if(myField %in% myNames.DataFields.Present==TRUE){##IF.START
538       #
539       data.import <- fun.CalcQCStats(data.import
540                                      ,myField
541                                      ,myThresh.Gross.Fail.Hi.Discharge
542                                      ,myThresh.Gross.Fail.Lo.Discharge
543                                      ,myThresh.Gross.Suspect.Hi.Discharge
544                                      ,myThresh.Gross.Suspect.Lo.Discharge
545                                      ,myThresh.Spike.Hi.Discharge
546                                      ,myThresh.Spike.Lo.Discharge
547                                      ,myThresh.RoC.SD.period.Discharge
548                                      ,myThresh.RoC.SD.number.Discharge
549                                      ,myThresh.Flat.Hi.Discharge
550                                      ,myThresh.Flat.Lo.Discharge
551                                      ,myThresh.Flat.Tolerance.Discharge)
552       #
553     }##IF.END
```

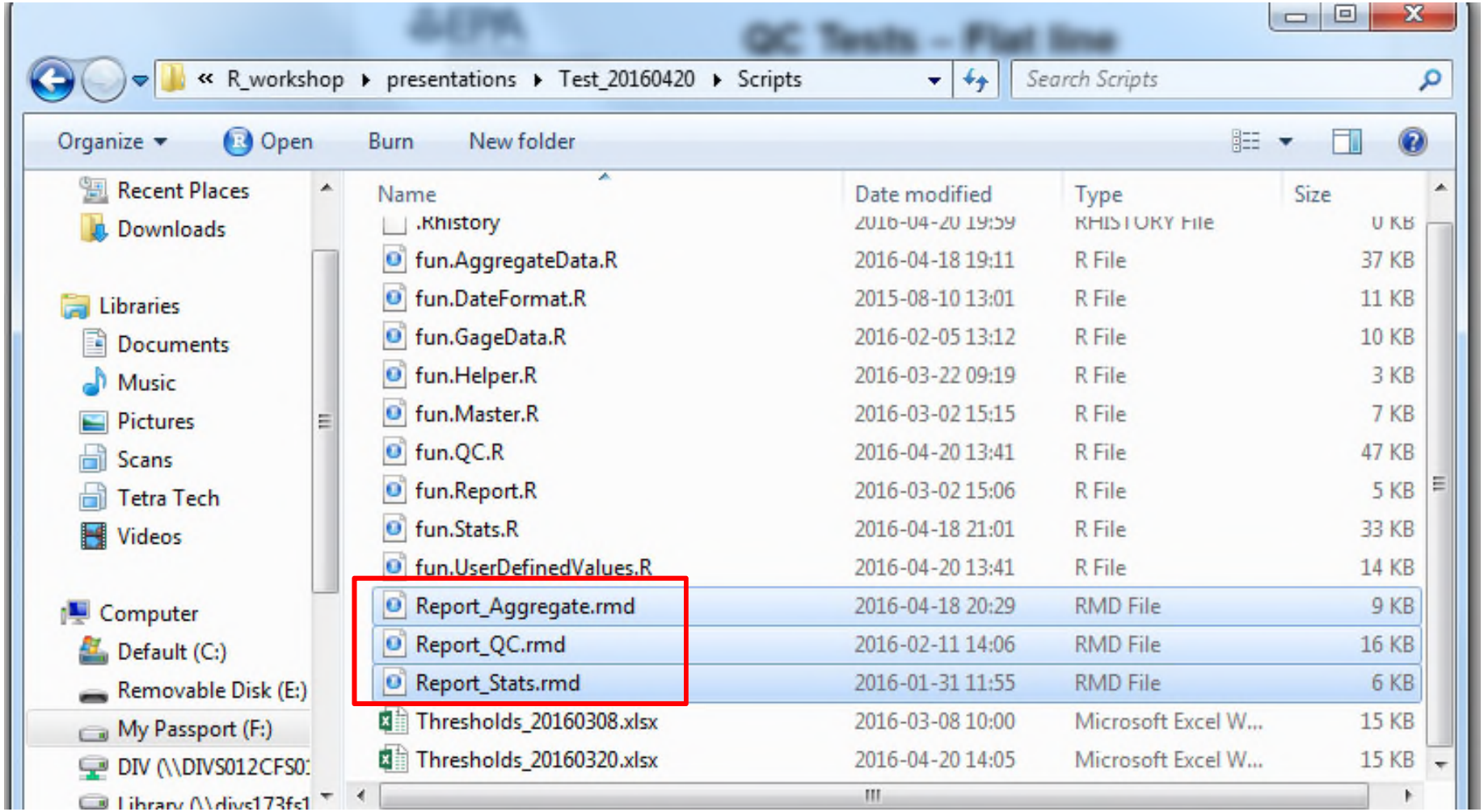
EPA United States Environmental Protection Agency

Potential places to modify the R scripts

Rmarkdown (rmd) files

- Reports (QC, stats, aggregate)

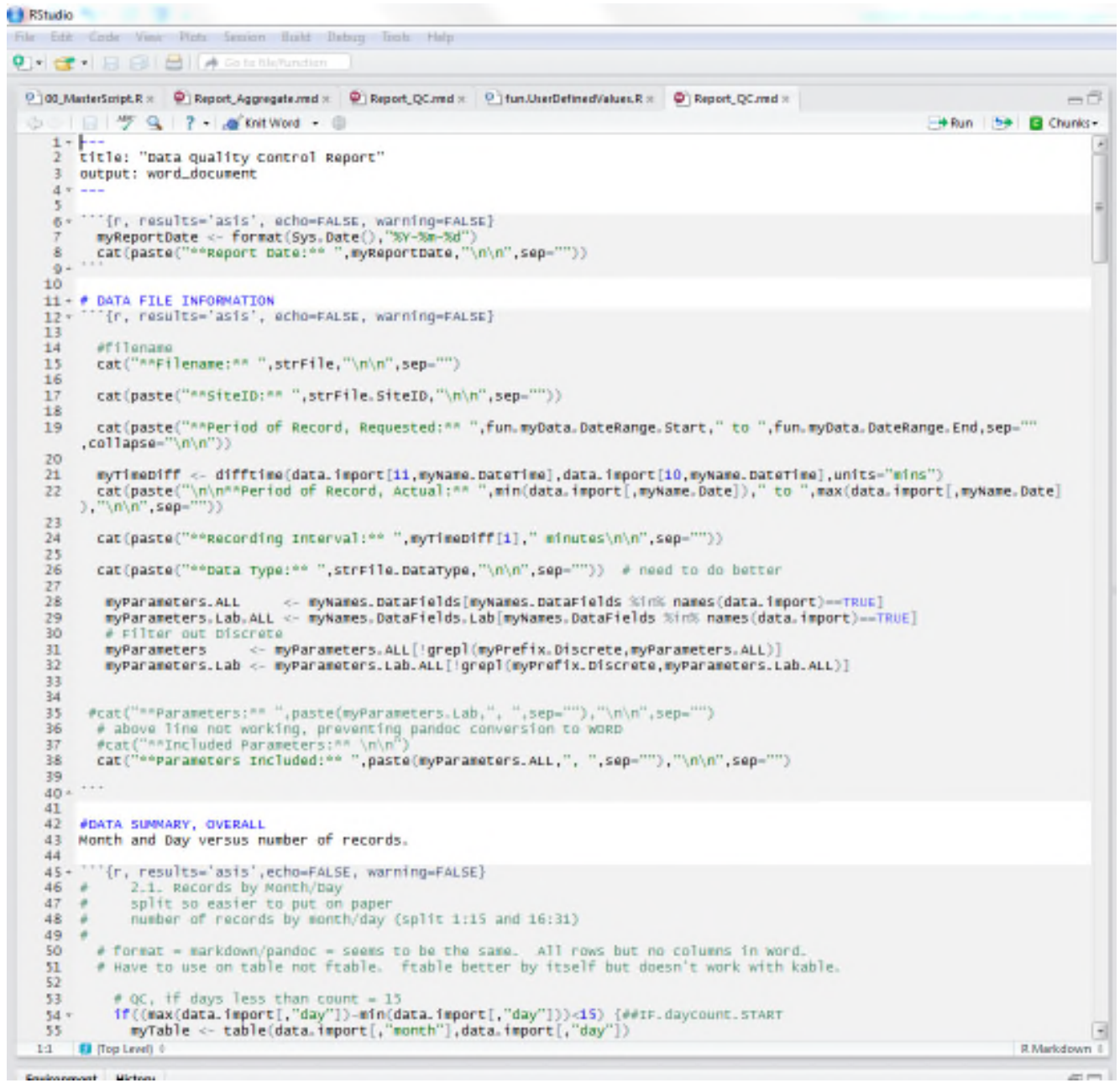
You can make modifications to the reports in the rmd files (QC, aggregate, stats)



United States Environmental Protection Agency

Example - Report_QC file

Double click on the file. The screen will look like this…



EPA United States Environmental Protection Agency

Seeking feedback

- Wish list – what additional features and functionality would you like these scripts to have?
- Are the scripts working properly? If not, please describe the errors you are getting (screenshots are great).
- Are the **instruction slides** detailed enough? How can we improve them?

QUESTIONS? COMMENTS?

Britta Bierwagen (bierwagen.britta@epa.gov)

Erik W. Leppo (Erik.Leppo@tetratech.com)

Jen Stamp (Jen.Stamp@tetratech.com)

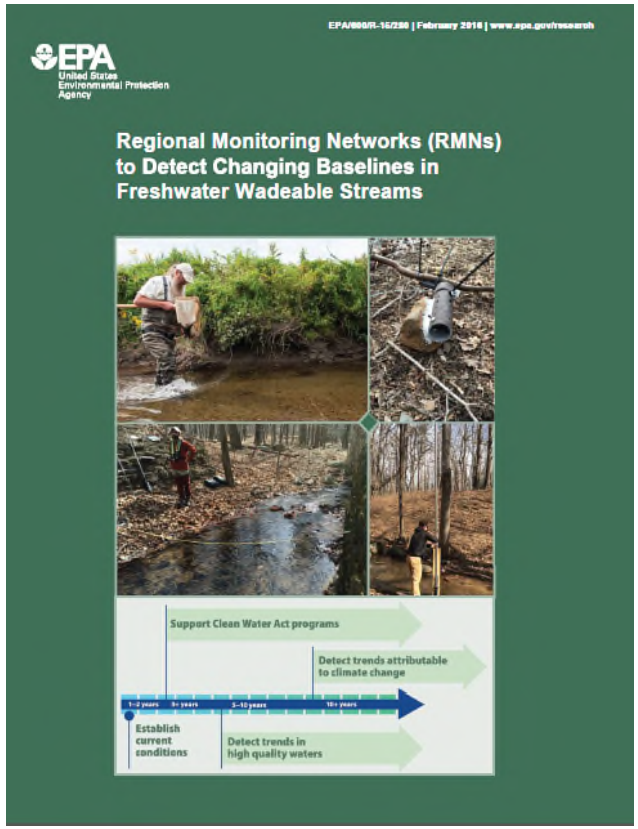
Anna Hamilton (Anna.Hamilton@tetratech.com)

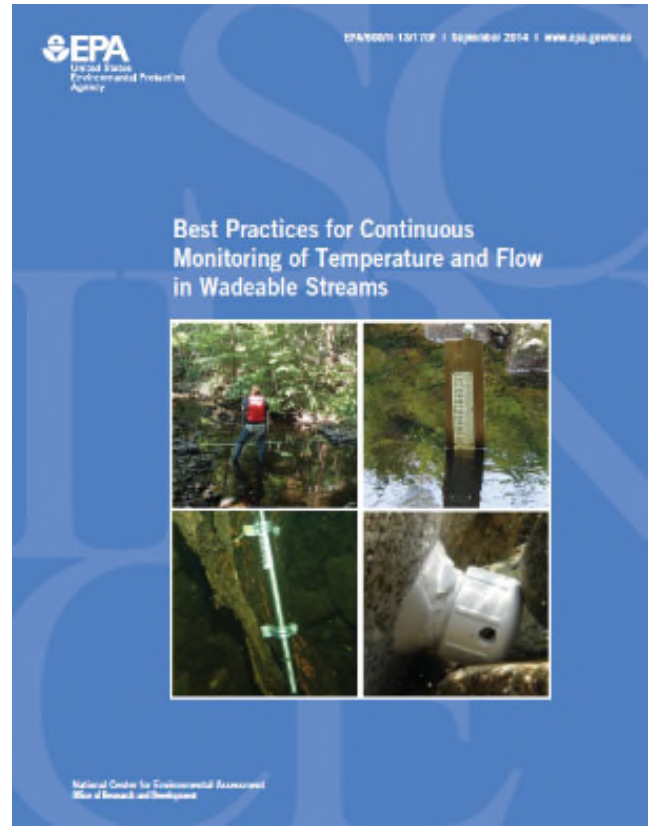
EPA
United States
Environmental Protection
Agency

EXTRAS

Additional Background

RMN and “Best Practices” reports





EPA United States Environmental Protection Agency

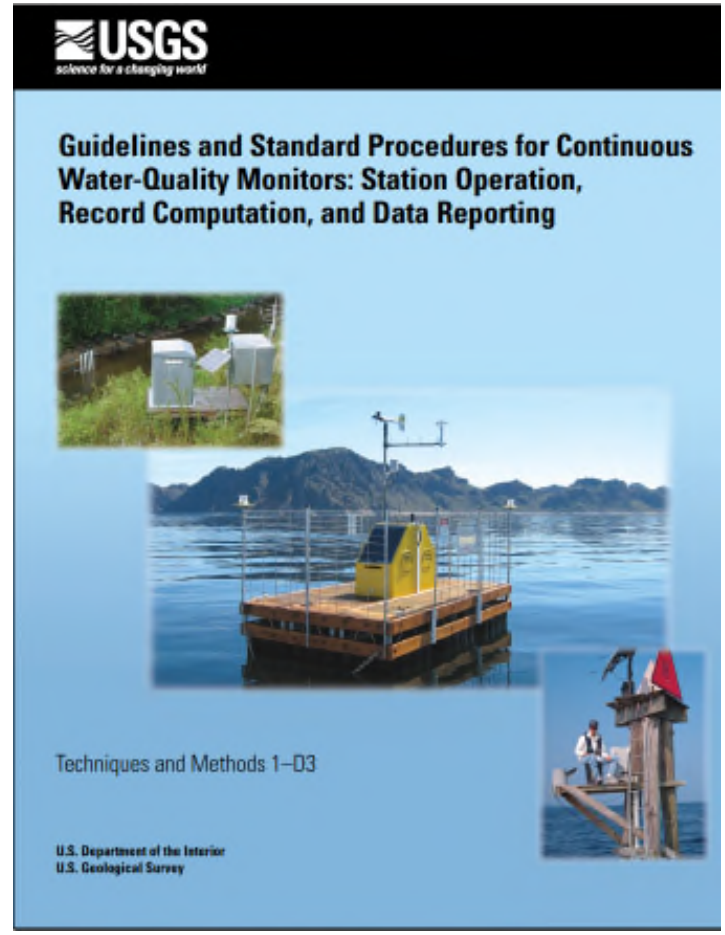
Additional R Resources (not required, just FYI)

- The R Book
  - Michael Crawley. 2007. Wiley and Sons.
- R Programming – Johns Hopkins / Coursera
  - https://www.coursera.org/course/rprog
- An Introduction to R
  - W. N. Venables, D. M. Smith and the R Development Core Team. 2012.
  - http://cran.r-project.org/doc/manuals/R-intro.pdf
- Statistical Training and Consulting Using R
  - Paul Geissler
  - http://paulrstat.com/Reference.aspx
- Introductory Statistics with R
  - Peter Dalgaard. 2008. Springer, New York

Automated QC report - sources/ideas

USGS Data Processing Procedures

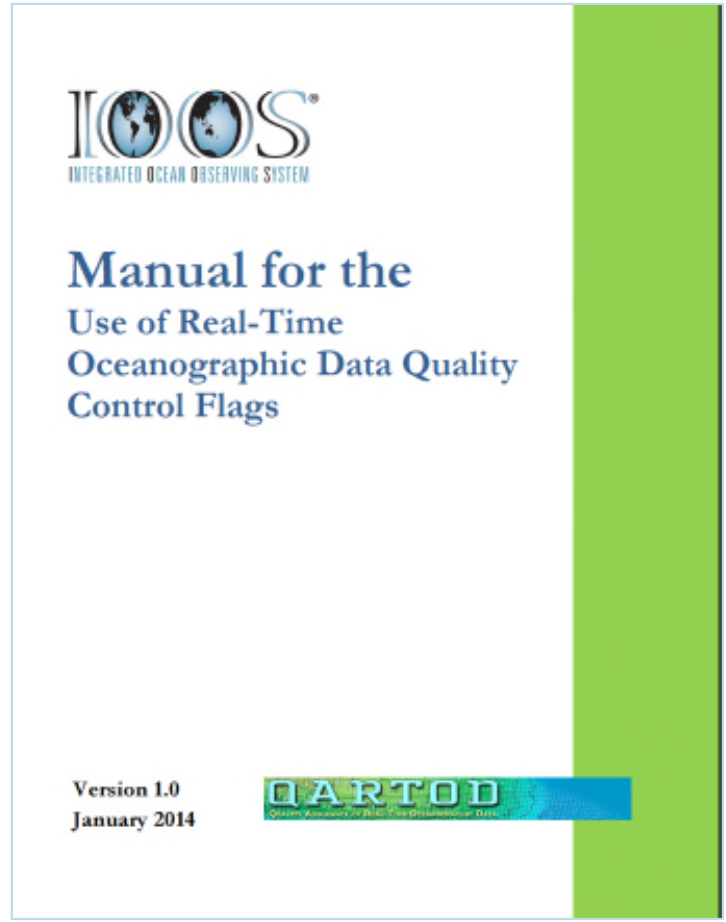
- Initial data evaluation
- Application of data corrections
- Application and evaluation of cross-section corrections
- Final data evaluation
- Record computation
- Final record review



Automated QC report - sources/ideas

NOAA QARTOD

- Timing/gap test
- Gross range test
  - High and low values
  - Real-world and Equipment
- Spike test
  - High and low thresholds
  - Compare adjacent points
- Rate of change test
  - N_DEV (default 3)
  - Time_Dev (default 25 hour)
  - Compare adjacent points
- Flat line test
  - Adjacent points
  - Repeat count (within a given tolerance)
  - Fail (5) and suspect (3)



EPA
United States
Environmental Protection
Agency

Having trouble getting the Aggregate script to run after you make edits and resave the files in the Data2_QC folder?

Troubleshooting potential problem after resaving .csv files in the Data2_QC

We discovered that on some computers, Excel changes the date format in the .csv file if you make a change to the file in the QC folder and resave it. We believe this has to do with the settings on Erik's computer; he wrote the scripts, and the scripts utilize his date time settings (which aren't the same across computers).

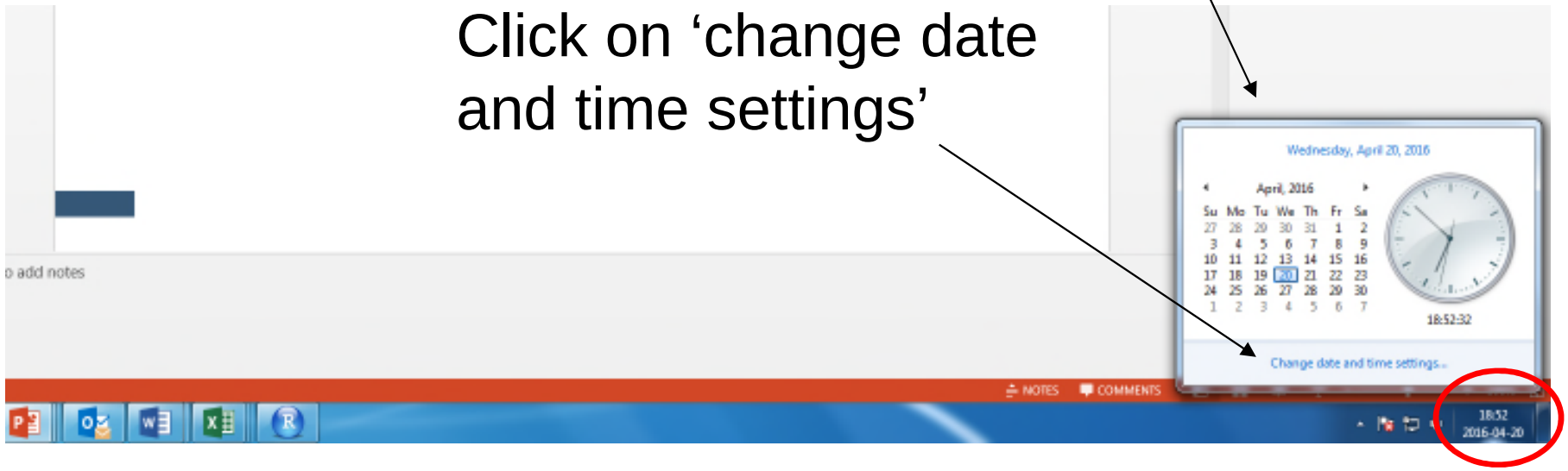
Unfortunately this issue prevents the Aggregate script from running.

The best solution we've come up with at this point is to reset the date-time setting on your computer to match with his. The next slides provide instructions on how to do this.

Left click on the time-date in the lower right hand corner of your computer screen.

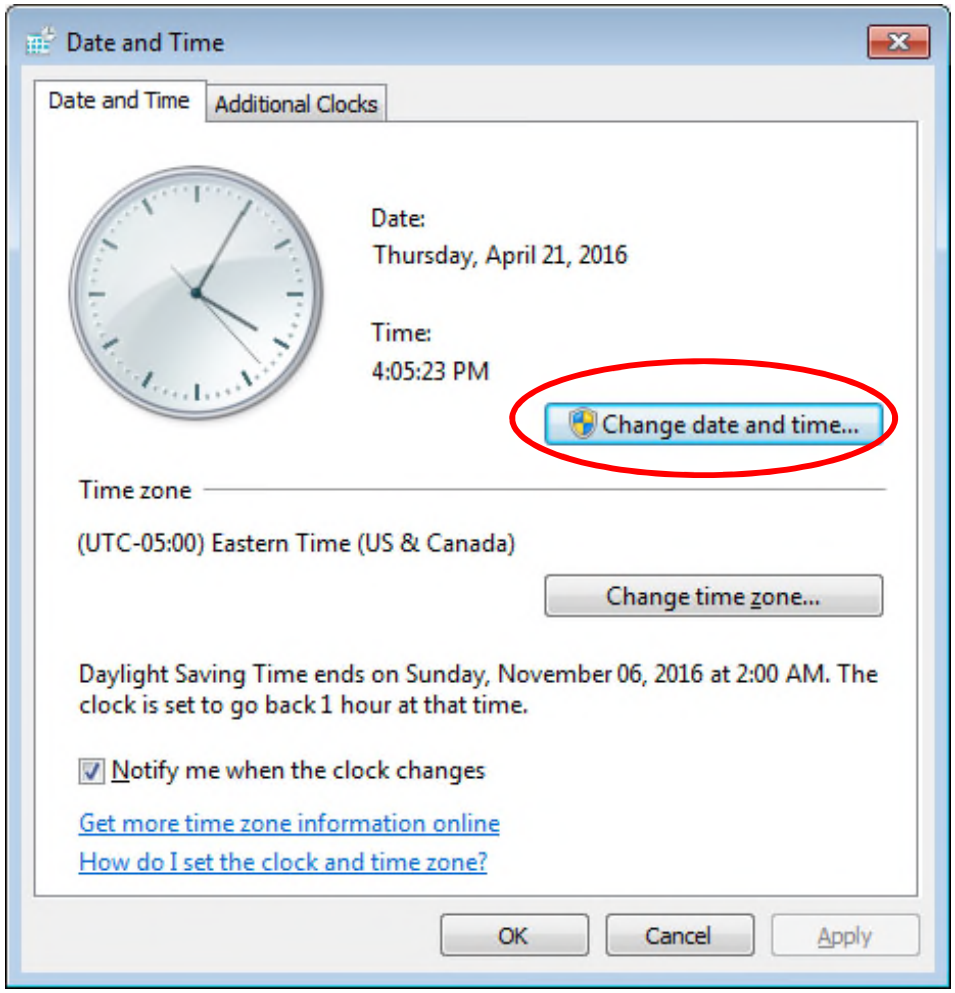
This box will then appear…

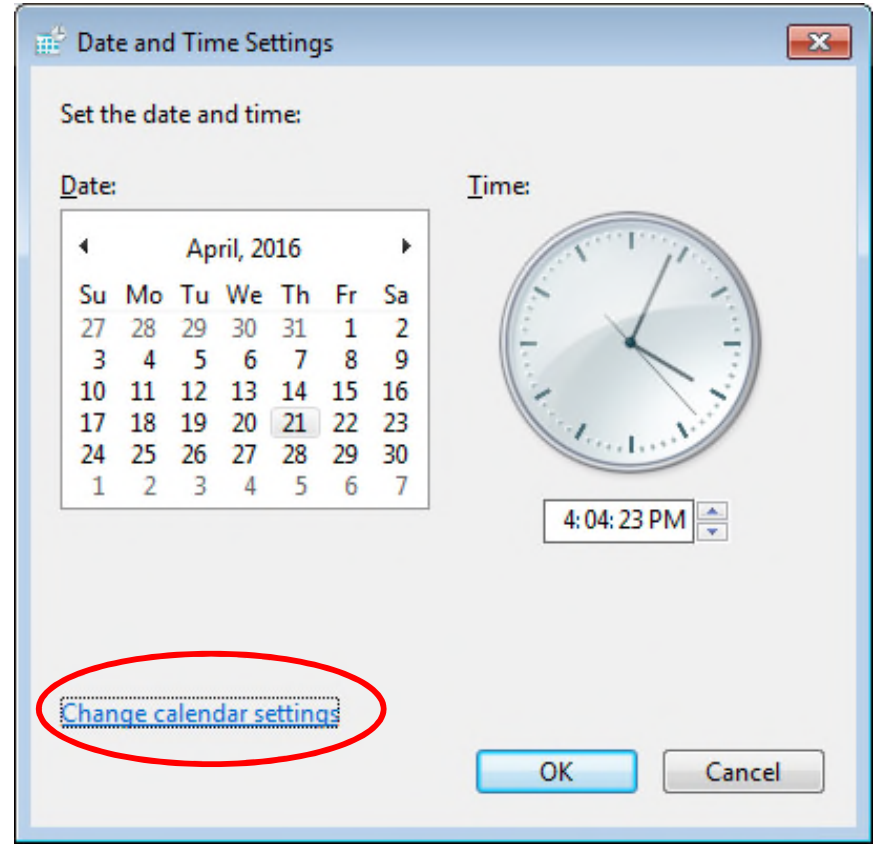
Click on ‘change date and time settings’



Left click here

Click ‘Change date and time…’
Then click ‘Change calendar settings’

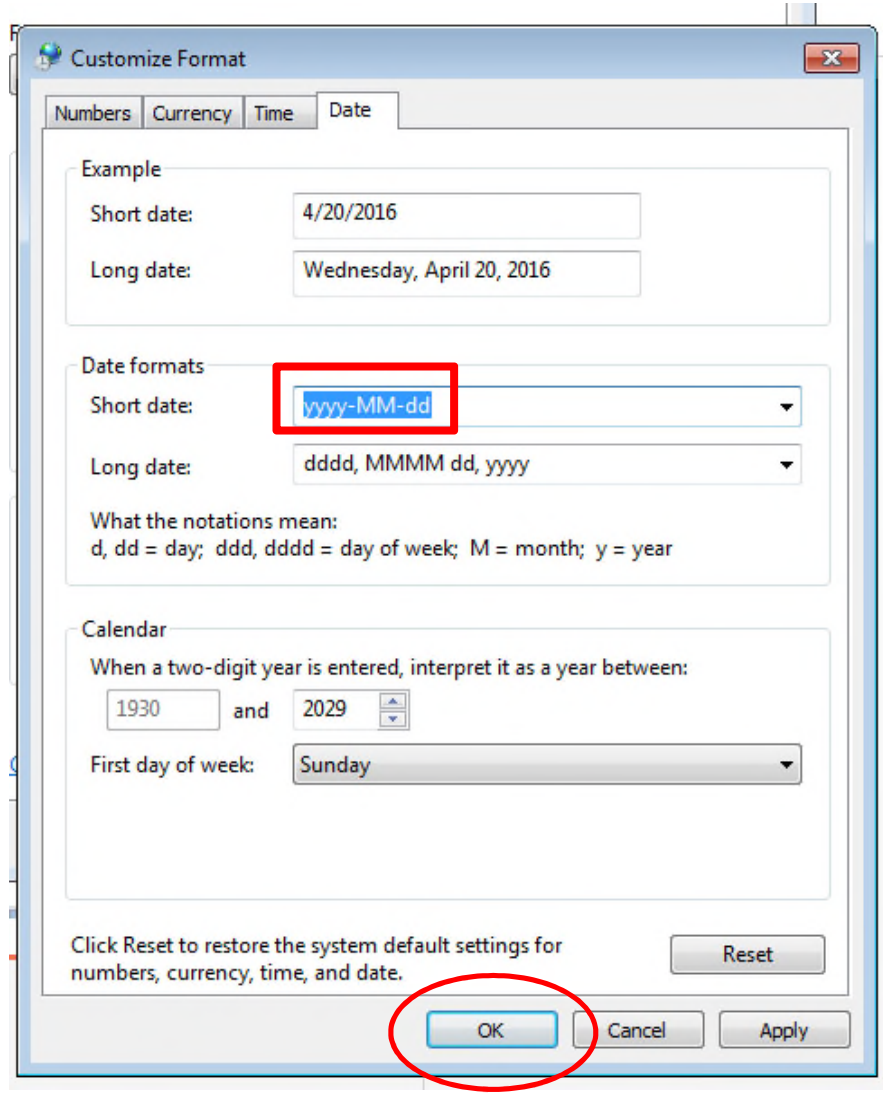




Customize Format

- Click on the **Date** tab
- For ‘Short date’ format enter **yyyy-MM-dd**
- Leave ‘Long date’ format as is

Note: you may need administrator privileges to do this…

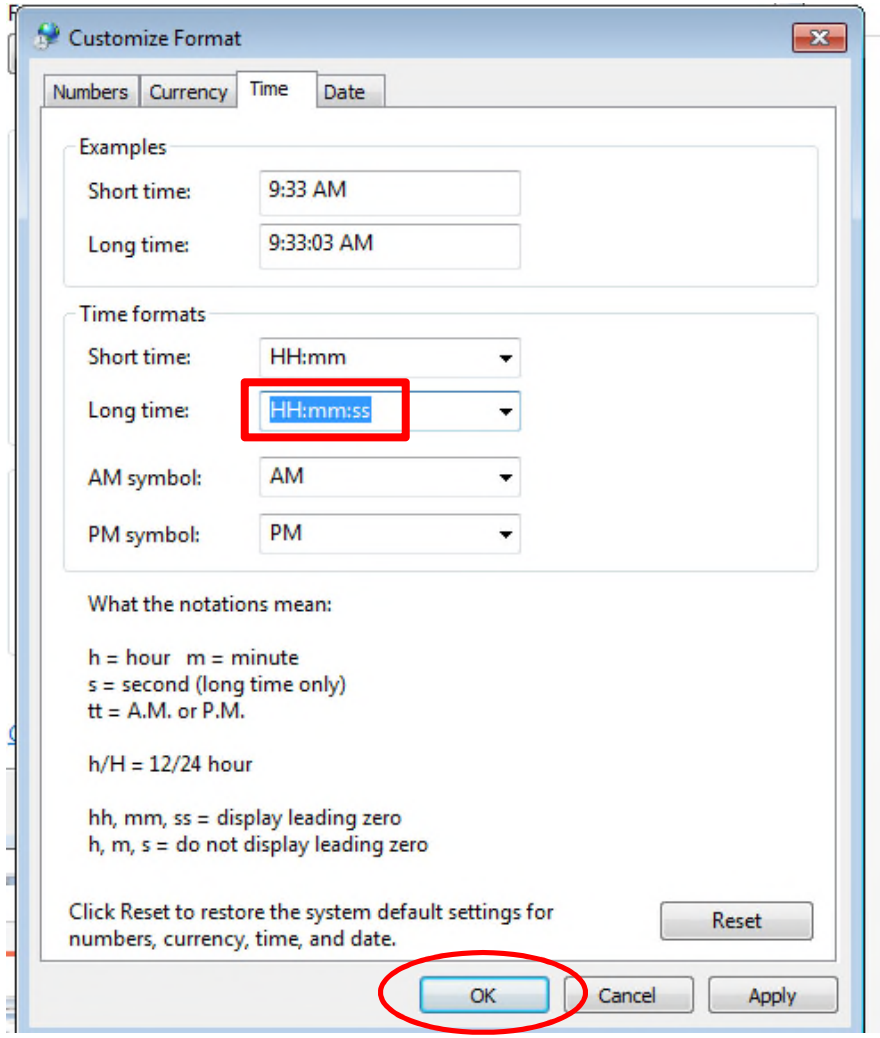


Then click ‘OK’

Customize Format

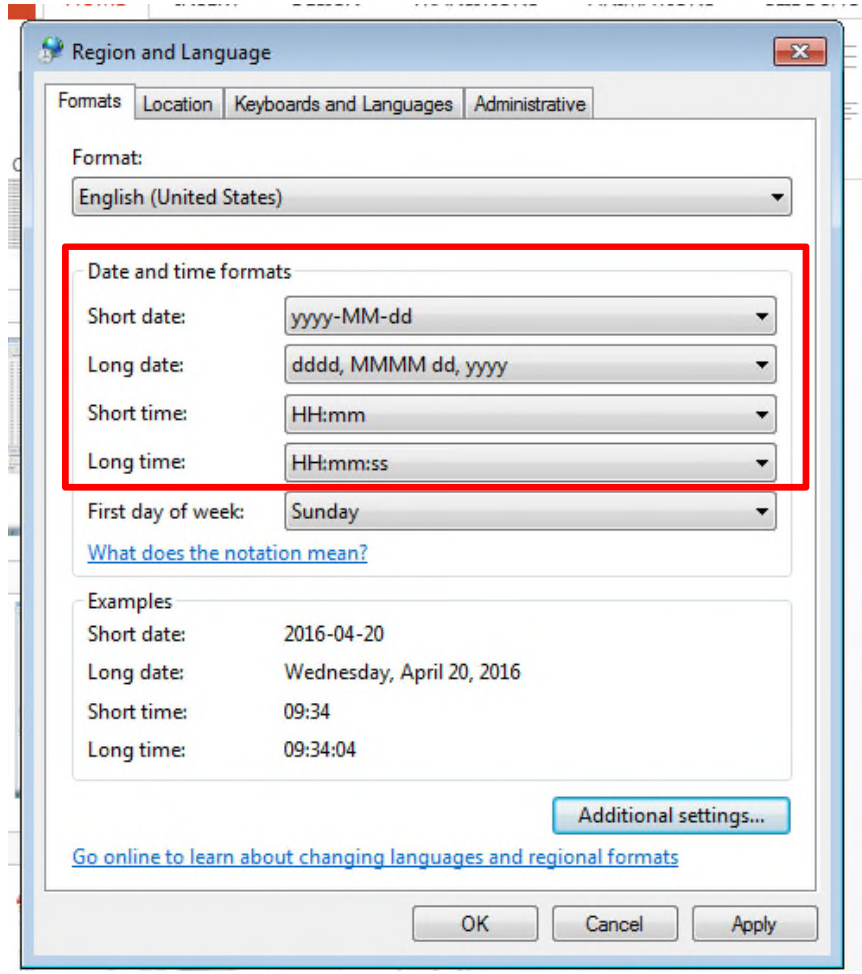
- Click on the **Time** tab
- For ‘Short time’ format enter **HH:mm**
- For ‘Long time’ format enter **HH:mm:ss**

Note: you may need administrator privileges to do this…



Then click ‘OK’

Date and time formats should then look like this



And this…