**Climate Data Collation Instructions (new dataset)**

1. Open the blank template **LTMN\_AWS\_raw\_data\_template\_2020** and save a new version before you make changes. The correct naming convention is **LTMN\_AWS\_<site>\_raw\_data**.
2. The **LTMN Weather Data Tracker** is a useful reference for what files of data exist for each site, and what time periods they cover. All the relevant files should be in the site folder on the shared drive: **N:\Exception\_ LTMN Working Data\01\_Raw\_Data\\_All\_sites\Climate\\_All\_sites\_collation\_up\_to\_2020**
3. **Paste your data** into the correct columns of the raw template. Explanation for the column headings can be found on the Lookup tab, and you can also use this to fill in the **Site** and **Data\_Source**. The column headers **must not be changed**, as the QA R script relies on them being exactly as they appear in the raw template.
4. Not every dataset contains all the variables. Just leave these columns blank, but don’t delete them. Some datasets have columns for ALBEDOSK, ALBEDOGR and SUF\_WET, but these columns should contain no data (blank cells or -6999). These do not get copied across to the raw template.
5. If the data you’re copying from has QA Flags already, don’t copy those columns across.
6. Check for and remove any blank rows.
7. Check that the **units used** in your source data are the **same** as those specified in the template. If they’re not, you may need to do some conversions.
8. Format the date as **dd/mm/yyyy**, with **hh:mm** in a separate column. This can be a little tricky, if your data has Julian days/dates instead of calendar dates, or if the date and time are together in one cell as a datetime string. See below for further help with dates and times.
9. If you have more than one file of data to add, move on to your second file, adding it below the first. Do not leave gaps for periods of time where you have no weather data (*still to resolve)*.
10. If there are any QA codes in the columns, they need removing before R script is applied.
11. Replace any ‘NAN’ cells with blanks.
12. When the data is all compiled and saved, save that tab as a CSV with the same name as the excel file. There will be a warning saying CSVs cannot save multiple sheets. As long as you have the data tab selected, this is OK, as this is the tab that the script will work on.

N.B. The most recent Campbell’s datasets are in **reverse date order**. Make sure you **reorder** them before you coy the data over. When you do, remember to expand the selection so that the data is reordered as well.

**Extracting Dates and Times**

**N.B. a Julian Day is not the same as a Julian Date.**

If you have **two columns (one with the year, the other with the Julian day**):

1. In your Julian Date column enter **=TEXT(VALUE(RIGHT(C2,2))\*1000+D2,"00000")** where C2 is the year column and D2 is the Julian day column.
2. You now have a 5 digit Julian Date (the first two digits are the year, the latter 3 are the day of that year, from 1 – 365). You can use this to derive a calendar date, see step 2.

If you have **a Julian Date, but no calendar date**:

1. If you’ve used formulas to derive the Julian Date, first **copy the Julian Date column** and **paste-special** so you just have the text and not the formulas. At this point you can erase both the Year and the Julian Day column. In your Date column enter:

**=("1/1/"&(IF(LEFT(D2,2)\*1<20,2000,1900)+LEFT(D2,2)))+MOD(D2,1000)-1**

where D2 is the Julian Date.

1. This will give you a 5 digit number. Reformat the cell as a date (dd/mm/yyyy) and you should have a calendar date. **Check a few dates** to make sure it’s converted correctly, using [these tables as references](https://landweb.modaps.eosdis.nasa.gov/browse/calendar.html). Reformat it to **dd/mm/yyyy**, and remember to **copy the column and paste special** to get rid of the formulas and just leave the values.

If you have a **calendar date, but no Julian Date**:

1. To extract a Julian Date from a calendar date **=TEXT(C2,"yy")&TEXT((C2-DATEVALUE("1/1/"&TEXT(C2,"yy"))+1),"000")** where C2 is the calendar date.
2. **Check a few dates** to make sure it’s converted correctly, using [these tables as references](https://landweb.modaps.eosdis.nasa.gov/browse/calendar.html). Remember to **copy the column and paste special** to get rid of the formulas and just leave the values.

If you have **a datetime string, and need to extract the date and time** into separate columns:

1. Extract the date into your date column using **=INT(A1).** This will give you a 5 digit number. Reformat the cell as a date (dd/mm/yyyy) and you should have a calendar date.
2. Extract time into a separate column using **=TIME(HOUR(C2),MINUTE(C2),SECOND(C2))**
3. Make sure you **copy-paste special** the date and time columns so that only the values remain, and not the formulas. Then delete the column containing the datetime string.

If your **time values appear as 10.00 instead of 10:00**:

* 1. Excel doesn’t recognise a time written with a period instead of a colon as a time. Use find and replace on that column to replace . with :

N.B. If you are pasting these formulas into Excel and they don’t seem to be working, sometimes the “ mark doesn’t paste properly from Word to Excel. Simply delete and retype “ in the formula bar of Excel, and it should work.

**Running the QA Process**

1. The R scripts must be run in the following order in R Studio (NOTE: the lubridate library must be installed):
2. **Gross Error Checks R Code Jun 2020**
3. **Suspect Checks R Code Jun 2020**
4. **Tidying Up R Code Jun 2020**
5. Run the codes one at a time in R Studio. You need to **set the working folder** in each of the R scripts, once you’ve loaded them and before you run them. This should be the site folder on the N drive that the raw template is saved in.

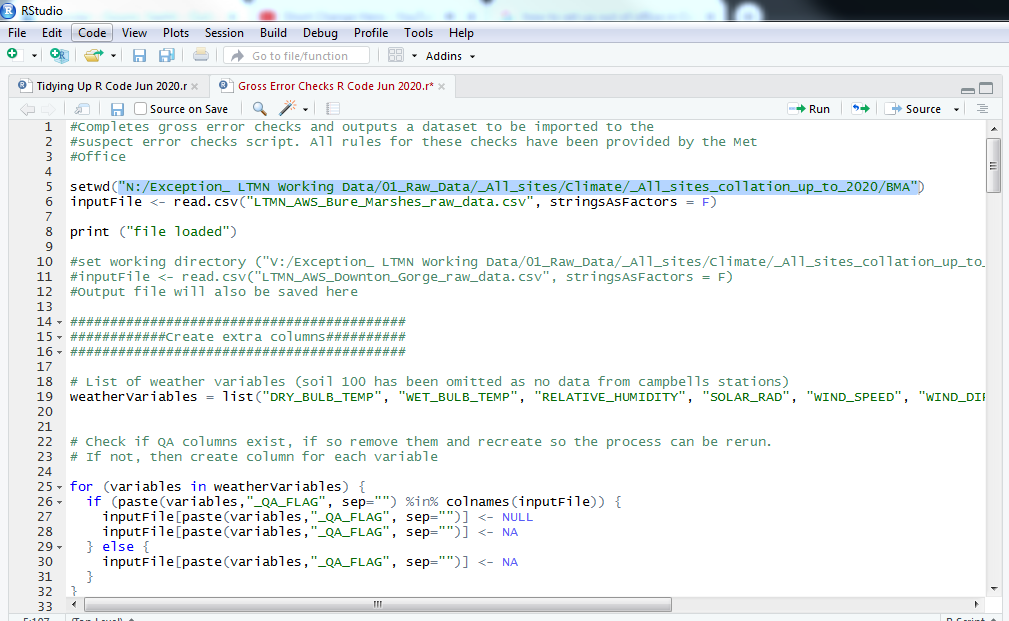


Figure 1 Set the working folder to the same place the raw data template is saved in.

1. You also need to **specify the names of the input and output files** in each of the scripts. Just change the filenames to whatever is appropriate. The **input for the Gross Error Checks is the csv file of raw data** that you saved in step 12. The **input for the Suspect Checks is the csv output from the Gross Error Checks**. The **input for the Tidying Up code is the csv output from the Suspect Checks**.

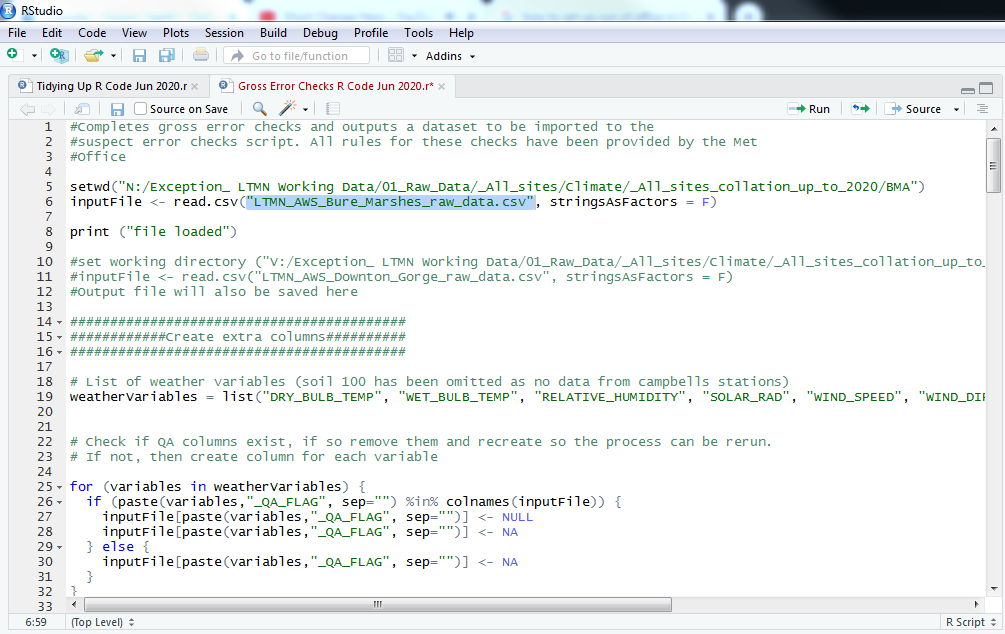


Figure 2 Set the name of the input file.

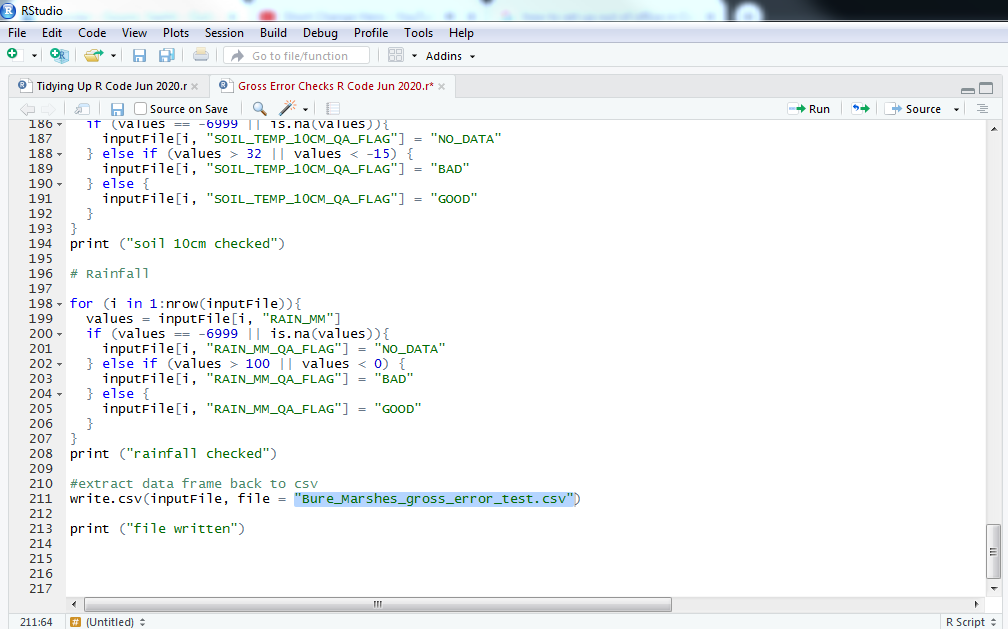


Figure 3 Set the output file name.

1. When running, check the console to ensure no red error messages occur, and to see progress. When you run the Suspect Code checks, you will get a red error message saying that on attaching the Lubridate package the date object is masked – this doesn’t seem to be a problem.



Figure 4 The console will tell you how the code is progressing and flag any errors.

1. Between each running of code, click the sweep/clear button in both the console and the environment, to make sure nothing is retained of the previous dataset or process.
2. It is also wise to **check the output** at the end of the three steps (the ‘…final\_error\_complete’ CSV), especially the QA code field, to ensure nothing has gone wrong. Look at a few of the QA codes that have been flagged, and double check the data does what the QA code is indicating. A list of QA codes are below:

|  |  |
| --- | --- |
| QA\_CODE | DESCRIPTION |
| QA101 | airTempStuckCode |
| QA102 | airTempStepCode |
| QA103 | airTempConsistencyCode |
| QA104 | wetBulbStuckCode |
| QA105 | wetBulbStepCode |
| QA106 | wetBulbConsistencyCode |
| QA107 | humidityStuckCode |
| QA108 | humidityStepCode |
| QA109 | radiationStuckCode |
| QA110 | radiationStepCode |
| QA111 | windSpeedStuckCode |
| QA112 | windSpeedStepCode |
| QA113 | windDirectionStuckCode |
| QA114 | windDirectionStepCode |
| QA117 | soil30StuckCode |
| QA118 | soil30StepCode |
| QA119 | soil10StepCode |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Range check** | **Step change** | **Stuck Check** | **Internal consistency** |
| **Dry Bulb** | -30C to +40C | change of more than 5 degrees/hour = SUSPECT | same value for 6 hours = SUSPECT | if wet bulb > dry bulb and dry bulb = SUSPECT, delete dry bulb |
| **Wet bulb** | -30C to +40C | change of more than 5 degrees/hour = SUSPECT | same value for 12 hours = SUSPECT | if wet bulb > dry bulb and wet bulb = SUSPECT, delete wet bulb and humidity |
| **RH** | 20% to 100% | change of more than 40%/hour = SUSPECT | same value for 24 hours = SUSPECT | if wet bulb > dry bulb and wet bulb = SUSPECT, delete wet bulb and humidity |
| **rainfall** | 0 to 100mm | None | None | None |
| **Wind direction** | 0 to 360 degrees | if wind speed >10 knots and direction changing >100 degrees per hour = SUSPECT | same value for 18 hours = SUSPECT (or 8 hours if wind speed > 0 knots) | None |
| **Wind speed** | 0 to 100 knots | change of more than 25 knots/hour = SUSPECT | same value for 12 hours = SUSPECT (or 3 hours if speed >= 15 knots) | None |
| **10cm soil temps** | 10CM SOIL TEMPERATURE BETWEEN -15 AND 32 | 10CM SOIL TEMPERATURE NOT CHANGING MORE THAN 6 DEGREES PER HOUR | None | None |
| **30cm soil temps** | -5C to +25C | change of more than 2 degrees/hour = SUSPECT | same value for 72 hours = SUSPECT | None |
| **Radiation** | -30 to 1500wm2 | if more than 5 in night time periods = SUSPECT (it isn't totally clear if this checks for a **value**of >5 at night, or a **change**of >5 at night - am clarifying) | same value for 24 hours = SUSPECT | None |

**Formatting the dataset**

1. You’ll notice that the QA process outputs a csv file that does not retain any of the formatting of the original spreadsheet. Therefore, **all formatting needs to be done after running the QA process**.
2. Open the **LTMN\_AWS\_QA\_data\_template\_2020** and save a new version named **LTMN\_AWS\_<site>\_QA\_data**.
3. **Make a copy of the csv final output** from the QA process (right-click the tab > move or copy > tick the “make a copy” box > choose the QA\_data file you’ve just created, and place it before sheet Quality Checking – outliers (*may want to remove this tab – it’s a bit useless)*)*.* The QA’d data will now be added as a new tab to the QA data file.
4. Change the new tab name to **AWS Data**.
5. Remove the first column on the new tab, so that the first column is SITE.
6. All QA codes will appear in the QA Code 1 column. Where there is more than one QA code for a row, move the additional codes into the QA Code 2 and QA Code 2 column.
7. Remove any -6999s left in the data and leave blank.
8. Change the date format to yyyy-mm-dd,
9. Format the new tab as per the formatting on the other tabs (arial font, size 8; no italics; bold only in column headers, which should also be centred; page zoomed to 100% and cell A1 selected).

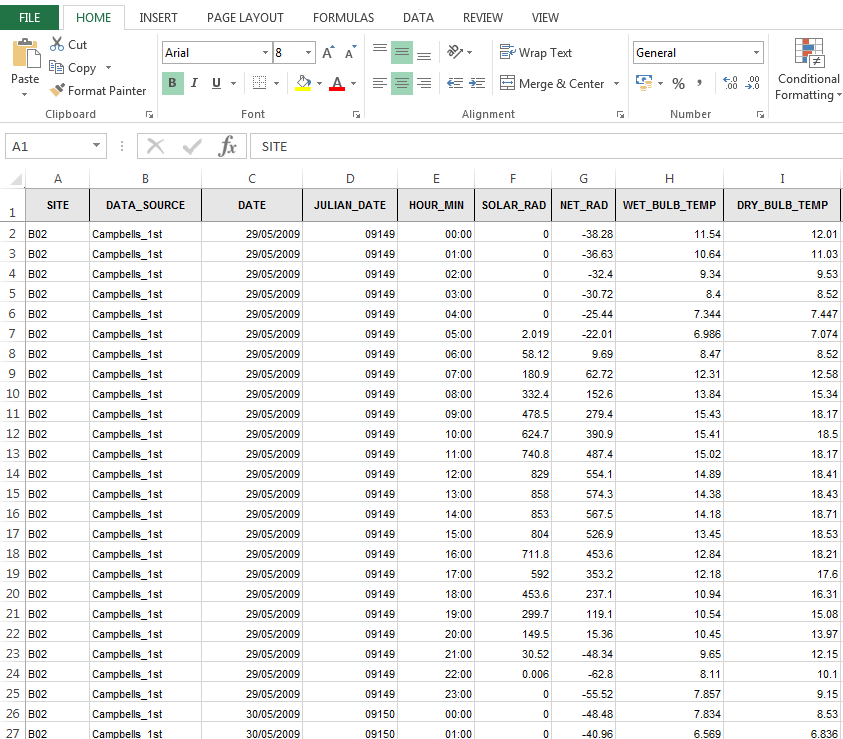
****

Figure 5 How the outputted AWS data tab should be formatted once it's in the QA data template.

1. The R scripts output a file in which leading 0s are removed. This affects some Julian dates, e.g. 2009-01-01 is 09001, which is changed to 9001. This can be rectified by selecting all the data in that column > right-click > format cells > Custom > Type 00000 and press OK.
2. *You may want to add a QA flag column and create a QA code for net radiation, if you calculate that. It could simply duplicate the solar radiation flag and code.*
3. When all your AWS data is compiled into one sheet for that site, **copy the date, dry bulb temperature and the rainfall column into the Graph Data tab**, and make a **pivot table** on that tab that summarises the data per day (your dates need to be proper dates, and not textstrings or the pivot table will not order them properly). Set the **Temp column to average** in Value Field Settings, and the **Rain column to sum**.
4. Select the pivot table columns and use **insert >recommended graphs** to make a **combined graph** of both temp (line) and rainfall (bar). **Copy and paste this also to the Home tab**.
5. *You may want to also create a rainfall stuck check QA code and manually identify periods when the rain gauge appears not to be working from the graph. Add the QA code to the data accordingly.*
6. **Fill out any other details** in the Home tab, and **delete any variables** from the Home tab list that are not recorded **at all** in that dataset.