**Solutions – Advanced**

Here’s the code you should have added:

advs <- advs\_temp %>%

# Calculate ABLFL

derive\_extreme\_flag(

by\_vars = vars(STUDYID, USUBJID, PARAMCD),

order = vars(ADT),

new\_var = ABLFL,

mode = "last",

filter = (!is.na(AVAL) & ADT <= TRTSDT)

) %>%

# Calculate BASE

derive\_var\_base(

by\_vars = vars(STUDYID, USUBJID, PARAMCD)

) %>%

# Calculate CHG

derive\_var\_chg() %>%

# Sort the data frame

arrange(USUBJID, PARAMCD, ADT)

1. **80.29 kg**
2. **-0.44 C**
3. Use the same code as before, but change the red extracts below in the derive\_extreme\_flag function:

advs2 <- advs\_temp %>%

…

order = vars(ADT, VSSEQ),

…

Use the following code to compare against the advs you created earlier:

library(diffdf)

# Compare 2 data frames advs & advs2 with the key variables USUBJID, PARAMCD, ADT

diffdf(advs, advs2, keys = c("USUBJID", "PARAMCD", "ADT"))

You should see printed to the console:

No issues were found!

This shows there was no impact of the change with the current data.

1. Insert the following code before your baseline derivations and you should get the answer **97.196 F** (or you may have rounded to 97.2). *Note: you might have done this using mutate, but an admiral function exists to help here – see derive\_summary\_records:* [*https://roche-gsk.github.io/admiral/reference/derive\_summary\_records.html*](https://roche-gsk.github.io/admiral/reference/derive_summary_records.html). *You could then add extra code as needed to populate any other blank variables such as ADY or AVALC for these newly created records.*

# Derive new parameter records

advs\_temp2 <- advs\_temp %>%

derive\_summary\_records(

by\_vars = vars(STUDYID, USUBJID, TRTSDT, VISITNUM, VISIT, VSDTC, ADT),

filter = PARAMCD=="TEMP",

analysis\_var = AVAL,

summary\_fun = function(x) ((x\*9/5) +32),

set\_values\_to = vars(PARAMCD="TEMPF", PARAM="Temperature (F)", PARAMN=7)

)