Specifications:

- You are allowed at most zero `DATA` steps and seven `PROC` steps to complete this assignment.

- Create a library that points to the provided data sets on your class' shared drive. Additionally, create a library named HW1 that points to the location where you will store any data sets you create. **Also save any non-SAS results you are required to produce (e.g., PDF) to the same directory as your HW1 data sets.**

  - Recall, any path outside of an `X` statement must be relative!
  - Lecture 2 gave you the name and paths for all the relevant information I provide to you and your assignments specifications provide any specific names you should use for that assignment.

- In this assignment, you'll start with the RTPtall data set from our shared library. You should recognize the data values from HW0 – a kind person has come along and rearranged the RaleighTempPrecip data set for you so that it has a more useful structure! (Guess what you'll be learning to do in a few weeks?!) Unfortunately, the kind person isn't perfect, so the data set is not correctly sorted. Nevertheless, you have all the skills necessary to produce a report that matches mine (*HW1 Duggins Weather Analysis.pdf*). As always, there are a few elements you'll need that cannot be inferred from the provided report or some critical information I want you to have easy access to, for those I've provided these pointers!

  - The PDF uses the Festival style.
  - In PROC MEANS, the keyword for standard deviation is STDDEV and for interquartile range the keyword is QRANGE. (For a full list of keywords, go to the SAS Documentation for the PROC MEANS statement here.)
  - My smaller titles and all footnotes use an 8 pt font. All other font sizes are the default and all the fonts and colors are defaults as well.
  - In the formats for temperature, I've used notation like [75,90) which would indicate that 75 *is* included in that range and that 90 *is not* included. Pay attention to that when building your formats.
  - You now know how to ensure your date, page numbers, and proc titles are each included/excluded as needed.
  - In the frequency analysis table, note that missing values are included in the table (which requires you modify the frequency analysis code to ensure that happens) and have a custom label (which requires you think about this when building your custom format - you want to make sure that missing values would be included with the appropriate label whether they appear as a row, a column, or both).
  - When producing output objects that require a subset of the data, I want you to practice creating subsets in a few different ways so that you are comfortable with different approaches:
    * For the numeric summary table, select year using a numeric variable and select month by using a character variable and specifying the values you *do* want.
    * For the weighted frequency analysis, select days of the week by using a character variable and specifying the values you *do not* want.
    * For the listing (printout of records), use only numeric variables when specifying your conditions.
  - For the output objects that require BY-group analysis, keep in mind that your BY statements do not have to match your PROC SORT, they just cannot *conflict*. We discussed this in class, but this is your chance to really think about what that means in a practical application. (Related question: why would we not just re-sort the data whenever we wanted to?!)
  - As discussed on the first day of class, we don't focus on analysis in this course because you have other classes for that. However, to simulate the inclusion of statistical results in your work – and to practice wonderful things like ODS! – you're being provided with some code to include in your program that will fit some statistical models and place part of the results in your report. While you are not required to understand the code, you *will* need to modify it using a concept that we *did* learn.
    * Insert the following into your program at the appropriate location to add the statistical model results to your report.

      ```
      ods select 'Type III Model ANOVA';
      proc glm data = HW1.<PUT-YOUR-DATA-SET-NAME-HERE>;
        <ADD-YOUR-STATEMENT-HERE>
        class DayC;
        model Prcp = tMax tMin DayC;
      run;
      ```

* In the PROC GLM statement, replace the pseudocode with your actual data set name.
* Inside the GLM step, replace the pseudocode statement with the an actual statement that will produce separate statistical models for each distinct 15-year group in the data set.
* Note that the provided ODS SELECT statement looks a little different than usual – it has quotes! You don't need to modify this at all and it will work fine if your GLM code is correct. However, to make sure you understand ODS I'd encourage you to figure out (a) how I found out what to put in those quotes and (b) why I chose to use this particular ODS SELECT statement instead of the syntax you've seen before – was it necessary?

• When creating the report, place the resulting output only in the PDF. No results should appear in any other destination - *regardless of who runs your code.*

• As our reports get longer because we can do more, it is important to develop a process you can use to help you validate your results.

  – Pay attention to titles & footnotes. They are not just there for you to copy and paste! Like in any data presentation, they give important context about what is **in** the presentation. (In fact, I strongly advise that you **never copy and paste text from rich/formatted to plain** (e.g., PDF or Powerpoint to SAS or VSC). Formatted text may include invisible characters used to support the formatting done by Adobe, Microsoft, etc.

  – For summary reports, like MEANS and FREQ output objects, it is easy enough to manually validate the tables if there are not many of them. In this case there's only a few tables, so manual comparisons are possible. (They just are not very fun. We'll learn how to automate this soon!)

  – For listing reports (the part you get from PROC PRINT), you certainly want to avoid reviewing it page by page!
    * Focus on key pieces of information. Personally, I start with the first and last records. Do they match?
    * What about the length of the listing? It should be roughly the same. (I say "roughly" because you are on the VTL and I am not, so there may be some small settings that we won't bother to learn how to control in this class that could change your margins, default fonts, etc.)
    * Next, think about key values. Examples include things like missing data values, very long (or short) strings, strings with punctuation, very large (or small) numeric values, etc. Really, any type of value that you think might trip up your code is a value you want to check in the printout. For this assignment, we have **sums** in this report, so if the sums match, it provides some evidence that the individual records match!
    * Finally, consider doing some random checks. Pick a few observations at random (or as close to random as you can manage) and make sure the output matches.

• Once your output file is completely generated, ensure the listing destination is reopened so that future code in the same SAS session does not generate unwanted messages in the log.