6 Adv Data Wrangling P1

12 Logical Vectors

1. What a Logical Vector Is
    - a logical vector contains only: TRUE, FALSE, or NA
    - you won't always see logical columns in raw data but you create them constantly in analysis
2. Comparisons Create Logicals
    - using comparison operators on numeric or character data returns logical vectors: <, <=, >, >=, ==, !=
    - is.na() to test for missing values
3. Boolean Algebra Combines Logicals
    - you can combine logical conditions
4. Summaries of Logical Vectors
    - logical vectors behave like 0/1 when summarized
    - any(x): is any value TRUE?
    - all(x): are all values TRUE?
    - sum(x): counts how many TRUEs
    - mean(x): proportion of TRUEs

13 Numbers

1. R Stores Numbers in Floating-Point Form
    - R does not store most numbers exactly.
    - some calculations produce tiny rounding errors
2. Never Check Numeric Equality Directly
    - use a tolerance instead
3. Decimal vs. Scientific Notation
    - R automatically prints very large or very small numbers in scientific notation
4. Integer vs. Double
    - most numbers in R are double-precision (floating-point).
    - ddd L to force an integer
5. Special Numeric Values
    - Inf = infinity
    - -Inf = negative infinity
    - NaN = "not a number"
    - NA = missing value
6. Summaries With Numbers Must Handle NAs
    - most numeric summaries require na.rm = TRUE:

16 Factors

1. Understanding Factors
    - factors represent categorical data
    - each factor has values and levels
    - levels define the allowed categories and their ordering
2. Why Level Order Matters
    - the order of levels affects how plots display categories
    - the level order influences how models interpret categories
    - reordering levels can clarify relationships in your data
3. Factors vs Characters
    - use characters when categories have no meaningful order
    - use factors when there is a meaningful category structure
    - convert characters to factors when needed
4. Creating And Adjusting Factors
    - use as.factor() to convert a variable into a factor
    - use fct_relevel() to manually reorder levels
    - use fct_reorder() to reorder based on another variable
    - use fct_lump() to combine rare categories
5. Using Factors In ggplot
    - ggplot displays categories in the order of factor levels
    - reordering factors before plotting improves readability
    - fct_reorder() is especially helpful for bar charts and boxplots

17 Dates and Times

1. Parsing Dates With Lubridate
    - use ymd(), mdy(), and dmy() to parse common date formats
    - use ymd_hms() or similar functions to parse date-times
    - the letter order tells lubridate how to interpret the string
2. Distinguishing Dates And Date-Times
    - dates store only year, month, and day
    - date-times store the date plus time and timezone
    - choose the correct type depending on your analysis
3. Extracting Date Components
    - use year(), month(), and day() to pull calendar components
    - use hour(), minute(), and second() for date-times
    - use wday() to extract day of week with optional labels
4. Modifying And Shifting Date Components
    - assign new values to components like year(x) <- 2025
    - shift dates using days(), weeks(), months(), or hours()
    - adding or subtracting time units adjusts the date or time accordingly
5. Working With Time Zones

- every date-time includes a timezone attribute
- use with_tz() to convert to a new timezone
- use force_tz() only when you need to change the label without converting

6. Rounding And Grouping Dates
    - use floor_date() to round down to the nearest unit
    - use ceiling_date() to round up
    - use round_date() when general rounding is appropriate
7. Understanding Durations, Periods, And Intervals
    - durations measure exact seconds
    - periods use human calendar units like months or years
    - intervals represent a span between two specific date-times
8. Handling Invalid Or Missing Dates
    - lubridate returns NA for impossible dates like “2020-02-30”
    - parsing functions simplify cleaning messy date data
    - checking for NA is important before analysis