Kate Messitte – Exam 2 Summary Sheet

0. Missing Data

• Types: MCAR, MAR, MNAR
• Detect: `is.na()`, `sum(is.na())`, `skim()`, `vis_miss()`
• Handle: `drop_na()`, `replace_na()`, mean/median/mode, predictive (regression)

1. Writing Functions

• Structure: `f <- function(arg1,arg2){return(out)}`
• Benefits: reuse, cleaner code, avoids repetition
• Defaults: `f(x,y=2){x+y}`

2. Base R

• apply-family: `apply()`, `lapply()`, `sapply()`, `tapply()`
• Base plots: `plot()`, `hist()`, `boxplot()`

3. Iteration

• `for(i in vec){}` • `while(cond){}`
• purrr: `map()`, `map_df()`, `map_dbl()` (efficient, consistent types)

4. APIs / JSON

• Workflow: `GET(url)` → `content("text")` → `fromJSON()`
• Flatten nested: `fromJSON(..., flatten=TRUE)`

5. Web Scraping

• `read_html(url)` → `html_element()` → `html_text()`
• Tools: `html_elements()`, `html_attr()`, CSS/XPath
• Ethics: check `robots.txt`, avoid rapid scraping

6. Databases

• Connect: `con <- dbConnect(duckdb(), "db")`
• Access: `tbl(con,"table")` • Execute: `collect()`
• Write: `dbWriteTable()` • List: `dbListTables()`

7. Visualization

• Principles: declutter, readable labels, correct scales, accessible colors (`viridis`)
• Mistakes: 3D charts, too many colors, chopped axes

8. Interactive Viz

• `plot_ly(type="scatter", mode="markers")`
• Convert: `ggplotly(p)`
• Shiny = UI + server + reactive components

9. Quality Control

• Reproducible code, comments, clear names, version control (GitHub)
• Validate: raw vs clean comparison + sample sizes + check assumptions

Key Packages

tidyverse • purrr • httr • jsonlite • rvest • DBI • duckdb • plotly • viridis • naniar • skimr