

# Session 5 Report: Best Practices in Computing

## Applied Quantitative Methods II

2026-02-18

### Contents

<b>Summary</b>	<b>1</b>
<b>Slides</b>	<b>2</b>
Section Overview . . . . .	2
Compilation Status . . . . .	2
Pedagogy Review Summary . . . . .	2
Visual Audit Summary . . . . .	2
Images Needed . . . . .	2
<b>Assignment</b>	<b>3</b>
Structure . . . . .	3
Datasets . . . . .	3
Checker Results . . . . .	3
Placeholders . . . . .	3
<b>Solutions</b>	<b>3</b>
Files Created . . . . .	3
Knitting Status . . . . .	3
<b>Quality Scores</b>	<b>4</b>
<b>Issues Requiring Attention</b>	<b>4</b>

### Summary

Artifact	Status	File Path
Slides (presentation)	PASS	slides/05_computing/computing.pdf
Slides (notes)	PASS	slides/05_computing/computing_notes.pdf
Assignment	PASS	assignments/assign5_computing.pdf
Solution Part 1	PASS	assignments/solution/assign5_1.pdf
Solution Part 2	PASS	assignments/solution/assign5_2.pdf

All artifacts compiled/knitted successfully.

# Slides

## Section Overview

The slide deck (`computing_body.tex`) contains **27 content frames** organized into **6 sections**:

Section	Slides	Key Content
Introduction	3	Why computing practices matter, motivating discussion prompt
Project Organization	5	Task folders, workflow_example tree, file naming, R/LaTeX integration, Makefiles
Writing Better Code	5	DRY principle, constants, checks/assertions, print diagnostics, code style
Plain Text and Tools	4	Why plain text, code editors, command line, discussion prompt
Version Control with Git	5	Git recap, .gitignore, commit habits, GitHub replication, collaboration
Wrap-up	4	Resources, summary, next week, questions

## Compilation Status

- **Presentation PDF:** 152 pages (including animation steps), compiles cleanly
- **Notes PDF:** 17 pages, compiles cleanly
- **Overfull hbox warnings > 10pt:** 0
- **LaTeX errors:** 0

## Pedagogy Review Summary

The pedagogy reviewer rated the deck positively (5/8 patterns fully followed, 2/8 partially applied, 1/8 not applicable). Key findings:

- **Strengths:** Good motivation-before-formalism structure, excellent worked examples after every concept, consistent semantic color usage (red for bad, teal for good), thorough presenter notes
- **Minor suggestions:** Could add one more discussion prompt in the Git section, could swap Makefile code/DAG slide order
- **Applied fix:** Added Makefile tab warning directly on-slide (was only in notes)

## Visual Audit Summary

The slide auditor identified 1 high-severity and 5 medium-severity issues:

- **Fixed:** Makefile slide reduced from 19 to 10 lines (switched to `\scriptsize`, removed blank lines)
- **Fixed:** Long URL on intro slide replaced with citation reference
- **Remaining:** No issues after fixes

## Images Needed

This deck contains no external images — all diagrams are built with TikZ. The instructor may want to add:

- A screenshot of a real GitHub replication repository

- An xkcd or similar comic about version control (the reference slides had `img/xkcd.png`)
- Screenshots showing good vs. bad project folder structures

## Assignment

### Structure

#### Assignment 5: Best Practices in Computing (7 pages, due March 12)

- **Part 1 (In-Class):** Project organization using the corruption dataset
  - Folder structure, analysis script, plots script, Makefile
- **Part 2 (Take-Home):** Code quality and Git practices
  - Rewrite a “bad” R script, practice Git commits, write reflection

### Datasets

- `corruption.dta` — 170 observations, 5 variables (`ccode`, `cname`, `ccodealp`, `ti_cpi`, `undp_gdp`)
- Available at: [github.com/franvillamil/AQM2/tree/master/datasets/other](https://github.com/franvillamil/AQM2/tree/master/datasets/other)
- URL verified: accessible

### Checker Results

- **URLs:** All accessible
- **Variable names:** All consistent with dataset columns
- **R code blocks:** All valid syntax, tested successfully
- **Packages:** `readstata13`, `modelsummary`, `ggplot2` — all available on CRAN
- **Compilation:** Clean (0 overfull hbox warnings, 0 errors)

### Placeholders

- **Deadline:** March 12, before class (set)
- No other placeholders remaining

## Solutions

### Files Created

File	Description	Status
<code>assign5_1.Rmd</code>	Part 1 solution (project organization)	Knits to PDF
<code>assign5_1.R</code>	Bare R script for Part 1	Created
<code>assign5_2.Rmd</code>	Part 2 solution (code quality, Git)	Knits to PDF
<code>assign5_2.R</code>	Bare R script for Part 2 (improved script)	Created
<code>code/assign5.R</code>	Combined script	Created

### Knitting Status

- `assign5_1.Rmd`: Knits successfully (warnings about unknown `markdown` and `makefile` engines are harmless)
- `assign5_2.Rmd`: Knits successfully (all code chunks are `eval=FALSE` since they reference hypothetical data)

# Quality Scores

Artifact	Score	Status
Slides	95/100	Acceptable
Assignment	100/100	Acceptable
Solution Part 1	95/100	Acceptable (minor engine warnings)
Solution Part 2	100/100	Acceptable

## Issues Requiring Attention

1. **Images:** The slide deck has no external images. Consider adding 1-2 images for visual variety (see suggestions above).
2. **Slide auditor note:** The “Key resources” slide (slide 25) is vertically dense with 5 resources. May need adjustment if it overflows on the specific projector setup.
3. **Assignment note:** The corruption dataset has no missing values on `ti_cpi` or `undp_gdp`, so the “drop missing” step removes 0 rows. Students may find this confusing. Consider mentioning this in class or in the solution.
4. **Solution note:** The `assign5_1.Rmd` solution downloads the corruption data from GitHub (URL) rather than loading it from a local path, since the solution runs independently of the assignment folder structure.