



Coding Styles and Linters: Can we learn from Software Development

Alex Brink, 22Aug2024

Executive Director, Biostatistics & Statistical Programming

parexel[®]

Topics

- › What are Coding Styles and Linters
- › What are the Advantages & Disadvantages
- › Why are Linters not really used in SP
- › Do we have an opportunity now
- › What would it look like in practice
- › Conclusion

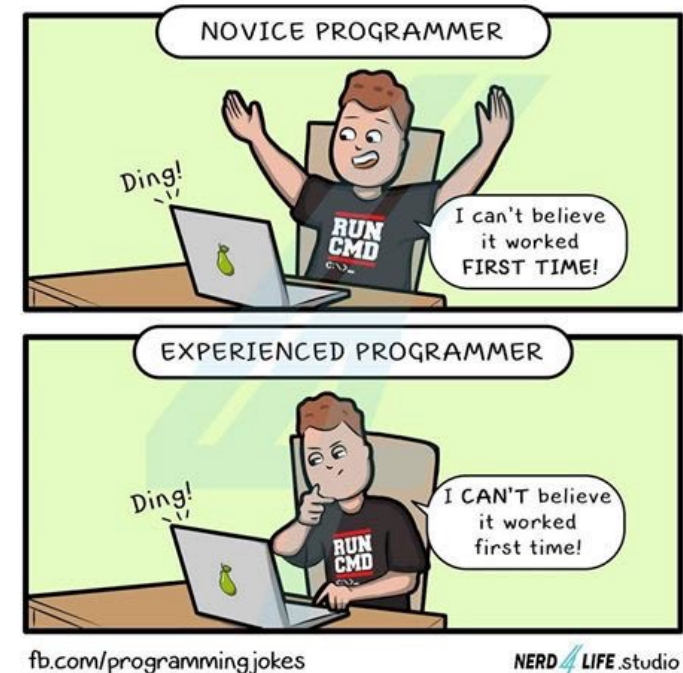


I HATE PROGRAMMING
I HATE PROGRAMMING
I HATE PROGRAMMING
IT WORKS!
I LOVE PROGRAMMING

shutterstock.com · 2235629943

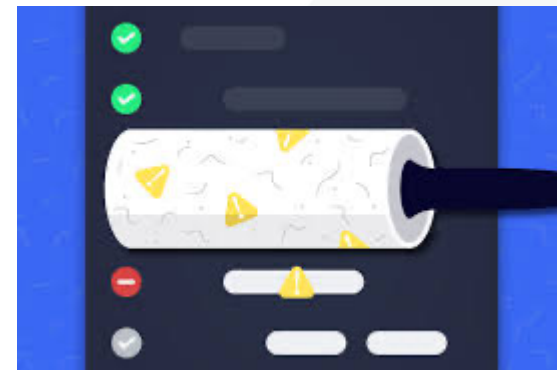
What are Coding Styles

- › Sets of guidelines and conventions for writing code
- › Define how code should be formatted, structured, and organized
- › Improve readability, maintainability, quality and consistency
- › Helps teams collaborate more effectively
- › Easier to understand and maintain each other's code.



What are Linters

- › Tools that analyze source code to flag errors & bugs
- › Improve code quality and enforce coding styles and standards
- › Can be integrated into development environments, build processes, or version control workflows
- › Helps developers catch and fix issues early in the development process
- › Leads to more robust and maintainable code



What are the Advantages & Disadvantages

Early Error Detection

Code Consistency

Best Practices

Improved Code Quality

Cost & Time-Saving

Customizable

Integration

Learning Curve

Overhead

Rigidity

Maintenance

Tool Dependence

Performance Impact

Language Limitations

Why are Linters not really used in SP

Historical practices

- Statistical programming has traditionally relied on other methods of code review and quality control.

Language specifics

- Some statistical programming languages, particularly SAS, may not have robust linter tools available

Focus on results

- In statistical programming, there's often a strong focus on the correctness of results rather than code style, which is what linters typically address.

Complex analyses

- Statistical code often involves complex analyses that may be difficult for automated tools to effectively evaluate.

Regulatory environment

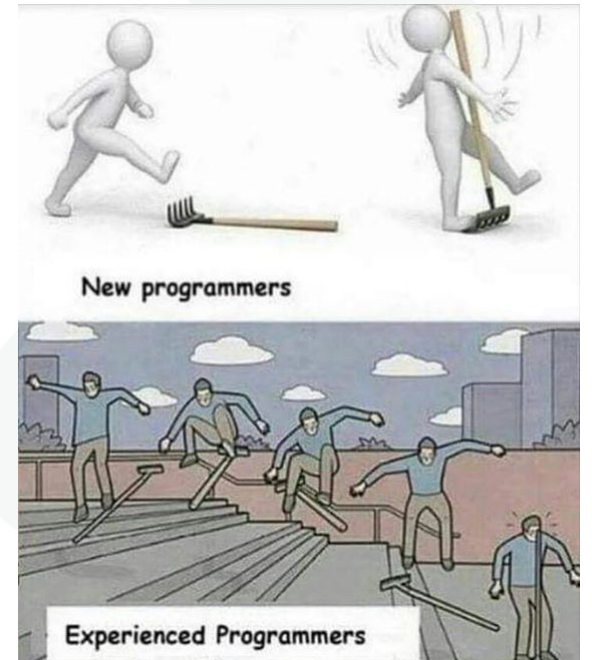
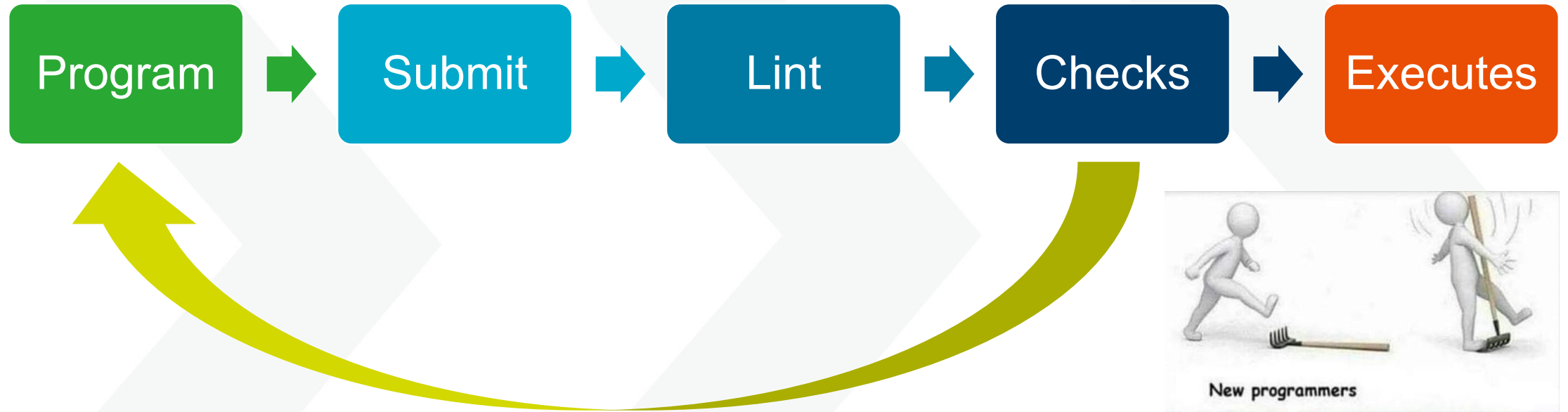
- In regulated environments (like clinical trials), there are often specific validation processes that may not include linting.

Do we have an opportunity now

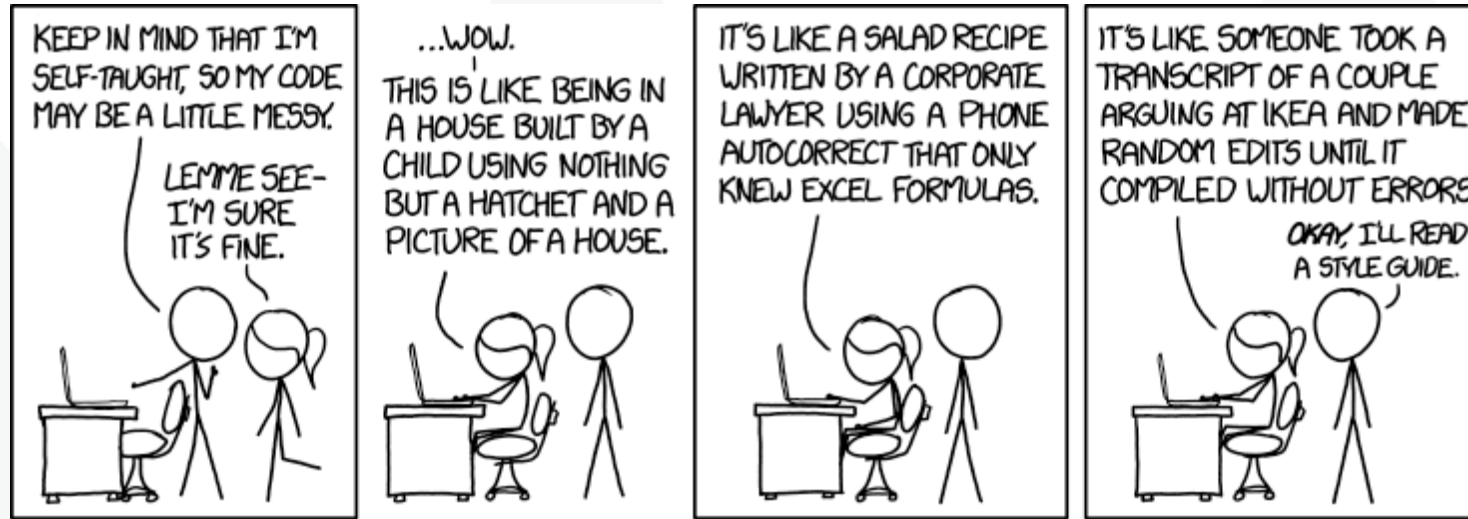
- Many Pharma/CROs moving towards R
- R has great linters available
 - Lint
 - Goodpractice
 - Styler
 - RStudio's built-in diagnostics
 - tidyverse style guide
- Note: SAS Lint is an open-source project available on GitHub
 - Command-line tool that can analyzes SAS code for various issues and style violations



What would it look like in practice



Conclusion



- › Adoption of rigid coding styles and linters can bring significant benefits to our field
- › Consistent coding styles enhance readability and maintainability
- › Linters serve as an automated first line of defense against common coding errors
- › Implementing these tools can reduce technical debt over time



*With Heart*TM