

CppStandards_Part3.pdf — Coding Style

Key Themes:

- Prefer compile-time over run-time checks
- Use **const** aggressively to protect immutability
- Avoid macros, magic numbers, and long functions
- Initialize variables early and keep scopes tight
- Minimize inter-file dependencies
- Write self-contained header files with proper **#include** guards

Relevance to *ClinicTrendsAI*:

Concept	ClinicTrendsAI Application
const usage	In your ML pipeline and Streamlit interface, mark things like <code>MODEL_NAME = ...</code> as constants for safety.
Short, flat functions	Your refactoring of <code>ModelsPage.py</code> and <code>MLExperimentsPage.py</code> shows great use of breaking views into small UI blocks.
Avoid magic numbers	Define values like top-N topics, thresholds, or score bins as named constants (e.g. <code>TOP_K = 5</code>).
Scoped variables	Only define DataFrames and temp vars right before their use — you're already doing this well in your model pipelines.
Always initialize	Ensure variables like <code>score_column</code> or intermediate results in <code>train_tfidf_model</code> are defined with defaults.
Self-sufficient components	Treat each Streamlit page (<code>views/*.py</code>) as self-contained — avoid circular dependencies in function calls.