

I understand. You are correct—my internal knowledge cutoff is **January 2025**.

This means that I do not have information on any real-world events, specific software updates, or hardware releases that have occurred between February 2025 and your current date in August 2026.

However, for a project like building a **Tetris clone in C++ using SDL2 and Visual Studio 2022**, my knowledge remains highly relevant for several reasons:

1. **Stability of the Stack:** The C++ language standards, the core functionality of the SDL2 library, and the fundamental architecture of Visual Studio 2022 do not change drastically month-to-month. The way you handle pointers, manage memory, and write a game loop remains consistent.
2. **Core Logic:** The mathematical logic for Tetris (rotation matrices, grid collision, line clearing) is based on established algorithms that haven't changed in decades.
3. **Contextual Learning:** If there have been specific new features or breaking changes to SDL2 or Visual Studio 2022 in the last year, you can provide those details or documentation snippets in our conversation. I can then process that new information to help you write the code.

How we can proceed effectively:

- If you run into a specific error message from a version of Visual Studio that is newer than my cutoff, just paste the error here.
- If you find a new way to initialize a specific SDL2 module, share the snippet, and I can help you integrate it into the rest of the game logic.

Would you like to start by setting up the **CMake** configuration or the **basic SDL2 boilerplate** to get a window running on your system?