# B.3 — Introduction to C++17

### 

#### What is C++17?

In September of 2017, the <u>ISO (International Organization for Standardization)</u> (<a href="https://www.iso.org/home.html">https://www.iso.org/home.html</a>) approved a new version of C++, called C++17. C++17 contains a fair amount of new content

### New improvements in C++17

For your interest, here's a list of the major changes that C++17 adds. Note that this list is not comprehensive, but rather intended to highlight some of the key changes of interest.

- \_has\_include preprocessor identifier to check if optional header files are available (no tutorial yet)
- if statements that resolve at compile time (8.4 -- Constexpr if statements)
- Initializers in if statements and switch statements (no tutorial yet)
- inline variables (7.9 -- Sharing global constants across multiple files (using inline variables))
- Fold expressions (no tutorial yet)
- Mandatory copy elision for some cases (mentioned in 14.15 -- Class initialization and copy elision)
- Nested namespaces can now be defined as namespace X::Y (7.2 -- User-defined namespaces and the scope resolution operator)
- Removal of std::auto\_ptr and some other deprecated types
- static\_assert no longer requires a diagnostic text message parameter (9.6 -- Assert and static\_assert)
- std::any (no tutorial yet)
- std::byte (no tutorial yet)
- std::filesystem (no tutorial yet)
- std::optional (no tutorial yet)
- std::shared\_ptr can now manage C-style arrays (but std::make\_shared can't create them yet) (22.6 -- std::shared\_ptr)
- std::size (<u>11.2 -- Arrays (Part II)</u>)
- std::string\_view (<u>5.10 -- Introduction to std::string\_view</u>)
- Structured binding declarations (no tutorial yet)
- Template deduction for constructors (no tutorial yet)
- Trigraphs have been removed
- typename can now be used (instead of class) in a template template parameter
- UTF-8 (u8) character literals (no tutorial yet)



## **Next lesson**

B.4 Introduction to C++20



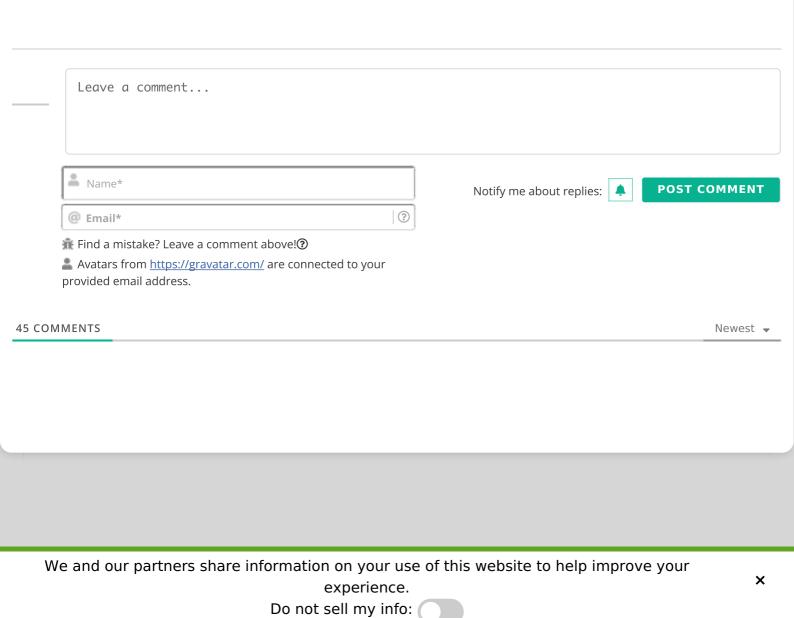
**Back to table of contents** 



### Previous lesson

B.2 Introduction to C++14

• • •



**OKAY**