

Jon Kalb Presents

Programming in Modern C++: Some Important C++20 Features



Programming in Modern C++: Some Important C++20

Features covers language and library features which the standard introduced with C++20. The material covers the mechanics of the features, but, more importantly, covers how to these features are part of producing high quality C++ code.

Covered major features include Concepts, Attributes, operator `<=>`, Improvements to Compile-Time Programming, Variants, Modules, Lambda Expression Augmentations, Ranges, and Coroutines.

The material for this course was developed jointly by **Steve Dewhurst**, **Jon Kalb**, and **Dan Saks**.

Course Highlights

Participants will gain:

- An understanding of concepts including concept/requires syntax.
- An understanding of attributes and introduction to standard attributes.
- An understanding of the “spaceship” operator (`<=>`) and defaulted operators
- An understanding of `constexpr` improvements and `constexpr` and `constexpr`.
- An understanding of modules.
- An understanding of C++20 enhancements to lambda expressions.
- An understanding of ranges
- An understanding of coroutines

Who Should Attend

Designers and developers who are using, considering using, or wish to know about the expanded capabilities of C++20. Attendees should be experienced with C++ and comfortable with its primary features (e.g., classes, templates, inheritance, STL, etc.).

Format

Lecture, question/answer. Exercises extend class time by one day.

Length

Two full days (six to seven lecture hours).



Topic Outline



- Concepts
- Attributes
- operator <==>
- Improvements to Compile-Time Programming
- Variant Visitors,
- Modules,
- Lambda Expression Augmentations
- Ranges
- Coroutines

For more information on this course, [contact Jon Kalb](mailto:jon@cpp.training) at jon@cpp.training.

