

Modern C++ Development with C++ 11

Course Summary Table

Duration:	32 hours instructor led course
Target Audience:	Windows platform C++ developers
Objectives:	The participant will be able to maximize development productivity by leveraging the new C++ language & library enhancements
Pre Requisites:	At least 2 years of experience in C++ development on Windows

Abstract

C++ 11 is a new C++ standard that enhanced the productivity of C++ while preserving the important properties of the language such as performance & efficiency. Visual Studio 2010 and Visual Studio 2012 cover many of the new features of the language and of the standard library.

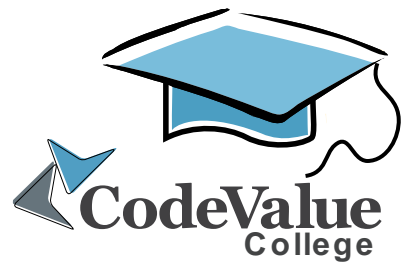
In this course we will see the new C++ 11 capabilities, and after participating in it your C++ skills will be renewed and your C++ code will never quite look the same.

Syllabus

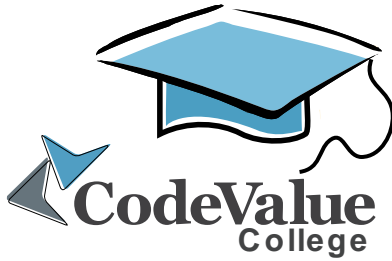
- ✔ Module 1: Introduction
 - ✔ The C++ 2003, C++ TR1 & C++ 11 standards
- ✔ Module 2: C++ 11 Language Tour
 - ✔ Declaring types
 - ✔ The *auto* keyword
 - ✔ The *decltype* keyword
 - ✔ RValue reference
 - ✔ Overview
 - ✔ Move semantics
 - ✔ Perfect forwarding
 - ✔ Improved STL performance
 - ✔ `nullptr`
 - ✔ Static assertion



- ✓ Range-based for loops
- ✓ Scoped enumerations support
- ✓ Lambda Function
- ✓ Module 3: Microsoft Specific Implementation
 - ✓ SCARY iterators
 - ✓ Auto-vectorizer
 - ✓ Auto-parallelizer
 - ✓ C++ Accelerated Massive Parallelism (AMP) Overview
- ✓ Module 4: The Standard Library – General Concepts Refreshment
 - ✓ The *std* namespace
 - ✓ STL header files
 - ✓ Lap around STL abilities
 - ✓ Containers
 - ✓ Iterators
 - ✓ Algorithm
 - ✓ Functions
 - ✓ Memory
 - ✓ Allocators
 - ✓ Working with strings
 - ✓ iostream programming
 - ✓ Output streams, input streams, input/output streams
 - ✓ stdout, stdin, stderr (cin, cout, cerr, clog, wcin, wcout, wcerr, wclog)
 - ✓ I/O manipulation
 - ✓ File manipulation
 - ✓ String streams
- ✓ Module 5: New STL Improvements
 - ✓ Memory handling & Shared Pointers
 - ✓ shared_ptr, weak_ptr and make_shared
 - ✓ New Containers
 - ✓ Array
 - ✓ Unordered set & map
 - ✓ Tuple
 - ✓ Functional programming
 - ✓ Bind, mem_fn & Lambda
 - ✓ New algorithm
 - ✓ find_if_not, copy_if, is_sorted
 - ✓ Const iterator
 - ✓ cbegin, cend, crbegin, crend
 - ✓ Transport exception between threads
 - ✓ Regular Expressions
 - ✓ Random
 - ✓ Type traits
 - ✓ Thread Support
 - ✓ thread, future, atomic, mutex, condition variables



- ▼ chrono
- ▼ filesystem
- ▼ ratio
- ▼ Module 6: Visual Studio Enhancement
 - ▼ Static Code Analysis
 - ▼ Unit Test Framework
 - ▼ Code Coverage
 - ▼ New Debugging capabilities



Course Compatibility Questionnaire

Please answer the following questions as accurately as possible:

Name: _____ Email: _____
 Company: _____ Phone: _____

Language / Technology / Platform	Years of Experience						Level of Familiarity				
C	0-1	1-2	2-3	3-4	4-5	5+	1	2	3	4	5
C++	0-1	1-2	2-3	3-4	4-5	5+	1	2	3	4	5
C#	0-1	1-2	2-3	3-4	4-5	5+	1	2	3	4	5
STL	0-1	1-2	2-3	3-4	4-5	5+	1	2	3	4	5
MFC	0-1	1-2	2-3	3-4	4-5	5+	1	2	3	4	5
ATL	0-1	1-2	2-3	3-4	4-5	5+	1	2	3	4	5
TR1	0-1	1-2	2-3	3-4	4-5	5+	1	2	3	4	5
Other _____	0-1	1-2	2-3	3-4	4-5	5+	1	2	3	4	5

What is the Visual Studio version that you use the most?

What are your expectations from the course?

Thanks!

<http://college.codevalue.net/>