

Comprehensive Guide to Learning C++ from Basics to Advanced



C++ is one of the oldest programming languages which are still in use today. It has one of the largest existing codebases and therefore cannot be thrown out even though various new programming languages have come around. C++ is extensively used in video games, finance, high performance real-time and embedded systems, manufacture, transportation and various other industries where precision and determinism are critical. Moreover there are very few alternatives. For real-time situations as well as for high performance desktop applications, C++ is extremely useful. Scope-Bound Resource Management (SBRM) is one of the killer features of C++. It is also the only programming language that is built around the concept of SBRM. The life times of all the objects are precisely known and C++ programs which

are well written can guarantee the acquisition and release of resources in a fully deterministic manner.

Applications with better overall performance can be built using C++ compared to most other programming languages. Abstractions are built in C++ with compiler-only items such as templates moving the computation from runtime to compile time which makes the overall application faster. Most embedded systems do not have the power to run various programming languages that require more runtime support compared to that required by C++ which require virtually no runtime support. Programming languages such as C-styled languages, Java and JavaScript can be easily understood by anyone who knows C++. Finally, writing server side programs, programs for Android applications and others that are efficient and secure can be easily done using C++.

Here is a comprehensive list of C++ tutorials, tips, tools that can help programmers learn C++ from scratch and become proficient in it:

C++ Resources:

1. 10 Tips for C++ Performance Improvement Code Optimization
2. Top Ten Tips for Correct C++ Coding
3. C++ Tips and Tricks
4. The Top 20 C++ Tips of All Time
5. Top Productivity Tips for Using Eclipse for Embedded C/C++ Developers
6. Tips and Tricks for Code Golfing in C++
7. Tips for C++ Development in Xcode
8. Ten tips on improving your programming skills
9. The Best Questions for Would-be C++ Programmers
10. Ten C++ Features Every C++ Developer Should Use
11. Experiences Converting a C++ Communication Software Framework to Java
12. An Introduction to Design Patterns in C++ with Qt 4
13. An Introduction to XML Data Binding in C++
14. C and C++: a Case for Compatibility

- 15.5 Reasons to Use C++ for Android Development
16. Why C++ is the perfect choice for modern app development
17. C vs. C++
18. Top 10 advantages of C++ over C.
19. Benefits with C++ in modern apps
20. C++ Best Practices

C++ Tutorials:

1. Beginner's Guide to C++
2. How to Write a Simple C++ Program
3. Creating Games in C++ – A Step-by-Step Guide
4. C++ Programming Tutorial Instructions for Practical Sessions
5. C++ Programming Tutorials for Beginners
6. How to Design your First C++ Programs
7. Debugging your C++ Program
8. Converting between binary and decimal in C++
9. Basic Inheritance in C++
10. Guide into OpenMP: Easy multi threading programming for C++

C++ Cheat Sheets:

1. C++ Reference Card
2. STL Quick Reference Card
3. C++ and Java Syntax Differences Cheat Sheet
4. C++ Containers Cheat Sheet
5. Java cheat sheet for C++ programmers

Video Resources:

1. C++ Tutorial For Absolute Beginners
2. Understanding a Simple C++ Program
3. Creating Functions That Use Parameters in C++

4. Using Variables in Classes in C++
5. How to make a basic shopping list program in C++
6. Converting MATLAB Code to Readable C++
7. How To Compile Your First C++ Program in Visual Studio
8. Create a Basic Graphics Program in C++
9. How to Create a Window in C++
10. How to Write C/C++ program on Ubuntu Linux terminal

Hope these resources will be helpful in giving you a complete gist of C++ programming and help you build remarkable C++ applications.