

# Common Syntax Errors

IT IS CRITICAL to understand the fundamental characteristics of syntax errors and to know how to interpret warnings. Beginning programmers can shorten their learning curve by studying the most commonly made syntax errors and then by avoiding such errors.



## Objective:



Compare and contrast syntax warnings and errors, and list 10 common syntax errors of beginning C++ programmers.

## Key Terms:



syntax  
syntax error  
syntax warning

## Fundamental Characteristics of Syntax and Semantic Errors

It is very common for beginning programmers to make syntax errors or to receive syntax warnings. But if you learn to identify the common mistakes, you will be able to correct them quickly and soon begin to make fewer errors or receive fewer warnings.

## SYNTAX WARNINGS AND SYNTAX ERRORS

A **syntax warning** is a tool that indicates that code structure is questionable. Although no violation of syntax rules are indicated by a warning, if the code structure is questionable, warnings should not be ignored. Warnings are detected when the program is compiled. Yet warnings do not prevent final executable code from being created by the compiler, so the program can still be run, even if syntax warnings exist. Warnings may cause unexpected results when a program is run.

**Syntax** is the rules that a programmer must follow when writing source code for a program. A **syntax error** is a violation of usage rules of the programming language being used. Syntax errors are detected when the program is compiled. Generally, they prevent final executable code from being created by the compiler, and the program cannot be run until all syntax errors are resolved.

## TOP 10 COMMON SYNTAX ERRORS

The top 10 most common syntax errors should be studied and put to memory.

1. Missing semicolon—Semicolons are used to indicate the end of a C++ program statement and are often forgotten. This may be the most common syntax error made by beginning programmers.
2. Misplaced semicolon—Another common beginner error is to place a semicolon after every line in the code. Appropriate semicolon placement should be checked for looping structures. Semicolons should not be placed following function headers, “If statement” test expressions, and preprocessor directives.
3. Mismatched parenthesis—Every opening parenthesis must have an appropriately placed closing parenthesis.
4. Mismatched curly braces—Every opening curly brace must have an appropriately placed closing curly brace.
5. Undeclared variable—A variable must be declared in the program before it can be used. A global variable can be used anywhere in the program. A local variable can only be accessed within the function in which it is declared.
6. Missing insertion operators—Insertion operators tell the compiler where data is to be directed. Without insertion operators, the compiler cannot send data to output consoles (e.g., a monitor) or store data in variables.



## FURTHER EXPLORATION...

### ONLINE CONNECTION: Common Syntax Errors and Warnings

One of the best ways to really learn new material is to be challenged to teach the material to someone else. Begin by further researching the topic of common syntax errors and warnings in C++ by reviewing the information at <http://fd.valenciacollege.edu/file/grhodes4/CommonBeginnerMistakes1.pdf>. Read and study the information on the site. Then create a poster presentation with supporting handouts for the class to facilitate their learning. You will be graded on your presentation content, thoroughness, clarity, and memorable suggestions for the learners. It would be best to create handouts, flash cards, etc.

7. Incorrect insertion operators—Insertion operators must point in the correct direction. Double less than signs point data out to an output device, such as the monitor. For example, `cout << "Hello";` sends the word Hello to the default output device. Double greater than signs take data input from the keyboard or another input location and store it in a named variable. For instance, `cin >> choice;` takes an input data value and sends it to be stored in a variable called “choice.”
8. Missing quotation marks around literals—Quotation marks indicate that a string of characters are to be taken literally and do not represent a variable name or a C++ keyword.
9. Mismatched quotation marks around literals—Every opening quotation mark must have an appropriately placed closing quotation mark to indicate the beginning and ending points of a literal.
10. Undeclared function—Just as with variables, functions must be declared before they can be called upon.



**FIGURE 1.** When taking a test or quiz for C++, remember the top 10 most common errors.

## Summary:



It is very common for beginning programmers to make syntax errors or to receive syntax warnings. But if you learn to identify the common mistakes, you will be able to correct them quickly and soon begin to avoid making such errors and receiving warnings. A syntax warning is a tool that indicates that code structure is questionable. Although no violation of syntax rules are indicated by a warning, if the code structure is questionable, warnings should not be ignored. Warnings are detected when the program is compiled. Warnings do not prevent final executable code from being created by the compiler, so the program can still be run, even if syntax warnings exist. Warnings may cause unexpected results when a program is run. A syntax error is a violation of usage rules of the programming language being used.

Syntax errors are detected when the program is compiled. Syntax errors prevent final executable code from being created by the compiler, and the program cannot be run until all syntax errors are resolved.

## Checking Your Knowledge:

---



1. Define and explain the term “syntax warning.”
2. Give an example of a syntax warning.
3. Define and explain the term “syntax error.”
4. Give an example of a syntax error.
5. What happens when you have an undeclared function?

## Expanding Your Knowledge:

---



Shrink your poster presentation onto a standard index card, using only the front side. Be as neat as you can in your organization and design. Use it during your next quiz as a study and quiz aid.

## Web Links:

---



### Common Syntax and Semantic Errors

[http://space.wccnet.edu/~pmillis/cps120/cps120\\_pgm\\_syntax.pdf](http://space.wccnet.edu/~pmillis/cps120/cps120_pgm_syntax.pdf)

### Syntax and Semantic

<http://www.scribd.com/doc/6742607/Syntax-and-Semantics>

### Syntax Errors

[http://cplusplus.syntaxerrors.info/index.php?title=Main\\_Page](http://cplusplus.syntaxerrors.info/index.php?title=Main_Page)

### Types of Syntax Errors

[http://www.ehow.com/list\\_6568693\\_types-syntax-errors.html](http://www.ehow.com/list_6568693_types-syntax-errors.html)