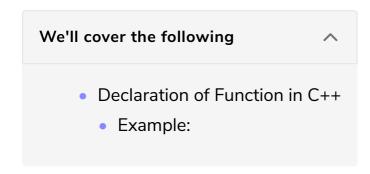
Function Declarations

This lesson explains how to declare functions in C++



Declaration of Function in C++#

Before we discuss *calling* functions, we must talk about *declarations*. Programs written in C++ are executed in logical order from the top down. A statement can only use *symbols*, or named constructs like *functions*, that are declared in code that has already been executed. Simply put; the compiler must be aware of the name used to describe a function before that function can be called in the code.

For example, to allow main to call functions that are defined after the main function itself in the code, we use **forward declarations**. A *forward* declaration tells the compiler that a function exists and what its *arguments* look like, but the *definition* of the function will be elsewhere.

The *declaration* begins much the same as the *definition*.

Example:

```
// Definition
int fctn2(int num1, int num2)
{ ... }

// Declaration
int fctn2(int, int);
```

Important Note: that the *argument* names are **optional** since the *declaration* does not need to use the arguments in any way. However, keeping the *arguments* needed in the *declaration* may be useful for documentation purposes.

Now that we've learned how to declare functions let's learn how to <i>call</i> functions in C++.	