

4.2 — Void

 learncpp.com/cpp-tutorial/void/

Void is the easiest of the data types to explain. Basically, **void** means “no type”!

Void is our first example of an incomplete type. An **incomplete type** is a type that has been declared but not yet defined. The compiler knows about the existence of such types, but does not have enough information to determine how much memory to allocate for objects of that type. **void** is intentionally incomplete since it represents the lack of a type, and thus cannot be defined.

Incomplete types can not be instantiated:

```
void value; // won't work, variables can't be defined with incomplete type void
```

Void is typically used in several different contexts.

Functions that do not return a value

Most commonly, *void* is used to indicate that a function does not return a value:

```
void writeValue(int x) // void here means no return value
{
    std::cout << "The value of x is: " << x << '\n';
    // no return statement, because this function doesn't return a value
}
```

If you use a return statement to try to return a value in such a function, a compile error will result:

```
void noReturn(int x) // void here means no return value
{
    std::cout << "The value of x is: " << x << '\n';

    return 5; // error
}
```

On Visual Studio 2017, this produced the error:

```
error C2562: 'noReturn': 'void' function returning a value
```

Deprecated: Functions that do not take parameters

In C, void is used as a way to indicate that a function does not take any parameters:

```
int getValue(void) // void here means no parameters
{
    int x{};
    std::cin >> x;

    return x;
}
```

Although this will compile in C++ (for backwards compatibility reasons), this use of keyword *void* is considered deprecated in C++. The following code is equivalent, and preferred in C++:

```
int getValue() // empty function parameters is an implicit void
{
    int x{};
    std::cin >> x;

    return x;
}
```

Best practice

Use an empty parameter list instead of *void* to indicate that a function has no parameters.

Other uses of void

The void keyword has a third (more advanced) use in C++ that we cover in section [19.5 -- Void pointers](#). Since we haven't covered what a pointer is yet, you don't need to worry about this case for now.

Let's move on!