Pointers

A pointer is variable that contains the address of another variable. In languages like Java, C# and Python, pointers are **hidden** from the programmer, and used only by the runtime system. In C++, understanding pointers is necessary for understanding how C++ programs work.

An expression that refers to an object in memory is an ***lvalue***. Variables are ***lvalues*** because you can store data in them. A named constant is a **non-modifiable *lvalue***. Many values in C++ **are not *lvalues***; the result of an expression **is a temporary value**, but it **is not** an ***lvalue***, because you cannot assign a new value it.

The following properties apply to modifiable ***lvalues*** in C++:

- Every ***lvalue*** is stored somewhere in memory; thus it **has an address**.
- **The address of an *lvalue* never changes**, even though the contents of those memory locations may change.
- The address of an ***lvalue*** is a **pointer or address value**, which can be stored in memory and manipulated as data.

To store an address value in memory, you create a **pointer variable**. Thus, a pointer is simply a variable that stores the address of some object in memory.

This course content is offered under a CC Attribution Non-Commercial license. Content in this course can be considered under this license unless otherwise noted.