# C++ Pointers

## Creating Pointers

You learned from the previous chapter, that we can get the memory address of a variable by using the & operator:

### Example

string food = "Pizza"; // A food variable of type string  
  
cout << food; // Outputs the value of food (Pizza)  
cout << &food; // Outputs the memory address of food (0x6dfed4)

A pointer however, is a variable that stores the memory address as its value.

A pointer variable points to a data type (like int or string) of the same type, and is created with the \* operator. The address of the variable you're working with is assigned to the pointer:

### Example

string food = "Pizza"; // A food variable of type string  
string\* ptr = &food; // A pointer variable, with the name ptr, that stores the address of food  
  
// Output the value of food (Pizza)  
cout << food << "\n";  
  
// Output the memory address of food (0x6dfed4)  
cout << &food << "\n";  
  
// Output the memory address of food with the pointer (0x6dfed4)  
cout << ptr << "\n";

#### Example explained

Create a pointer variable with the name ptr, that points to a string variable, by using the asterisk sign \* (string\* ptr). Note that the type of the pointer has to match the type of the variable you're working with.

Use the & operator to store the memory address of the variable called food, and assign it to the pointer.

Now, ptr holds the value of food's memory address.

Tip: There are three ways to declare pointer variables, but the first way is preferred:

string\* mystring; // Preferred  
string \*mystring;  
string \* mystring;

# C++ Dereference

## Get Memory Address and Value

In the example from the previous page, we used the pointer variable to get the memory address of a variable (used together with the & reference operator). However, you can also use the pointer to get the value of the variable, by using the \* operator (the dereference operator):

### Example

string food = "Pizza"; // Variable declaration  
string\* ptr = &food; // Pointer declaration  
  
// Reference: Output the memory address of food with the pointer (0x6dfed4)  
cout << ptr << "\n";  
  
// Dereference: Output the value of food with the pointer (Pizza)  
cout << \*ptr << "\n";

Note that the \* sign can be confusing here, as it does two different things in our code:

* When used in declaration (string\* ptr), it creates a pointer variable.
* When not used in declaration, it act as a dereference operator.

# C++ Modify Pointers

## Modify the Pointer Value

You can also change the pointer's value. But note that this will also change the value of the original variable:

### Example

string food = "Pizza";  
string\* ptr = &food;  
  
// Output the value of food (Pizza)  
cout << food << "\n";  
  
// Output the memory address of food (0x6dfed4)  
cout << &food << "\n";  
  
// Access the memory address of food and output its value (Pizza)  
cout << \*ptr << "\n";  
  
// Change the value of the pointer  
\*ptr = "Hamburger";  
  
// Output the new value of the pointer (Hamburger)  
cout << \*ptr << "\n";  
  
// Output the new value of the food variable (Hamburger)  
cout << food << "\n";