Templates: CRTP

In this lesson, we will learn about curiously recurring template patterns in modern C++.

CRTP

The acronym CRTP stands for the C++ idiom **C**uriously **R**ecurring **T**emplate **P**attern. CRTP is a technique in Modern C++ in which a `Derived` class derives from a class template `Base`. The key is that `Base` has `Derived` as a template argument.

Let’s have a look at an example:

```
template<class T>
class Base{
    ...
};

class Derived: public Base<Derived>{
    ...
};
```

> CRTP enables static polymorphism.

Typical use-case

There are two typical use-cases for CRTP: Mixins and static polymorphism.

Mixins

Mixinsis a popular concept in the design of classes used to mix in new code, meaning this technique it is commonly used in Python to change the behavior of a class by using multiple inheritances. In contrast to C++, in Python, it is possible to have more than one definition of a method in a class hierarchy. Python simply uses the method that is first in the Method Resolution Order (MRO).

You can implement mixins in C++ by using CRTP. A prominent example is the class `std::enable_shared_from_this`. By using this class, you can create objects that return an `std::shared_ptr` to themselves. We must derive your class `MySharedClass` public from `std::enable_shared_from_this`. Now, our class `MySharedClass` has a method, `shared_from_this`.

An additional typical use-case for mixins is for a class that you want to extend with the capability that their instances support the comparison for equality and inequality.

Static Polymorphism

Static polymorphism is similar to dynamic polymorphism. Contrary to dynamic polymorphism with virtual methods, the dispatch of the method calls will take place at compile-time.

```
class ShareMe: public std::enable_shared_from_this<ShareMe>{
  std::shared_ptr<ShareMe> getShared(){
    return shared_from_this();
  }
};
```

- `std::enable_shared_from_this` creates a `shared _ptr` for an object.
- `std::enable_shared_from_this:` base class of the object.
- `shared_from_this:` returns the shared object

To learn more about CRTP, read here.

In the next lesson, we'll take a look at the examples of CRTP.

In the next lesson, we'll take a look at the examples of SRP.