

Smart Pointer

Es un tipo de puntero en c++ que gestiona automáticamente la memoria asignada a un objeto ayuda a prevenir problemas comunes relacionados con la gestión manual de memoria, como fugas de memoria y accesos a memoria no válidos. Un ejemplo de smart pointer en c++ es `std::unique_ptr`, que garantiza que solo un puntero a la vez tenga la propiedad de un objeto

```
#include <iostream>
#include <memory>

class MiClase {
public:
    MiClase() {
        std::cout << "Constructor de MiClase" << std::endl;
    }
    ~MiClase() {
        std::cout << "Destructor de MiClase" << std::endl;
    }
    void metodo() {
        std::cout << "Método de MiClase" << std::endl;
    }
};

int main() {
    std::unique_ptr<MiClase> puntero(new MiClase()); // Creación de un unique_ptr
    puntero->metodo(); // Acceso a un método de la clase a través del puntero
    return 0;
} // Al salir de este bloque, el destructor de MiClase se llama automáticamente y libera la memoria asignada
```

1. `std::unique_ptr`: Proporciona la propiedad exclusiva de un objeto. Solo puede haber un `std::unique_ptr` que posea un objeto en un momento dado. Cuando el `std::unique_ptr` sale del ámbito o se elimina explícitamente, libera automáticamente la memoria asignada al objeto.

2. `std::shared_ptr`: Permite que varios punteros apunten al mismo objeto. Lleva un contador de referencias interno y cuando el último `std::shared_ptr` que apunta a un objeto se elimina, se libera automáticamente la memoria asignada.

3. `std::weak_ptr`: Similar a `std::shared_ptr`, pero no aumenta el contador de referencias. Se utiliza para romper ciclos de referencia en estructuras de datos que utilizan `std::shared_ptr`.

Estos Smart Pointers son parte de la biblioteca estándar de C++ (<memory>) y proporcionan una forma segura y eficiente de administrar la memoria en aplicaciones orientadas a objetos. Al utilizar Smart Pointers, se reducen los errores de gestión de memoria, como fugas de memoria y accesos a memoria no válidos, lo que hace que el código sea más seguro y menos propenso a errores.

