

第2章 内联和嵌套命名空间

《现代C++语言核心特性解析》 谢丙堃

内联命名空间的定义和使用

```
#include <iostream>
namespace Parent {
    namespace Child1 {
        void foo() { std::cout << "Child1::foo()" << std::endl; }
    }

    inline namespace Child2 {
        void foo() { std::cout << "Child2::foo()" << std::endl; }
    }
}
int main() {
    Parent::Child1::foo();
    Parent::foo();
}
```

内联命名空间的用途

```
namespace Parent {  
    namespace V1 {  
        void foo() { std::cout << "foo v1.0" << std::endl; }  
    }  
  
    inline namespace V2 {  
        void foo() { std::cout << "foo v2.0" << std::endl; }  
    }  
}  
  
int main()  
{  
    Parent::foo();  
}
```

嵌套命名空间的简化语法

- 对比

```
namespace A {  
    namespace B {  
        namespace C {  
            int foo() { return 5; }  
        }  
    }  
}
```

```
namespace A::B::C {  
    int foo() { return 5; }  
}
```

嵌套命名空间的简化语法

- 对比

```
namespace A::B {  
    inline namespace C {  
        int foo() { return 5; }  
    }  
}  
namespace A {  
    inline namespace B {  
        namespace C {  
            int foo() { return 5; }  
        }  
    }  
}
```

```
namespace A::B::inline C {  
    int foo() { return 5; }  
}  
namespace A::inline B::C {  
    int foo() { return 5; }  
}
```



感谢聆听
欢迎关注