“IT Expert 윈도우 프로그래머를 위한 MFC 구조와 원리”

저자 서문

이 책에서 하는 일은

“MFC AppWizard(exe) → Single Document를 선택했을 때 어플리케이션 위저드(application wizard)가 생성한 코드를 모두 분석하는 것”

입니다. 우리는 코드 분석을 위해서 C++과 Win32 API를 이용해서 거의 비슷하게 동작하는 프로젝트(project)를 직접 만들 것입니다.

우리는 이 책의 독자들이

① C++의 중급 이상 사용자
② Win32 API 프로그래밍의 경험이 조금 있는 독자
③ MFC 프로그래밍의 경험이 (전혀 없거나) 조금 있는 독자

로 간주합니다. C++의 중급 이상 사용자는 가상함수를 사용하는 이유를 알고 있을 것이며, STL(standard template library)의 std::vector<int>의 의미를 알고 있을 것입니다. 또한 Win32프로그래밍의 경험이 있는 독자들은 윈도우즈 응용 프로그램의 시작 함수는 main()이 아니며, 윈도우 클라이언트 영역에 문자열을 출력하기 위해서는 printf()가 아닌 TextOut()을 사용해야 한다는 것을 알고 있을 것입니다. 즉 이 책은 프로그래밍에 입문하려는 초보자를 위한 책이 아닙니다. 그러므로 다음과 같은 독자들에게도 이 책은 적당하지 않습니다.

① C와 C++의 경험이 6개월 미만인 독자

② 학교에서 C++ 코스를 막 듣고 윈도우 프로그래밍을 시작하려는 독자
③ C++의 템플릿이나 STL을 전혀 사용해 보지 않은 독자

위의 사항에 해당하는 독자들은 좋은 C++과 Win32 API 프로그래밍 책을 골라 각 책의 반 정도를 먼저 읽어 보기 바랍니다. 이 책은 또한 MFC 프로그래밍의 경험이 전혀 없는 독자들을 염두에 두고 쓰여 졌지만, 어느 정도 MFC 프로그래밍 경험이 있는 사람에게 훨씬 유리합니다.

이 책은 MFC 프로그래밍 책이 아닙니다. 이 책의 목적은 MFC의 핵심 원리를 이해하고자 하는 것입니다. 그래서 이 책을 모두 읽은 독자들은 다른 좋은 MFC 프로그래밍 책을 골라 본격적인 공부를 시작해야 할 것입니다.

책의 소스가 편집되고 실행된 환경은 다음과 같습니다.

① 운영체제: Windows 2000(SP3)
② 컴파일러: Visual C++ 6(SP5)
③ Pentium IV 2.6

필자의 프로그래밍 경력은 1988년 대학 입학 때부터입니다. 필자는 Pascal 언어로 프로그래밍을 시작했고, 94년부터는 C 언어를 사용했으면, 95년부터는 C++ 언어로 프로그래밍 하고 있습니다. 필자는 KOG 스튜디오(www.kogstudios.com)에서 일하고 있습니다. 책의 내용과 관련한 질문은 필자에게 연락(seojt@kogstudios.com)하면 성실히 답해 드리겠습니다. 책에서 사용된 소스는 모두 실행해 보고 검증한 것입니다. 또한 성실하게 책의 내용을 검토했지만, 오류가 있을 거라 생각됩니다. 오류는 출판사의 홈페이지를 통해서 다운받을 수 있습니다.

나의 삶을 허락해 주신 하나님께 감사드립니다. 삶의 의미를 가르쳐 준 많은 분들께 감사드리며, 사랑하는 아내 미경과 예쁜 두 딸에게 감사합니다. 이 책은 나의 둘째 딸에게 바칩니다.

2005년 2월 서진택

이 책의 주요 내용

01장. 윈도우 프로그래밍

MFC 코드 작성에 필요한 부분을 중심으로 API로 윈도우 프로그래밍을 하는 방법을 설명한다. 메시지 루프와 윈도우 프로시저의 동작 원리를 포함한다.

02장. C++의 주제들 I

MFC 코드 작성에 필요한 C++의 주제들을 설명한다. 플래그, 함수포인터 변환 연산자 오버로딩을 포함한다. 독자들이 이미 C++에 익숙하다고 가정하므로, 일반적인 C++의 설명이 아니라 C++의 기능이 MFC에서 어떻게 사용되는지 위주로 설명한다.

03장. 전처리 명령어

MFC에서 사용되는 전처리 명령어를 설명한다.

04장. C++의 주제들 II

2장에서 이어지는 장으로, MFC 코드 작성에 필요한 C++의 고급 주제를 설명한다. 일반적인 C++의 설명이 아니라 그러한 기능들이 MFC에서 어떻게 사용되는지 위주로 설명한다.

05장. MFC의 디자인 패턴

디자인 패턴중 MFC에서 가장 많이 사용되는 브리지 패턴, 관찰자 패턴과 스테이트 패턴을 설명한다. 5장까지의 내용은 MFC 관련 코드를 포함하지 않는다.

06장. 메시지 맵

6장은 이 책에서 가장 중요한 장으로 가장 길다. MFC의 메시지 맵을 순수한 C언어로 시작하여 MFC의 그것까지 단계별로 구현하면서 원리를 파악한다. 또한 첫 번째 MFC 프로젝트를 작성하여, 책에서 익힌 내용이 실제로 그렇게 구현되었는지를 확인하는 단계를 거친다.

07장. 첫번째 MFC 프로젝트: Single

6장까지의 내용을 바탕으로 첫번째 MFC 프로젝트 Single을 만들고, 소스를 분석한다.

08장. CClientDC의 원리 분석

GDI의 래퍼 클래스인 CDC의 원리를 이해하기 위해 직접 CClientDC를 구현하고, MFC 코드를 확인하여 익힌 원리가 그대로 적용되었는지 파악한다.

09장. 리소스 편집

리소스 에디터의 원리와 MFC에서의 동작 과정을 설명한다.

10장. MFC의 RTTI

이 책에서 두 번째 중요한 장으로 MFC에서 RTTI를 어떻게 구현하였는지 이해하기 위해 순수한 C++로 직접 MFC의 RTTI에 해당하는 기능을 구현하고, MFC 코드에서 직접 이를 확인한다.

11장. Single 프로젝트의 소스 분석 2

7장에서 처음 작석한 MFC Single 프로젝트의 소스를 CDC와 RTTI위주로 분석한다.

12장. DDX의 원리

DDX를 직접 구현하고, MFC 코드에서 이를 직접 확인한다.

13장. 직렬화(Serialization)

직렬화를 직접 구현해 보고, MFC 코드에서 이를 직접 확인한다. MFC Single프로젝트에서 분석하지 못한 DDX와 직렬화 코드를 분석한다.

한빛이 제시하는 MFC 학습 로드맵

이 책에서 다루는 예제

읽기 전에

우리는 이 책의 내용을 설명하는데 경어를 사용하지 않는다. 그것은 이 책이 기술서적이기 때문이다. 경어를 사용하지 않음으로 해서, 독자들에게 요구하는 사항이 명령형처럼 느껴질 수도 있는데 독자들의 양해를 구한다^^.

우리는 이 책에서 제시하는 모든 프로그램 소스에 관한 설명에서 특별한 언급이 없다면 Windows 2000 운영체제에서 실행된 비주얼(Visual) C++ 6(SP5)이라고 가정한다. 예를 들면

'WINAPI위에서 F12키를 누르면'

이라는 설명이 있다면 그것은 비주얼 C++ 6.0의 에디터에서 커서를 WINAPI 위에 위치시키고 F12를 누르는 것을 의미한다.

비주얼 C++의 서비스 팩 5(service pack 5) 이상을 인스톨하는 것은 중요하다. 우리는 곳곳에서 실제 MFC(Microsoft Foundation Class)의 소스를 언급할 것인데, SP5가 인스톨되어 있지 않다면 독자들의 컴퓨터에 인스톨된 MFC의 소스와 줄 번호가 일치하지 않을 것이다.

윈도우즈(Windows)와 윈도우(Window)를 구분하자. 전자는 윈도우즈 운영체제(OS, operating system)을 후자는 특정한 응용 프로그램 윈도우를 의미한다. 모호한 상황에서 우리는 운영체제란 용어를 사용할 것인데 그것은 윈도우즈 운영체제를 의미한다.

표기 관례

소스는 가능하다면 **헝가리식 표기법(Hungarian notation)**을 따른다. 헝가리식 표기법의 규칙은 다음과 같다.

① 변수는 변수의 타입을 의미하는 소문자로 시작한다.
② 함수는 대문자로 시작하며, 함수의 이름을 구성하는 단어들은 모두 대문자로 표시한다.

완전한 소스는 특별한 소스 블록으로 표시한다. 하지만, 설명을 위해 사용된 일부 소스는 소스 블록과는 다른 강조로 표시한다.
소스의 중요한 부분이나 설명이 연결되는 부분은 굵게 표시한다. 소스에서 특별하게 지적된 설명이 있는 곳은 라인 코멘트(line comment)의 번호로 표시한다. 예를 들면 소스에 포함된 아래의 문장은

```
DECLARE_DYNAMIC() // (1)
```

특별하게 (1)을 언급한 설명이 있음을 나타낸다.

우리는 명칭(identifier)의 역할을 명확히 하기 위해 함수 이름 끝에는 ()를 붙인다. 또한 클래스 이름은 C 혹은 K로 시작하고 연이은 대문자로 표시한다.
예를 들면 Test()는 함수를 CTest는 클래스를 iTest는 변수를 m_iTest는 멤버 변수를 m_fTest는 float타입의 멤버 변수를 의미한다.

템플릿(template)이나 C++의 키워드를 명시할 때는 문법을 같이 표시한다. 예를 들면 CTest가 템플릿 클래스인 경우 CTest보다는 CTest<>로 표시한다. dynamic_cast의 경우도 dynamic_cast<>()처럼 표시한다.

많은 원어는 광범위하게 사용되는 것들을 제외하고는 원어의 발음으로 표시한다. 예를 들면 mapping은 '사상'보다는 '매핑'으로 표시한다.

그림에서 설명이 집중되는 부분은 원으로 강조하였다. 각 장은 중복된 내용을 포함하지는 않지만, 참조가 필요하다고 판단되는 부분 중 내용이 짧은 것들은 유연한 설명을 위해 부득이 중복되게 설명하였다.

차례

저자 약력 [[[[[저자 소개로 활용]]]]]

- 1969년생
- 1999년: 경북대학교 컴퓨터공학과 박사과정 수료
- 2000년: ‘만화가 있는 C’ 집필, (주)KOG 창업 멤버
- 2003년: ‘게임 개발자를 위한 C++’ 집필
- 2000년 - 2003년: 넷마블 와일드 랠리 초기버전 개발(팀장)
- 2003년 - 2004년 현재: 넥슨 범퍼킹 개발(리드 프로그래머)
- 현재: (주) KOG 범퍼킹 팀 리드 프로그래머

[문서의 끝]