

WEEK 1

1

비주얼 C++ 개발 환경과 첫 예제 프로그램 만들기

“초보자를 위한 비주얼 C++ 6 21일 완성”의 첫장을 열었으니, 이미 반은 마친 셈이다. 이후의 3주 정도의 기간 동안 마이크로소프트 비주얼 C++를 가지고 꽤 많은 애플리케이션을 같이 만들면서 공부하게 될 것이다. 매 장마다 조금씩 붙어가는 실력을 느끼면서 즐겁게 이 책을 읽어가길 바라는 마음 간절하다.

이 장에서는 비주얼 C++의 개발 환경과 애플리케이션을 만들기 위해 같이 제공된 여러 가지 도구에 대해 먼저 알아보도록 하겠다. 비주얼 C++가 제공하는 개발 도구들은 사실 1장 분량으로 공부하기에는 턱없이 부족할 정도로 엄청나고 방대하지만, 비주얼 C++를 가지고 애플리케이션을 만들 때 1차적으로 가장 많이 필요한 것만을 중심으로 다루어 주는 것이 좋을 것 같다. 하여튼 이 장에서는 다음 것들을 가뿐하게 공부하고 지나가자.

- ❖ 비주얼 C++ 개발 환경이 가진 주요한 것들
- ❖ 애플리케이션 위저드 – 이것을 사용해서 애플리케이션의 기본 골격을 구축하는 과정
- ❖ 디아일로그 에디터 – 비주얼 베이직이나 파워빌더, 델파이에서 봐왔던 디아일로그 윈도우의 시각적 편집을 가능하게 해주는 이 도구의 사용법
- ❖ 클래스위저드 – 애플리케이션 윈도우에 동작 지시를 내릴 수 있는 함수를 부착 시킬 수 있는 도구

비주얼 C++ 개발 환경

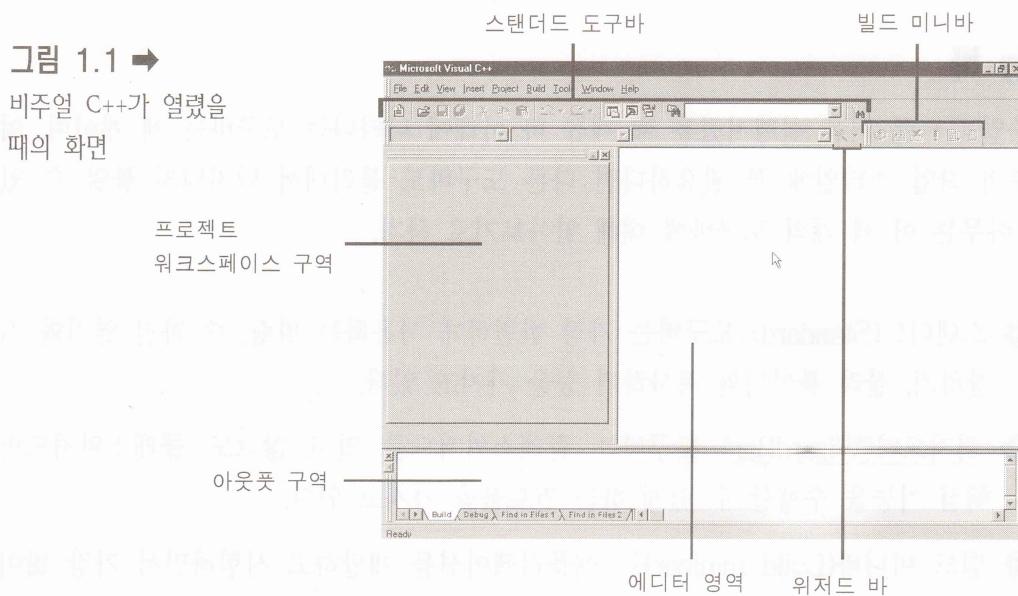
비주얼 C++의 개발 환경을 훑어보려면 일단 여러분의 시스템에 있는 비주얼 C++를 실행시켜서 각 부분이 어떻게 되어 있고, 어떻게 여러분 취향대로 다시 배열시킬 수 있을지를 알아보는 것이 급선무이다.

디벨로퍼 스튜디오(마이크로소프트의 비주얼 개발 환경을 말한다)가 시작되면, [그림 1.1]과 같은 윈도우를 보게 될 것이다. 이 윈도우의 각 부분이 디벨로퍼 스튜디오를 구성하는 나름대로의 기능을 가지고 있으며, 마음에 들지 않으면 바꿀 수도 있다.

프로젝트 워크스페이스

처음 비주얼 C++를 시작했을 때 디벨로퍼 스튜디오의 왼편에 나타나 꽤 넓은 자리를 빼어차고 앉은 녀석이 하나 있는데, 이 구역이 프로젝트 워크스페이스(project workspace)라고 부르는 것이다. 여러분의 프로젝트를 구성하고 있는 여러 가지를 일관성 있게 모두 살펴보게 해주는 역할을 가지고 있는 이 프로젝트 워크스페이스는 크게 3가지 뷰(view)를 가지고 있다.

- ◆ **클래스뷰(ClassView)**는 소스 코드를 C++ 클래스 수준에서 살펴보고 조작할 수 있게 한다.
- ◆ **리소스뷰(ResourceView)**는 애플리케이션에서 사용하는 여러 가지 리소스인 디자인로그 박스, 아이콘, 메뉴 등을 찾아서 편집할 수 있도록 한다.
- ◆ **파일뷰(File View)**는 애플리케이션을 만드는데 필요한 모든 파일을 찾아볼 수 있게 해준다.



■ 아웃풋(Output) 구역

아웃풋(Output) 구역은 비주얼 C++를 처음 시작했을 때에는 나타나지 않은 상태이며, 애플리케이션을 컴파일하고 나면 디벨로퍼 스튜디오의 아래에 불쑥 생기고 여러분이 닫을 때까지 없어지지 않는다. 아웃풋 구역은 디벨로퍼 스튜디오가 여러분에게 주었으면 하는 정보를 표시하는 역할을 한다. 즉, 여기서 컴파일러의 진행 과정, 경고 메시지, 에러 메시지 등을 볼 수 있으며, 비주얼 C++ 디버거를 실행시켰을 경우에 각각의 값을 가진 실행중의 모든 변수도 볼 수 있다. 아웃풋 구역은, 닫힌 후에도 비주얼 C++가 여러분에게 무언가를 알려주어야 할 것이 있으면 다시 열린다.

■ 에디터 영역

디벨로퍼 스튜디오 환경의 오른쪽은 에디터 영역이 자리잡고 있는데, 비주얼 C++를 사용하면서 필요한 모든 편집을 모두 여기서 할 수 있다. 즉, C++ 소스 코드를 편집할 수 있는 코드 에디터와 다이얼로그 박스를 그릴 수 있는 다이얼로그 에디터, 심지어 아이콘까지 그릴 수 있는 에디터도 이 영역에서 사용한다. 디벨로퍼 스튜디오를 기본적으로 덮고 있는 것이 에디터 영역이지만, 메뉴, 도구바 등과 자리를 같이 쓰기도 한다.

■ 메뉴 바

비주얼 C++를 처음 실행시켰을 때 메뉴 바 아래에 나타나는 도구바는 세 개이며, 여러분의 작업 스타일에 꼭 필요하다면 다른 도구바도 불러내서 덕지덕지 붙일 수 있다. 아무튼 이 세 개의 도구바에 대해 알아보기로 하자.

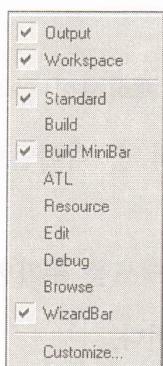
- ❖ 스텠더드(Standard) 도구바는 가장 빈번하게 사용하는 명령, 즉 파일 열기와 저장하기, 잘라 붙이기와 복사하기 등을 가지고 있다.
- ❖ 위저드바(WizardBar) 도구바는 클래스위저드를 열지 않고도 클래스위저드의 핵심 기능을 수행할 수 있게 하는 컨트롤을 가지고 있다.
- ❖ 빌드 미니바(Build minibar)는 애플리케이션을 개발하고 시험하면서 가장 많이 사용할 법한 프로그램 빌드(build)와 실행에 관련된 명령을 가지고 있다. 이것의 풀 버전인 빌드(Build) 도구바를 사용하면 여러 개의 빌드 설정사항(디버그(debug)나 릴리즈(Release) 같은)을 옮겨 다니면서 프로그램을 개발할 수 있다.

■ 디벨로퍼 스튜디오 환경 조정하기

디벨로퍼 스튜디오가 가지고 있는 도구바를 배치하는 방법은 두 가지이다. 첫번째는 도구바 영역에서 오른쪽 클릭을 하는 것으로, [그림 1.2]와 같은 팝업 문맥 메뉴가 나타나면서 여러 가지 도구바와 구역의 표시를 켜고 끌 수 있다.

그림 1.2 →

즉시 나타내고 없앨 수 있는 도구바



두번째는 도구바를 재배열할 때 사용하는 것으로, 어떤 도구바와 구역이든지 왼쪽 끝에 가지고 있는 더블 바(두 줄이 그어져 있는 부분)를 마우스로 잡고 나서 이후에 몇 가지 행동을 하는 것이다. 이 상태에서 도킹된 위치로부터 떼어내면 [그림 1.3]과 같이 부유(floating) 도구바로 만들 수 있으며, 디벨로퍼 스튜디오의 다른 테두리에 도킹시킬 수도 있다. 도구바가 도킹된 상태에서도 더블 바를 잡아끌면 위치를 조정 할 수 있게 된다.

프로젝트 워크스페이스와 아웃풋 구역의 경우, 더블 바의 위치는 각각이 어떻게 도킹되어 있느냐에 따라 위쪽 혹은 왼쪽에 나타난다.

그림 1.3 →

부유(浮遊) 도구바

**Note**

프로젝트 워크스페이스와 아웃풋 원도우의 왼편 혹은 윗쪽을 잘 보면 마우스로 잡아끌 수 있는 더블 바가 붙어 있는데, 이 위치는 현재 해당 원도우가 어디에, 어떻게 도킹되어 있느냐에 따라 다르다.

프로젝트를 처음으로 만들어 보자

여러분이 비주얼 C++로 만들 첫번째 애플리케이션은 [그림 1.4]를 보면 알겠지만 두 개의 버튼이 붙어있는 썰렁한 애플리케이션이다. 첫번째 버튼은 사용자에게 [그림 1.5]와 같은 환영 메시지를 띄워주는 역할을 하며, 두번째 버튼은 애플리케이션을 닫는다. 명하니 있지 말고 다음의 단계를 차근히 따라가야 한다.

1. 새 프로젝트 워크스페이스를 만든다.
2. 애플리케이션 위저드를 사용해서 애플리케이션의 골격을 만든다.
3. 애플리케이션 위저드가 자동으로 만들어 준 다이얼로그 박스를 재조정해서 원하는 모양으로 만든다.
4. 사용자에게 환영 메시지를 띄우는 C++ 코드를 추가한다.
5. 애플리케이션에 새 아이콘을 만들어 붙인다.

그림 1.4 ➔

여러분의 첫번째 비주얼 C++ 애플리케이션

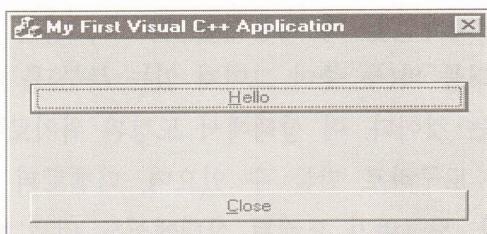
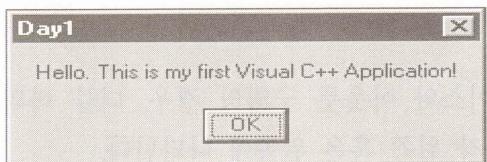


그림 1.5 ➔

사용자가 첫번째 버튼을 클릭하면 간단한 환영 메시지가 나타난다.



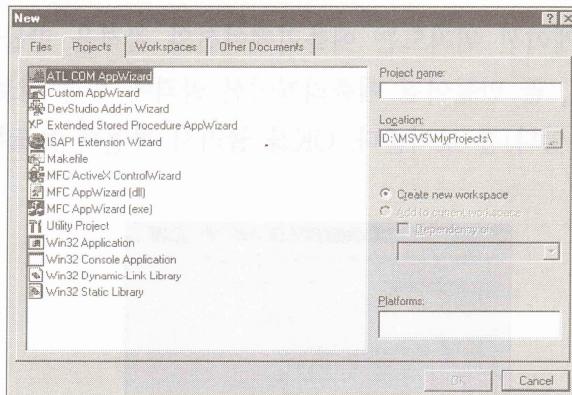
■ 프로젝트 워크스페이스 만들기

애플리케이션 개발 프로젝트라면 모두 비주얼 C++에서 각자의 프로젝트 워크스페이스를 가져야 한다. 이 워크스페이스는 애플리케이션을 구성하는 소스 코드가 어느 디렉토리에 위치해 있고 여러 가지 설정 정보를 담은 파일이 어느 디렉토리에 위치해 있는지를 디벨로퍼 스튜디오에게 알려주는 역할을 한다. 하여간 다음의 단계를 따라서 새 프로젝트 워크스페이스를 만들기로 하자.

1. File | New를 선택한다. [그림 1.6]과 같은 New 다이얼로그 박스가 나타난다.
2. Projects 탭에서 MFC AppWizard (exe)를 선택한다.
3. 여러분의 프로젝트의 이름을 Project Name 필드에 써넣는다. 멋진 이름이 생각나지 않으면 Hello가 적당할 것이다.
4. OK를 클릭한다. 이때 두 가지 일이 발생되는데, 즉 프로젝트의 디렉토리가 만들어지고(Location 필드에 설정된 대로) 애플리케이션 위저드가 시작된다.

그림 1.6 →

New 다이얼로그 박스



■ 애플리케이션 위저드를 사용해서 애플리케이션의 골격 만들기

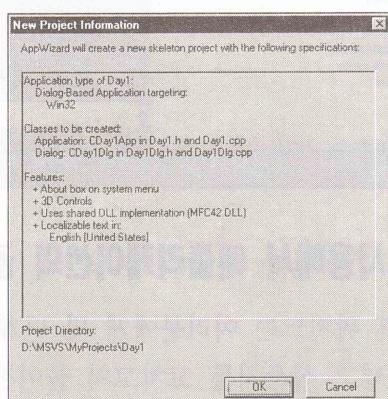
이윽고 나타난 애플리케이션 위저드는 여러분에게 몇 단계에 걸쳐서 어떤 타입의 애플리케이션에 어떤 기능을 넣고 싶은지를 꼬치꼬치 물어본다. 여러분이 입력한 정보를 바탕으로 애플리케이션의 골격을 만들어 내며, 바로 컴파일해서 실행해 볼 수 있다. 이제 여러분은 이 골격을 바탕으로 원하는 애플리케이션을 만들면 되는 것이다. 다음의 단계를 따르자.

1. 애플리케이션 위저드의 Step 1에서는 다이얼로그 기반의 애플리케이션(Dialog-based application)을 선택하고 나서 Next를 클릭한다.
2. 애플리케이션 위저드의 Step 2는 애플리케이션에 포함시킬 여러 가지 기능을 선택하도록 하는데, ActiveX 컨트롤을 사용하지 않을 것이면 ActiveX 컨트롤에 관련된 옵션을 모두 해제하도록 하자(바로 지금이 그 경우이다).
3. 이 상태에서 아래 부분의 펜드에 적혀 있는 프로젝트 이름(Hello)을 지우고 다른 이름을 쓸 수 있는데, 이 이름이 주 애플리케이션 윈도우에 나타나게 된다. 우리는 My First Visual C++ Application이라고 쓰자. Next를 클릭한다.
4. 애플리케이션 위저드의 Step 3에서는 디폴트로 설정된 대로 둔다. 즉, 소스 파일에 주석문을 추가하고 MFC 라이브러리를 DLL로 사용하도록 하는 것이다. Next를 클릭해서 마지막 단계로 가자.
5. 애플리케이션 위저드의 마지막 단계는 이제 곧 만들어질 C++ 클래스 이름을 볼 수 있는 곳이다. Finish를 클릭하면 드디어 애플리케이션 골격이 만들어지게 된다.

6. 애플리케이션 위저드는 애플리케이션의 골격을 만들어 내기 직전에 [그림 1.7]과 같이, 곧 만들어질 애플리케이션 골격에 관한 정보(여러분의 선택 결과가 반영되어 있다)를 보여준다. OK를 눌러서 확인해 보도록 하자.

그림 1.7 →

New Project Information
다이얼로그 화면



7. 애플리케이션 위저드가 애플리케이션 골격을 모두 만들어 낸 이후에는 디벨로퍼 스튜디오 환경으로 되돌아온 여러분을 느낄 수 있을 것이다. 또한, [그림 1.8]과 같이 프로젝트 워크스페이스 구역에 클래스 구조가 트리 형식으로 나와 있는 것도 새로 볼 수 있을 것이다. 에디터 영역에 메인 다이얼로그 박스가 나와 있을 수도 있다.
8. Build | Build Hello.exe를 선택해서 애플리케이션을 컴파일하자.
9. 비주얼 C++ 컴파일러가 여러분 애플리케이션을 컴파일해갈 때 아웃풋 구역을 잘 보기 바란다. 컴파일 진행 상황과 메시지가 출력될 것이다. 애플리케이션이 만들어지면, 아웃풋 구역에 “에러나 경고가 없습니다”라는 메시지가 나타난다 ([그림 1.9] 참조).
10. Build | Execute Hello.exe를 선택해서 애플리케이션을 실행시키자.
11. 강렬한 하드 디스크의 움직임과 함께 TODO 메시지와 OK, Cancel 버튼이 달린 다이얼로그 박스가 나타난다([그림 1.10] 참조). 두 버튼 중 하나만 클릭하면 애플리케이션이 끝날 것이다.

그림 1.8 →

프로젝트의 클래스 정보를
트리뷰로 가지고 있는
워크스페이스

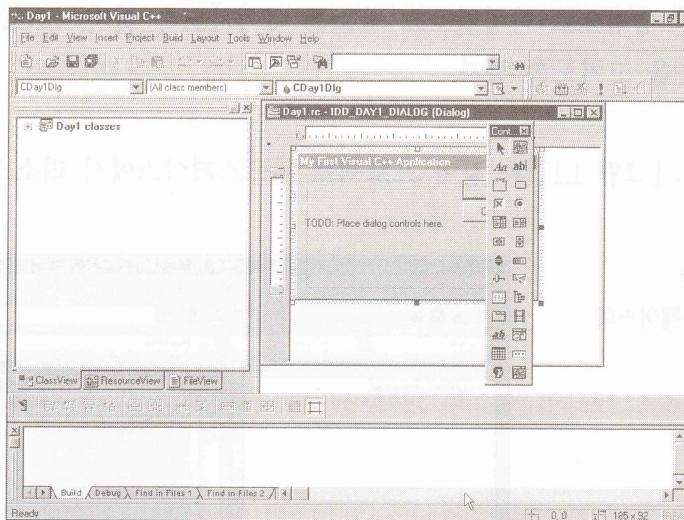


그림 1.9 →

아웃풋 윈도우에는 어떠한
컴파일러 에러라도 나타난다.

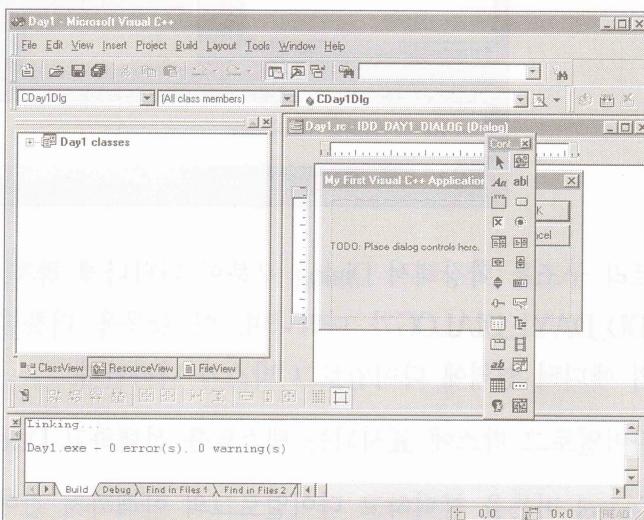
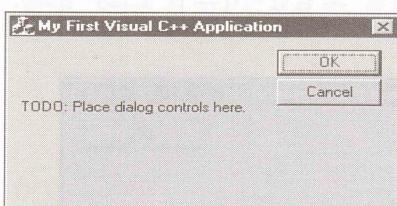


그림 1.10 →

수정되지 않은 애플리케이션
글격



애플리케이션 윈도우를 직접 꾸미자

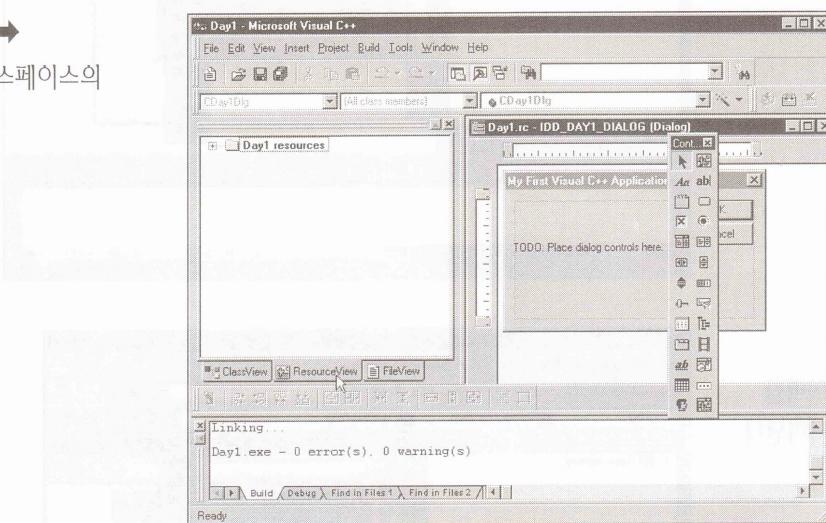
현재 만들어진 애플리케이션 글격이 실행이 안될 염려는 없기 때문에, 이제 할 일은 윈도우 꾸미기에 전력을 다하는 것이다. 메인 디자인으로 원도우가 에디터 영역에 이미 나타나 있는 경우도 있겠지만, 이후에도 이럴 것이란 법이 절대 없으므로 정상

적인 방법으로 다이얼로그 박스를 찾아가도록 하자. 그리고 나서 다이얼로그의 레이아웃을 고치도록 하자.

1. [그림 1.11]과 같이 프로젝트 워크스페이스에서 리소스뷰 탭을 선택한다.

그림 1.11 →

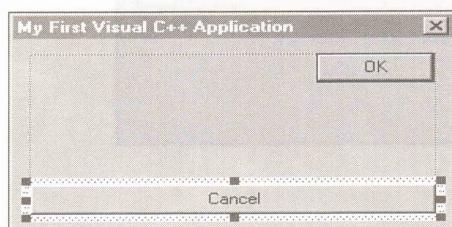
프로젝트 워크스페이스의
리소스뷰 탭



2. 트리 구조를 확장해서 Dialog 부분이 나타나게 하자. 이 부분을 한번 더 펼치면 IDD_DAY1_DIALOG가 나타나며, 이 항목을 더블클릭하면 디벨로퍼 스튜디오의 에디터 영역에 다이얼로그 박스가 나타난다.
3. 다이얼로그 박스에 표시되는 텍스트를 선택하고 Delete 키를 눌러서 지우자.
4. Cancel 버튼을 클릭하고 다이얼로그의 아래까지 잡아끈 다음, 윈도우의 폭을 완전히 차지하도록 크기를 조절한다([그림 1.12] 참조).

그림 1.12 →

Cancel 버튼을 위치시킨다.



5. Cancel 버튼에서 오른쪽 클릭하면 [그림 1.13]과 같은 문맥 메뉴가 뜨는데, 여기서 Properties를 선택하면 [그림 1.14]와 같은 프로퍼티 다이얼로그 박스가 나타나게 된다.

그림 1.13 →

마우스를 오른쪽 클릭해서 팝업 문맥 메뉴를 연다.

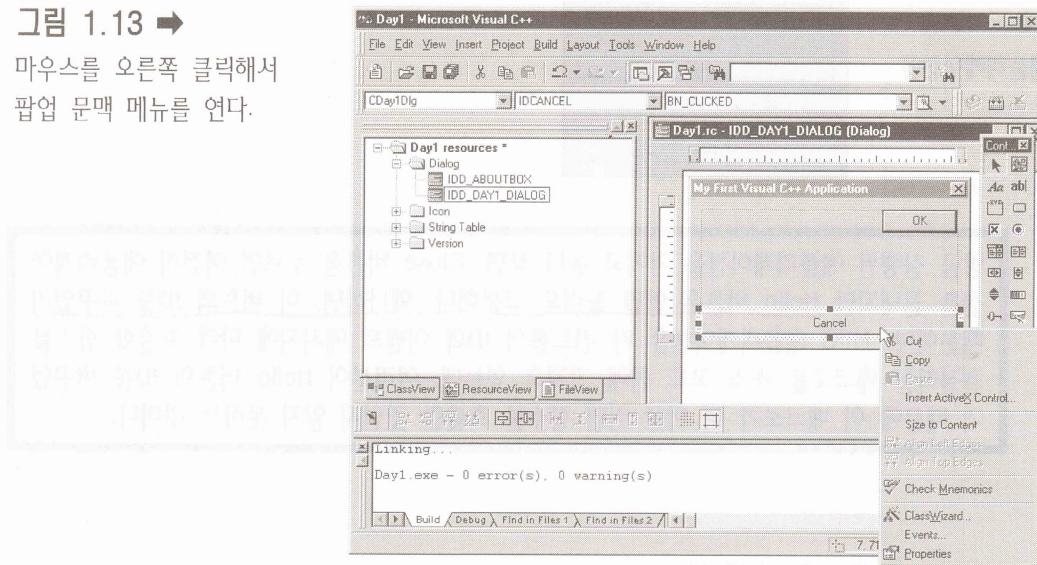
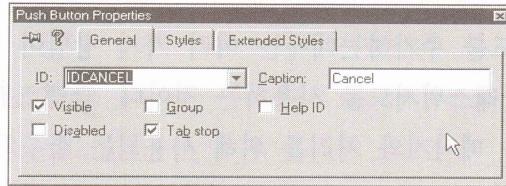


그림 1.14 →

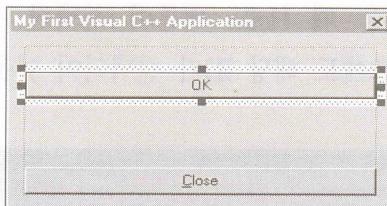
Cancel 버튼의 프로퍼티
다이얼로그



6. Caption 필드를 &Close로 바꾸고 나서 오른쪽 상단에 있는 X자 아이콘(닫기 아이콘)을 눌러서 다이얼로그 박스를 닫도록 하자.
7. OK 버튼의 위치와 크기를 [그림 1.15]와 비슷하게 만들자.

그림 1.15 →

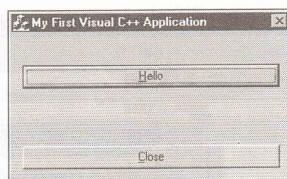
OK 버튼을 위치시킨다.



8. OK 버튼의 프로퍼티 다이얼로그 박스에서는 ID 값을 IDHELLO로 만들고, 캡션은 &Hello로 하자.
9. 이제 애플리케이션을 컴파일하고 실행시키면, [그림 1.16]과 같이 여러분이 바꾼 그대로의 다이얼로그 박스가 뜨게 된다.

그림 1.16 →

다시 꾸며서 애플리케이션을 실행시킨 모습



Note

방금 실행된 애플리케이션을 가지고 놀다 보면, Close 버튼을 누르면 여전히 애플리케이션을 끝내지만 Hello 버튼은 암만 눌러도 그뿐이다. 왜냐하면, 이 버튼의 ID를 바꾸었기 때문이다. MFC 애플리케이션은 각 컨트롤의 ID와 이벤트 메시지에 따라 호출할 함수를 결정하는 매크로를 소스 코드 안에 가지고 있는데, 여러분이 Hello 버튼의 ID를 바꾸었기 때문에 이 매크로가 더 이상 어떤 함수를 호출할 것인지 알지 못하는 것이다.

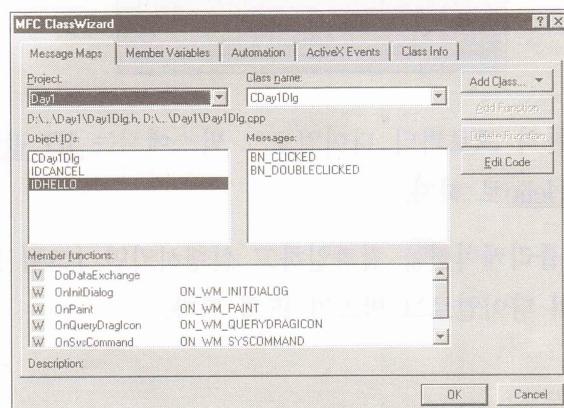
애플리케이션에 코드 추가하기

다이얼로그에 코드를 추가하는데에는 여러 가지 방법이 있을 수 있는데, 가장 많이 쓰이는 방법이 클래스위저드를 사용하는 것이다. 클래스위저드는 애플리케이션이 받아 처리할 윈도우 메시지와 처리를 위해 사용되는 함수의 테이블을 만들 때 사용하며, 이때 MFC에서 지정된 매크로가 쓰여 윈도우 컨트롤과 함수를 연결한다. 말은 좀 복잡하지만, 사용자 입장에서는 간단하게 할 수 있다. 다음의 단계를 통해서 말이다.

1. Hello 버튼을 눌렀을 때 실행될 동작을 연결해 주려면, 일단 이 버튼을 오른쪽 클릭한 다음 팝업 문맥 메뉴에서 Class Wizard를 선택한다.
2. 클래스위저드가 열렸을 때 Hello 버튼이 선택된 상태였다면 [그림 1.17]처럼 Object ID 리스트 박스에도 해당 ID가 선택되어 있을 것이다.

그림 1.17 →

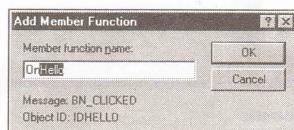
클래스위저드



3. Object ID 리스트에서 IDHELLO가 선택된 상태에서, 메시지 리스트로부터 BN_CLICKED를 선택하고 Add Function을 선택한다. [그림 1.18]과 같이 Add Member Functionダイヤログ 박스가 나타나며, 함수 이름이 하나 보일 것이다. OK를 눌러서 그대로 쓰기로 하자. 함수의 몸체와 메시지 맵 엔트리가 작성될 것이다.

그림 1.18 →

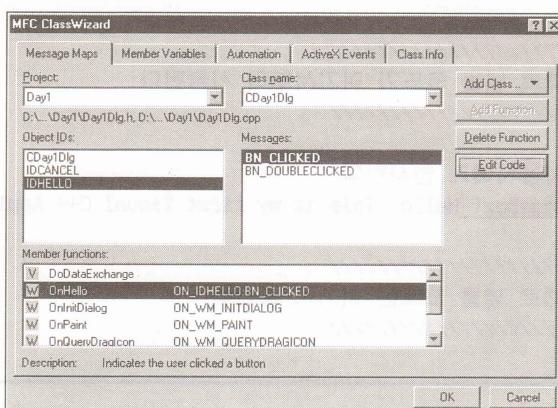
클래스위저드의 Add Member Functionダイヤログ



4. Hello 버튼을 클릭했을 때의 메시지에 함수가 연결된 다음에는, [그림 1.19]에 나온 대로 Member Function 리스트에서 OnHello를 선택하자. Edit Code 버튼을 클릭하면 해당 함수가 있는 소스 코드가 에디터 영역에서 열리면서 커서가 정확하게 위치해 있을 것이다.

그림 1.19 →

현재 사용할 수 있는 함수
리스트가 클래스위저드에
나타난다.



5. TODO 주석문 뒤에다가 [리스트 1.1]에 나온 대로 코드를 입력하자. 잘 모르겠으면 [그림 1.20]을 참고하도록 하자.

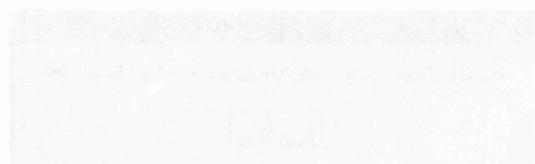
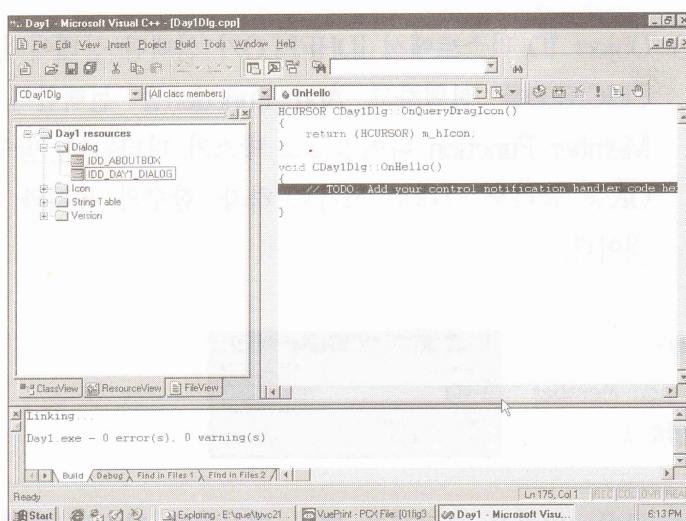


그림 1.20 →

여러분이 (리스트 1.1)을 끼워 넣을 소스코드 부분

**리스트 1.1 OnHello 함수**

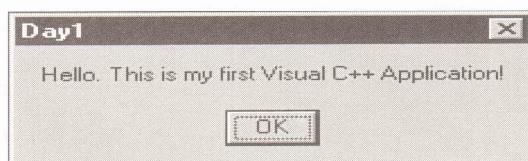
```

1: Void CHelloDlg::OnHello()
2: {
3:     // TODO: Add your control notification handler code here
4:
5:     /////////////////
6:     // 새로 넣을 코드가 여기서부터 시작된다
7:     /////////////////
8:
9:     // 사용자에게 인사한다.
10:    MessageBox("Hello. This is my first Visual C++ Application!");
11:
12:    /////////////////
13:    // 새로 넣을 코드는 여기서 끝난다
14:    /////////////////
15: }
```

6. 이제 애플리케이션을 컴파일하고 실행시킨 다음, Hello 버튼을 누르면 [그림 1.21]과 같은 메시지 박스가 나타나게 될 것이다.

그림 1.21 →

이제 여러분의 애플리케이션이 환영 메시지를 출력한다.



이제 끝마무리만 남았다

이제 애플리케이션이 모두 만들어진 상태인데, 프로젝트를 마무리 지을 두 가지 정도만이 우리를 기다리고 있다.

- ◆ 다이얼로그 박스의 아이콘 만들기
- ◆ 최대화, 최소화 버튼 추가하기

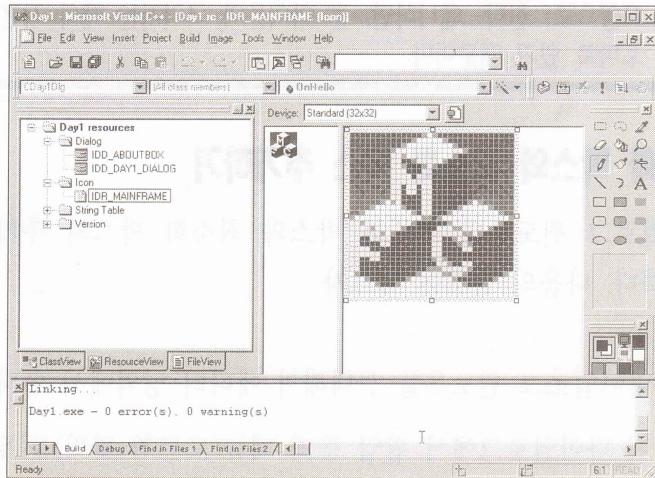
■ 다이얼로그 박스 아이콘 만들기

방금 실행해 본 애플리케이션 윈도우의 왼쪽 상단을 가만히 보면, M, F, C가 적힌 상자가 쌓여 있는 아이콘이 알 수 있는데, MFC가 대체 무슨 일을 저질렀기에 그럴까? MFC는 Microsoft Foundation Classes의 약자로서, 비주얼 C++가 사용하는 클래스 라이브러리의 이름이다. 방금 만든 애플리케이션은 이것을 토대로 한 것이다. 하지만, 여러분이 만든 멋진 애플리케이션을 M, F, C의 썰렁한 아이콘이 붙어있는 채로 남들에게 보여주기란 참으로 많은 용기를 필요로 할 것이다. 어서 서둘러서 여러분의 애플리케이션을 확실히 나타내기 위한 아이콘을 그리도록 하자.

1. 리소스뷰의 트리 구조에서 Icon 항목을 펼치고 나서 IDR_MAINFRAME을 선택한다([그림 1.22] 참조). 디벨로퍼 스튜디오의 에디터 영역에 애플리케이션 아이콘이 나타날 것이다.

그림 1.22 →

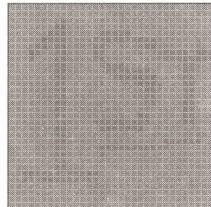
표준 MFC 아이콘



2. 옆에 같이 나타난 그리기 도구를 사용해서 여러분이 방금 만든 멋진 애플리케이션을 잘 나타내어 줄 수 있는 그림을 그려보자. 필자는 [그림 1.23]과 같이 그려보았다.

그림 1.23 →

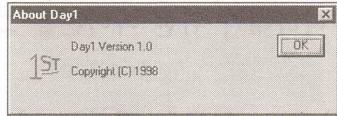
여러분의 애플리케이션을 위한 아이콘



3. 이제 애플리케이션을 다시 컴파일하고 실행하면, 왼쪽 상단에 새로 그린 아이콘이 나타났음을 확인할 수 있을 것이다. 이 아이콘을 클릭하고, 드롭다운 메뉴에서 About Hello를 선택해 보자.
4. 나타나는 Aboutダイ얼로그 박스는 비주얼 C++가 자동으로 만들어준 것인데, 중간쯤에 여러분이 그린 아이콘이 크게 나왔을 것이다([그림 1.24] 참조).

그림 1.24 →

About 윈도우



Note
아이콘 에디터로 애플리케이션의 아이콘을 열었을 때 나타나는 아이콘의 크기는 기본적으로 32×32픽셀이다. 그려지는 아이콘의 바로 위에 있는 드롭다운 박스를 사용하면 16×16픽셀 크기도 선택할 수 있다. 이 두 가지는 반드시 그려주어야(가능하면 똑같은 모양으로) 한다. 왜냐하면, 큰 아이콘이 그려지는 때와 작은 아이콘이 그려지는 때가 따로따로 있기 때문이다.

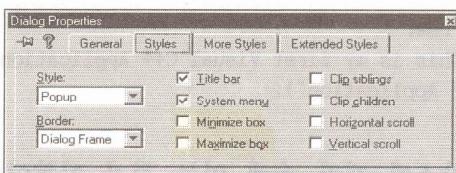
■ 최대화 박스와 최소화 박스 추가하기

다이얼로그 윈도우의 최대화 박스와 최소화 박스의 추가는 다이얼로그 에디터에서 가능하다. 다음의 단계를 따르자.

1. 다이얼로그 윈도우를 선택해서 에디터 영역에서 연다.
2. 이 다이얼로그에서 팝업 문맥 메뉴(오른쪽 클릭)를 열고 Properties를 선택한다.
3. [그림 1.25]와 같이 Styles 탭을 선택한다.

그림 1.25 →

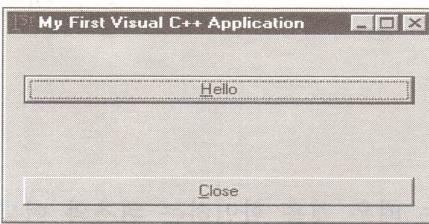
최소화, 최대화 박스를 생기게 하거나 없앨 수 있다.



4. Minimize box와 Maximize box 박스에 체크한 다음, 애플리케이션을 컴파일하고 실행시키자. [그림 1.26]처럼 타이틀바의 오른쪽에 최대화 박스와 최소화 박스가 생길 것이다.

그림 1.26 →

이제 여러분의 애플리케이션에 최대화 박스와 최소화 박스가 사라졌다.

**요약**

이 장에서는 비주얼 C++를 사용한 애플리케이션 만들기의 첫 맛을 들여보았다. 비주얼 C++의 디벨로퍼 스튜디오를 구성하고 있는 영역과 각각의 기능에 대해서 잠깐 살펴보았고, 디벨로퍼 스튜디오의 환경을 여러분의 입맛에 따라 재조정하는 방법도 간단하게나마 알아보았다. 아울러서, 비주얼 C++의 위저드를 사용해서 애플리케이션의 골격을 만들고 컨트롤에 연결동작을 설정해 보는 것으로 이번 장의 멋진 공부를 마쳤다.

Q&A

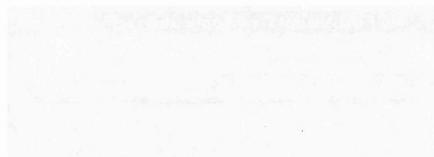
? 애플리케이션의 이름 말고 메시지 박스의 타이틀을 바꾸려면 어떻게 하나요?

! 기본적으로, 메시지 박스 윈도우는 애플리케이션의 이름을 메시지 박스의 타이틀로 사용하게 되어 있습니다. 이것을 바꾸려면 MessageBox 함수에 두번째 인수를 덧붙여주면 되죠. 이 함수의 첫번째 인수는 표시될 메시지이고, 두번째 인수는 윈도우 타이틀에 표시될 텍스트입니다. 이를테면 OnHello에 다음과 같이 합시다.

```
// Say hello to the user
MessageBox("Hello. This is my first Visual C++ Application!",
    "My First Application");
```

Q About 다이얼로그 박스에 제가 속해 있는 회사 이름이나 저작권같은 것을 넣어도 될까요?

Q About 다이얼로그 박스는 프로젝트 워크스페이스 구역의 리소스뷰에서 찾으실 수 있으세요. IDD_ABOUTBOX란 이름으로 되어 있습니다. 이것을 디자인 에디터에서 연 다음, 고치면 되지요.



실습해 보기

“실습해 보기” 절에서는 배운 것을 확인하는 퀴즈와 이를 활용해서 응용력을 높이기 위한 연습문제를 풀어볼 기회를 가질 수 있게 될 것이다. 퀴즈와 연습문제의 답은 부록 B, “퀴즈 및 연습문제 해답”에 있지만, 알아서들 하시길.

퀴즈

- 버튼의 캡션은 어떻게 바꾸는가?
- 비주얼 C++ 애플리케이션 위저드를 가지고 무엇을 할 수 있는가?
- 버튼에 어떠한 동작(함수)을 연결하려면 어떻게 하는가?

연습문제

여러분의 애플리케이션이 가지고 있는 About 윈도우에 버튼을 하나 더 추가하자. 이 버튼을 눌렀을 때 첫번째 윈도우에서의 그 버튼과 다른 메시지를 띄우도록 해보자.