

객체 지향 프로그래밍 응용

Chap 4. 대화상자와 컨트롤 (#2)



2012.10.08.

오 병 우

컴퓨터공학과
금오공과대학교

Dialog 개요

Control들을 가진 윈도우

◆ 사용자의 입력을 받기 위한 Object의 집합

- 프로그램 수행 도중 사용자의 입력이 필요할 때 다이얼로그 박스 출력
- 다이얼로그 박스는 사용자로부터 입력 받은 데이터를 메인 루틴에 넘기고 소멸

종류

◆ Modal Dialog

- Parent window 비활성화 (열기)

◆ Modeless Dialog

- Parent windows 활성화 유지 (찾기)

◆ Common Dialog

- 운영체제에 내장 (글꼴)

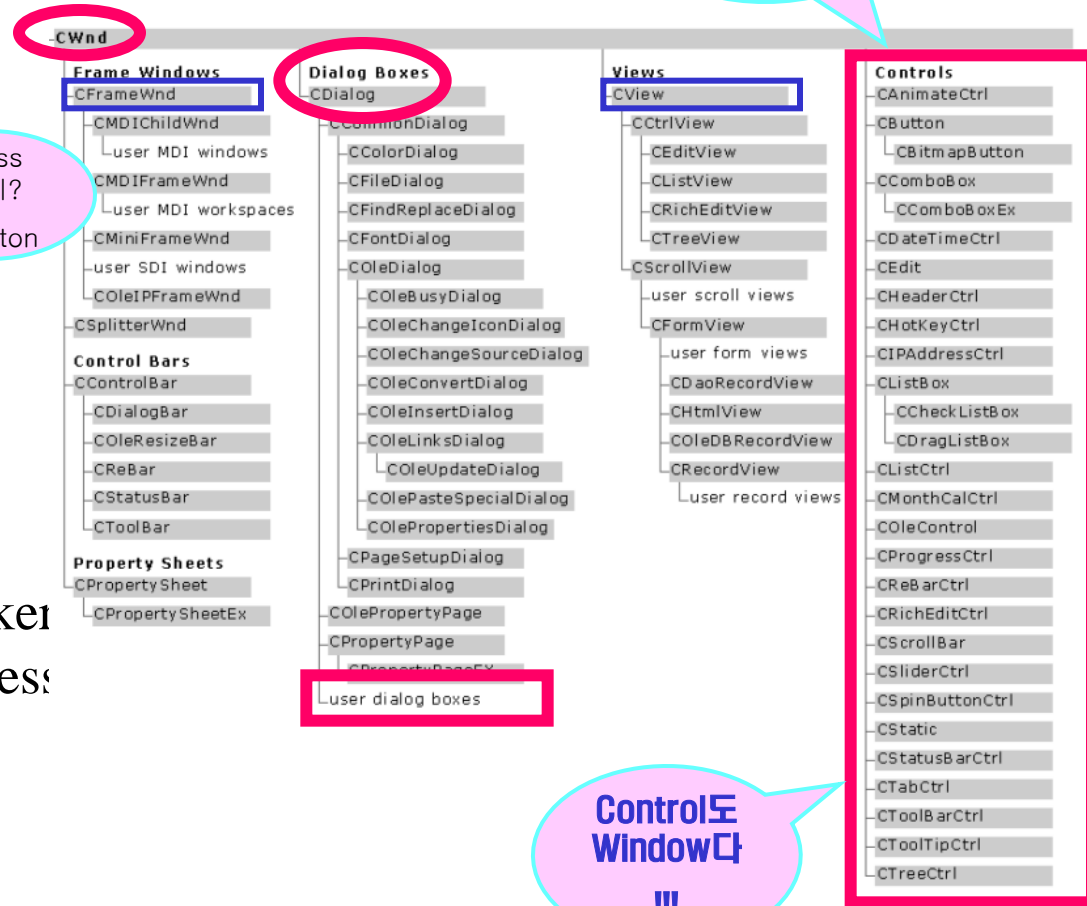
Control의 종류

같은 CWnd의
파생
클래스이므로
WM_PAINT 등
처리 가능

Control

- ◆ Button, *Check Box*
- ◆ Edit, Spin
- ◆ Picture, Static Text
- ◆ *Group Box, Radio Button*
- ◆ List Box, Combo Box
- ◆ Slider, Scroll Bar
- ◆ Hot key, Tree, List
- ◆ Rich Edit, Date Time Picker
Month Calendar, IP Address
Progress, Tab, Animation

Class
없네?
CButton

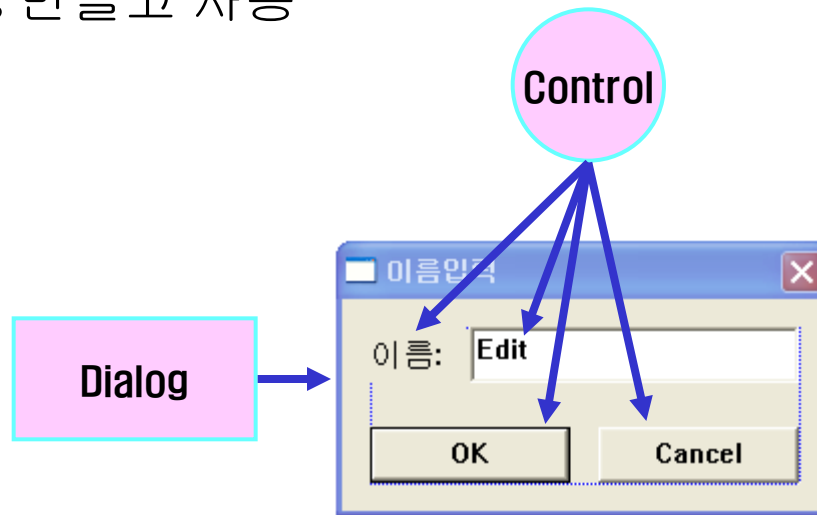


Control도
Window다

!!!

MFC Class 및 MFC 파생 Class

- Dialog 위의 Control
 - ◆ MFC Class 그대로 사용 가능
- 사용자 정의 Dialog
 - ◆ CDialog의 파생 Class 만들고 사용
 - Function overriding
 - Message Map



CDialog 파생 Class에 추가해야 하는 기능

● 사용자가 컨트롤에 입력한 내용 관리

◆ Value Type Member Variable

- e.g., Edit Control의 입력 내용
- DDX_Text(- , **ID, 변수**)

● 컨트롤 제어

◆ Control Type Member Variable

- e.g., Edit Control의 Disabled (비활성화)
 - m_ctrlBeauty.EnableWindow(FALSE);
- DDX_Control(- , **ID, 변수**)

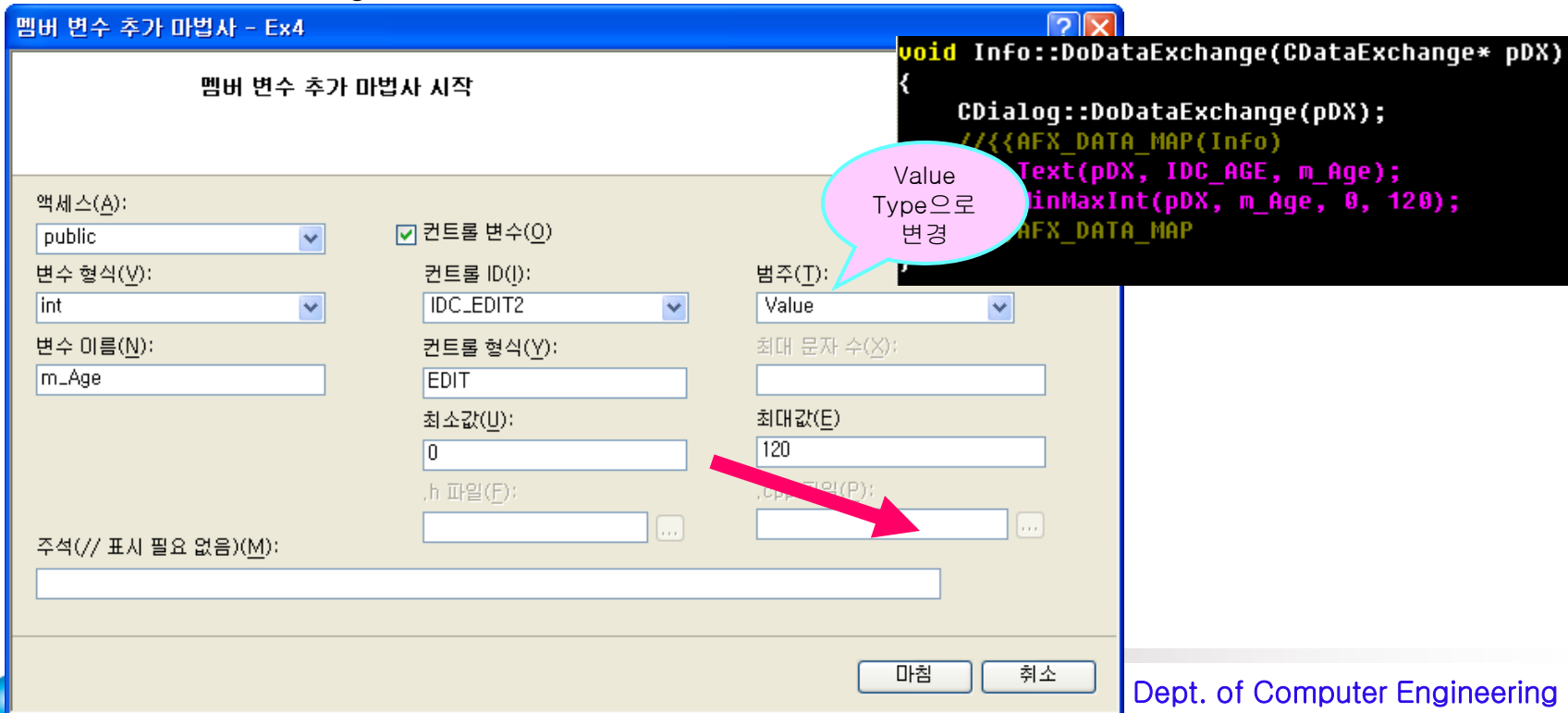
IDC_
로 시작

IDC_
로 시작

DoDataExchange()
함수 내에서 처리

Value Type Member Variable

- 멤버 변수 생성
 - ◆ Ctrl을 누른 상태에서 Control 더블 클릭
- DDV(Dialog Data Validation)도 지정가능
 - ◆ 멤버 변수 추가 마법사에서 지정
 - int: Min & Max Value
 - CString: Max Characters



멤버 변수 추가 마법사 - Ex4

멤버 변수 추가 마법사 시작

액세스(A): public

변수 형식(Y): int

변수 이름(N): m_Age

주석(// 표시 필요 없음)(M):

컨트롤 변수(O)

컨트롤 ID(I): IDC_EDIT2

컨트롤 형식(Y): EDIT

최소값(U): 0

최대값(E): 120

범주(T): Value

최대 문자 수(X):

이 파일(F):

이 파일(P):

마침 취소

```
void Info::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    //{{AFX_DATA_MAP(Info)
    Text(pDX, IDC_AGE, m_Age);
    MinMaxInt(pDX, m_Age, 0, 120);
    //AFX_DATA_MAP
```

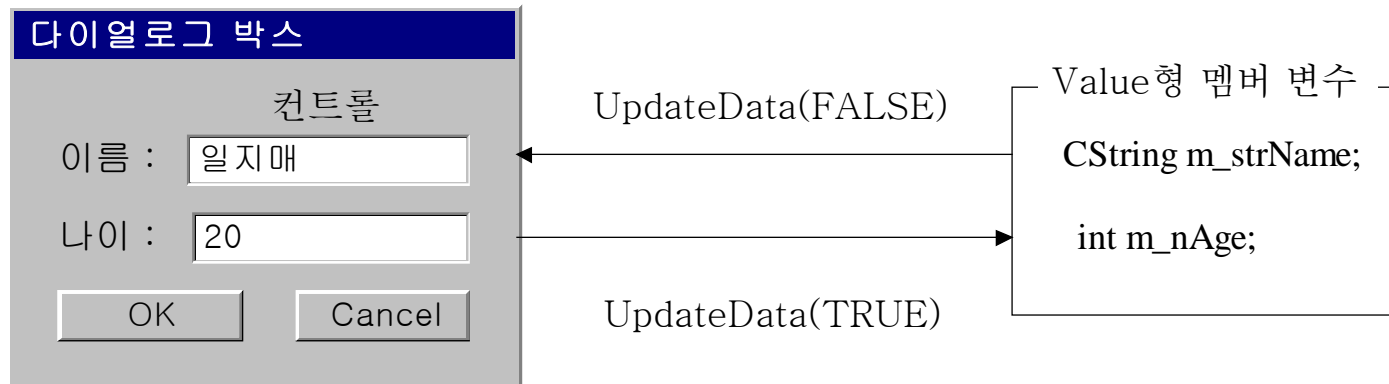
Control별 Value Type Member Variable

- CheckBox
 - ◆ BOOL
- Radio Button
 - ◆ int
 - Group 별로 Tab order에 따라서 몇 번째가 선택되었는지 정수
- Combo Box & ListBox
 - ◆ CString
 - 선택된 문자열
- Edit
 - ◆ CString, int, UINT, long, DWORD, float, double, BYTE, short, BOOL, COleDateTime, COleCurrency
- Spin
 - ◆ 직접 연결 못함
 - Edit Control과 indirect 연결
 - Auto Buddy, Set Buddy Integer, Alignment-Right로 설정

Control과의 Data Sync.

UpdateData()

◆ TRUE: 화면에서 가져옴



Dialog의 처리 과정

Constructor

Dialog 시작 DoModal()

Window에서
는
WM_CREATE

VS2003부터는
Overriding으로
변경됨

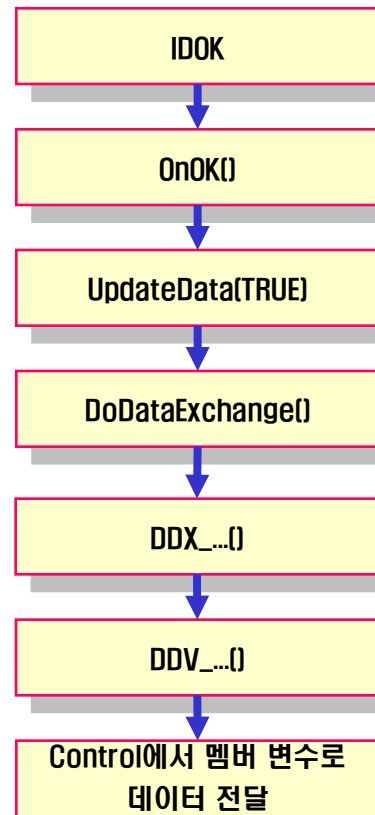
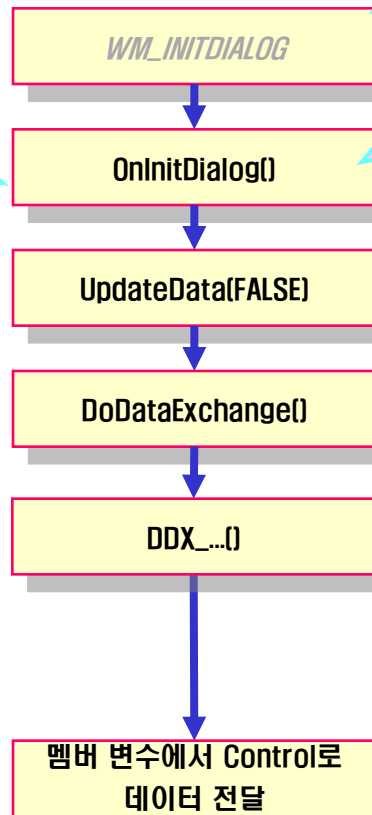
사용자
입력

Dialog
Data
eXchange

Dialog
Data
Validation

Destructor

Dialog 종료 OK 버튼



컨트롤 초기화는
여기에~

Control Type Member Variable

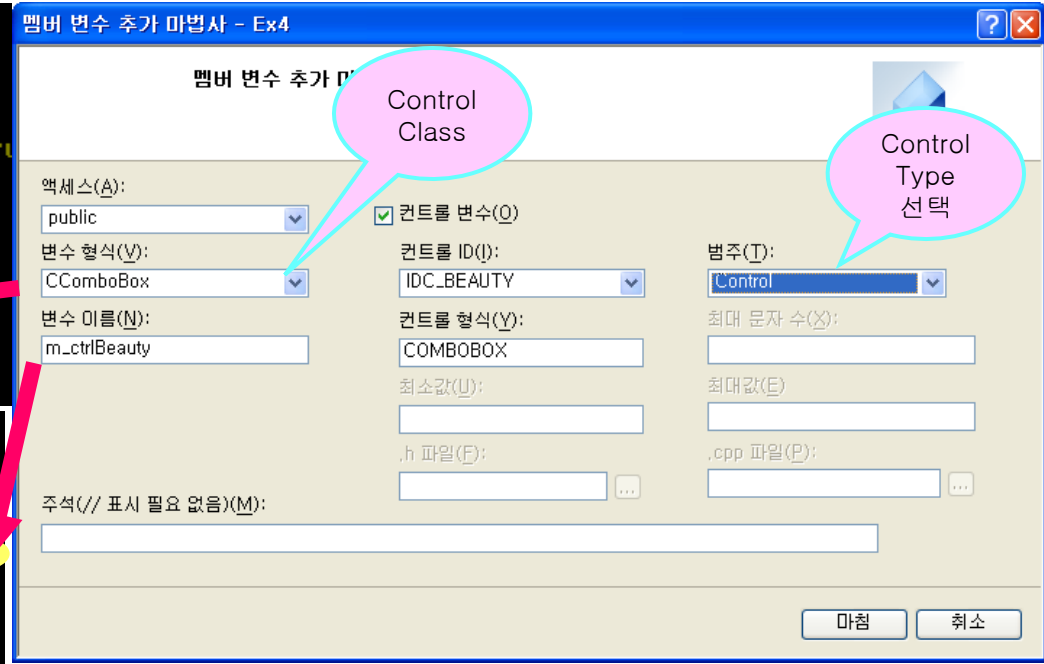
- Dialog 위에 놓여진 Control들을 직접 제어
- 컨트롤과 Control형 멤버변수 연결
 - ◆ Ctrl을 누른 상태에서 Control 더블 클릭
 - ◆ Category(범주)에서 Control 선택

```

class Info : public CDialog
{
// Construction
public:
    Info(CWnd* pParent = NULL); // standard construction

// Dialog Data
//{{AFX_DATA(Info)
enum { IDD = IDD_DIALOG1 };
    CComboBox m_ctrlBeauty;
    int m_Age;
//}}AFX_DATA

void Info::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    //{{AFX_DATA_MAP(Info)
    DDX_Control(pDX, IDC_BEAUTY, m_ctrlBeauty);
    DDX_Text(pDX, IDC_EDIT1, m_Age);
    DDU_MinMaxInt(pDX, m_Age, 0, 120);
    //}}AFX_DATA_MAP
}
    
```



Control Member Functions

주요 함수 (MSDN 참조)

◆ CWnd

- Create, ShowWindow, **EnableWindow**, SetWindowPos, SetWindowText

◆ CEditClass

- GetSel, SetSel, ReplaceSel, GetLineCount, GetLine

◆ CButton (Check Box 및 Radio Button 포함)

- GetCheck, SetCheck

◆ CSpinButtonCtrl

- GetRange, SetRange, GetPos, SetPos, GetBuddy, SetBuddy

◆ CScrollBar

- GetScrollRange, SetScrollRange, GetScrollPos, SetScrollPos

◆ CListBox, CComboBox

- GetCount, GetCurSel, SetCurSel, AddString, InsertString, DeleteString, ResetContent, Dir

Control 초기화

Dialog의 OnInitDialog()를 재정의하고 여기서 초기화

◆ DoModal()을 해야지 비로소 윈도우 생성

- WM_INITDIALOG
- c.f.) 일반 윈도우는 WM_CREATE

VS2003부터는
Overriding으로
변경됨

◆ 초기상태를 비활성화로 만들기

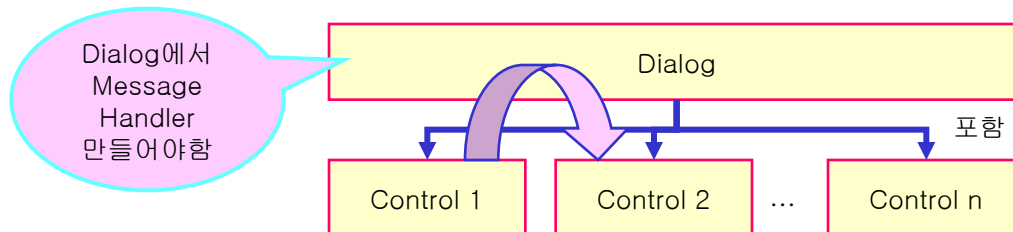
- m_[control 변수명].EnableWindow(FALSE);

Event

◆ Control로부터 Dialog로 전달되는 Message

◆ Control 자체에서 처리하는 것이 아니고 Dialog를 거쳐서 처리

- 자체에서 처리하면 다른 Control 처리가 힘들기 때문
- Control의 파생 Class를 만들지 않아도 됨



Notification Messages

주요 Message

◆ Button (Check Box 및 Radio Button 포함)

– BN_CLICKED, BN_DOUBLECLICKED

◆ Edit

– EN_CHANGE, EN_SETFOCUS, EN_KILLFOCUS

◆ List Box

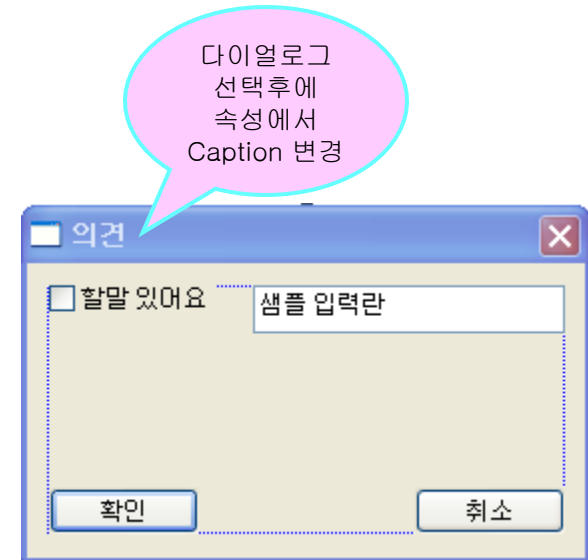
– LBN_DBLCLK, LBN_SELCHANGE,
LBN_SETFOCUS, LBN_KILLFOCUS

◆ Combo Box

– CBN_DBLCLK, CBN_SELCHANGE, CBN_SETFOCUS,
CBN_KILLFOCUS, CBN_EDITCHANGE

실습

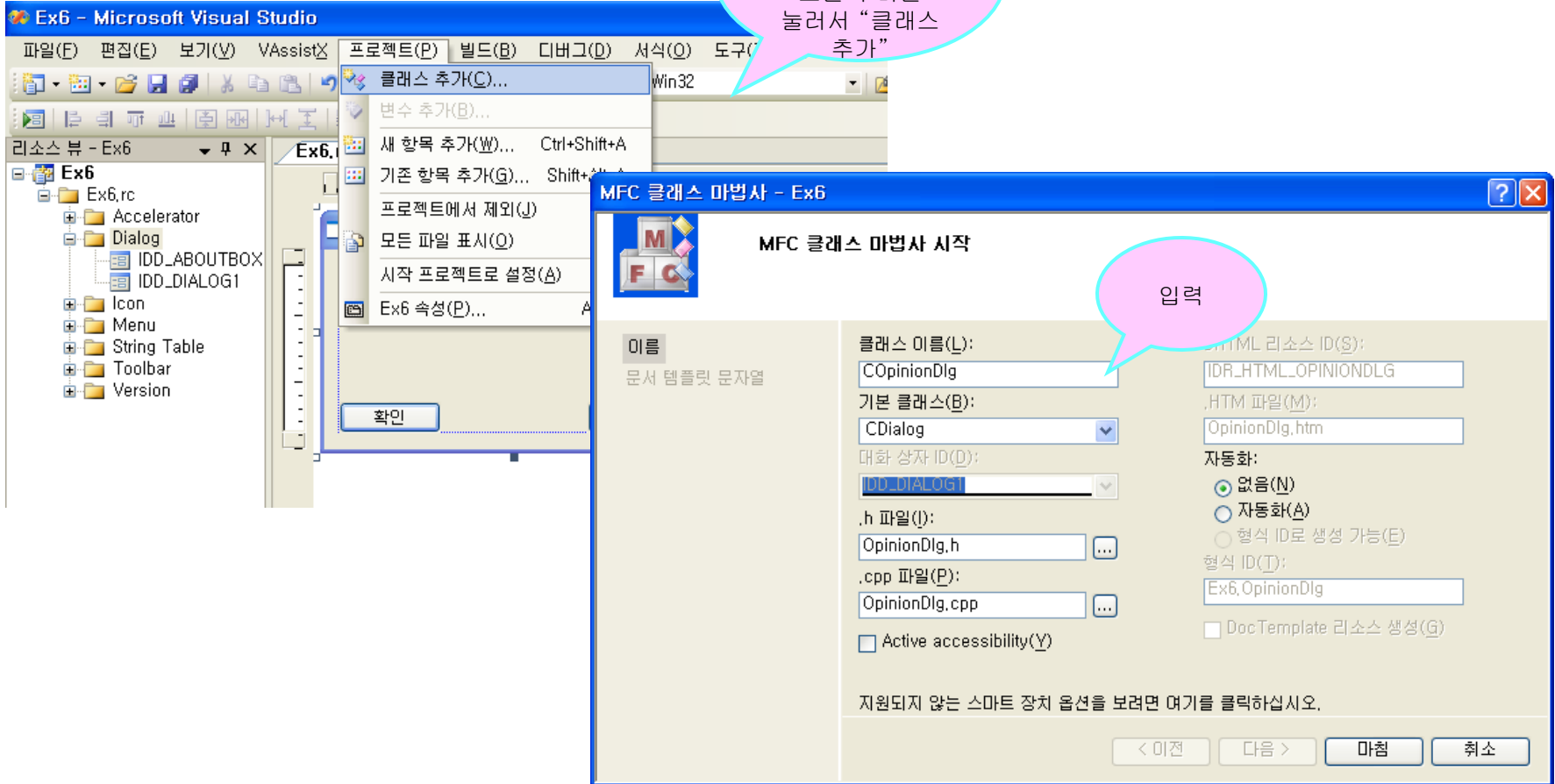
Project 및 Dialog 생성



실습

CDialog의 파생 Class 생성

또는
다이얼로그
선택후 마우스
오른쪽 버튼
눌러서 “클래스
추가”



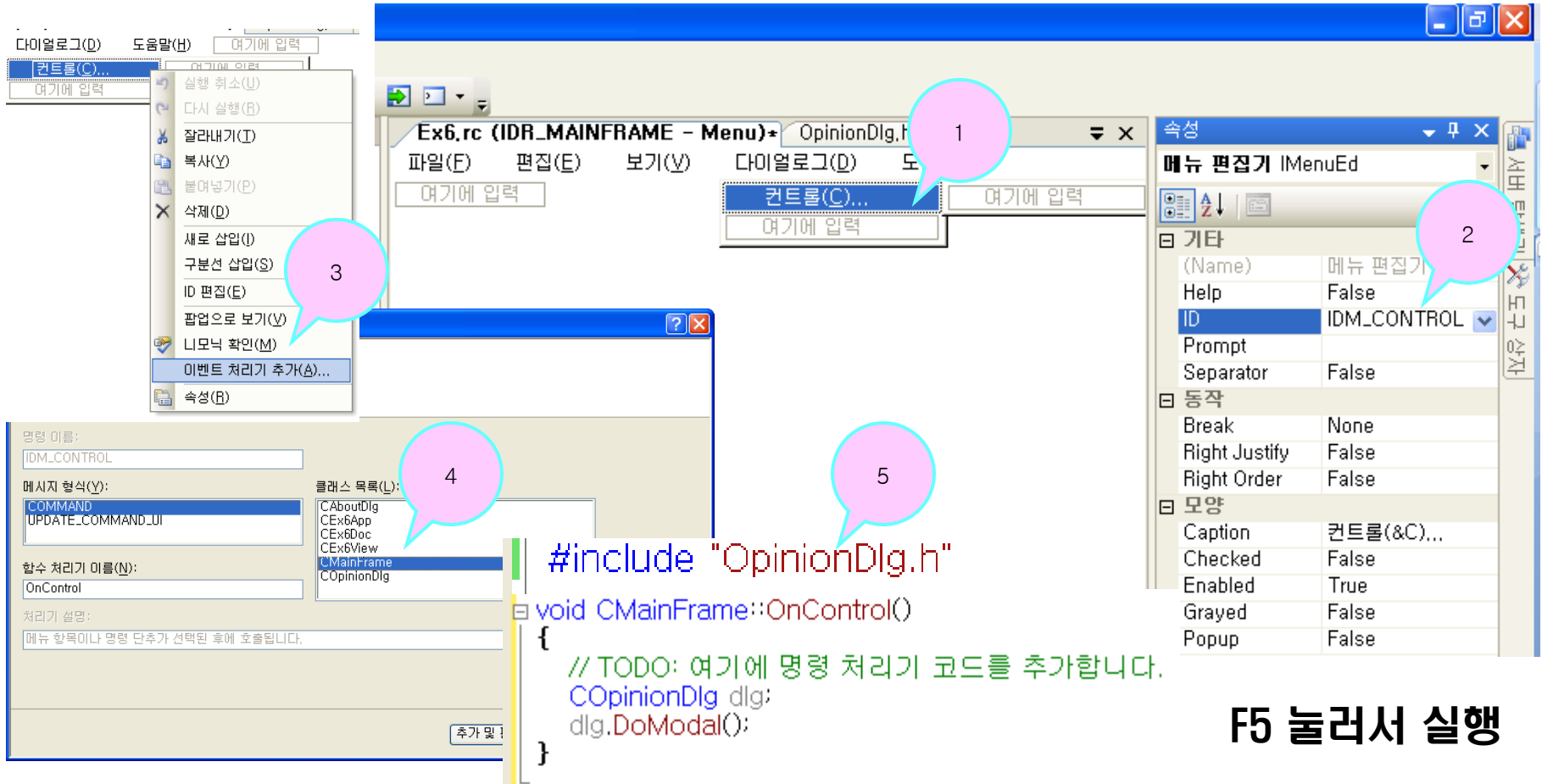
The screenshot shows the Visual Studio IDE with the 'Add Class' menu open. The 'MFC Class Wizard' dialog box is also open, showing the configuration for a new class named 'COpinionDlg' derived from 'CDialog'. The dialog box has several fields and options:

- 이름 (Name):** 문서 템플릿 문자열
- 클래스 이름 (L):** COpinionDlg
- 기본 클래스 (B):** CDialog
- 대화 상자 ID (D):** IDD_DIALOG1
- .h 파일 (I):** OpinionDlg.h
- .cpp 파일 (P):** OpinionDlg.cpp
- Active accessibility (Y)
- HTML 리소스 ID (S):** IDR_HTML_OPINIONDLG
- .HTM 파일 (M):** OpinionDlg.htm
- 자동화 (Automation):**
 - 없음 (N)
 - 자동화 (A)
 - 형식 ID로 생성 가능 (E)
- 형식 ID (T):** Ex6,OpinionDlg
- DocTemplate 리소스 생성 (G)

A pink callout bubble with the text '입력' (Input) points to the '클래스 이름 (L)' field.

실습

메뉴 연결



1

2

3

4

5

```

#include "OpinionDlg.h"

void CMainFrame::OnControl()
{
    // TODO: 여기에 명령 처리기 코드를 추가합니다.
    COpinionDlg dlg;
    dlg.DoModal();
}
    
```

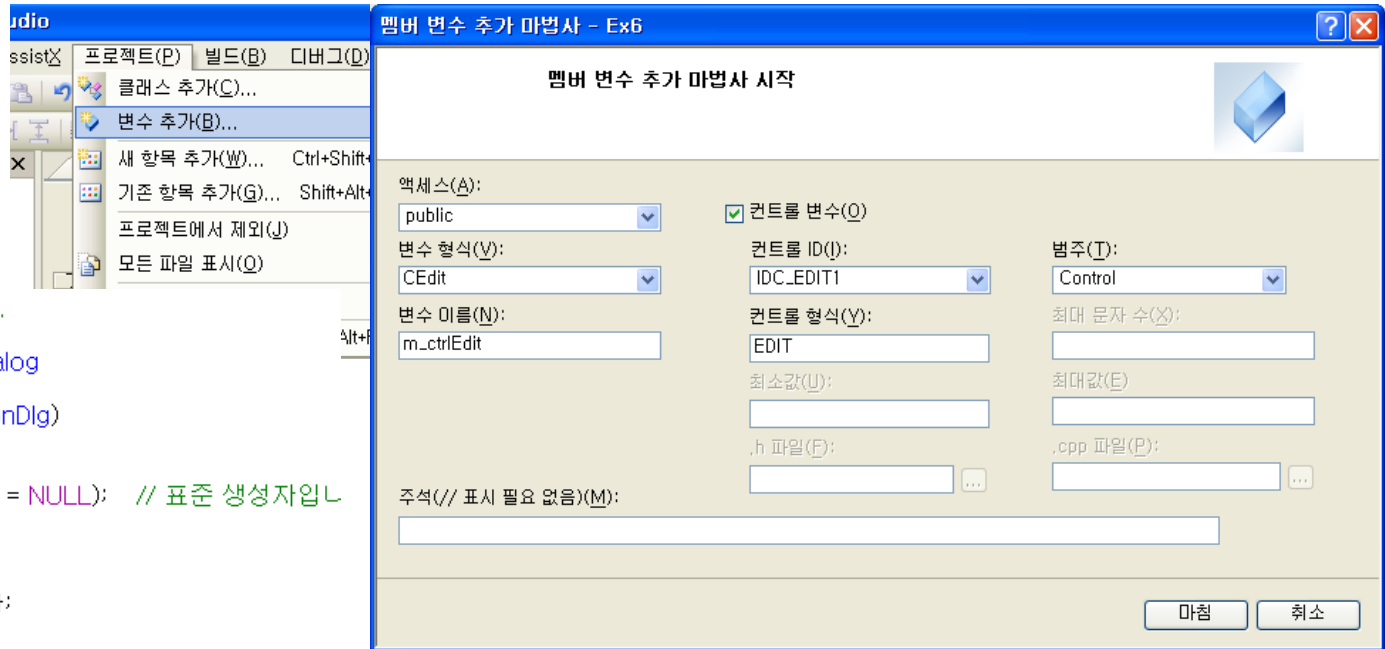
메뉴 편집기 IMenuEd	
(Name)	메뉴 편집기
Help	False
ID	IDM_CONTROL
Prompt	
Separator	False
동작	
Break	None
Right Justify	False
Right Order	False
모양	
Caption	컨트롤(&C)...
Checked	False
Enabled	True
Grayed	False
Popup	False

F5 눌러서 실행

실습

● Edit Control을 위한 Control Type Member Variable 생성

◆ Edit Control 선택 후에 Ctrl+DoubleClick



Header File

```
// COpinionDlg 대화 상자입니다.
class COpinionDlg : public CDialog
{
    DECLARE_DYNAMIC(COpinionDlg)

public:
    COpinionDlg(CWnd* pParent = NULL); // 표준 생성자입니다.
    virtual ~COpinionDlg();

// 대화 상자 데이터입니다.
    enum { IDD = IDD_DIALOG1 };

protected:
    virtual void DoDataExchange(CDataExchange* pDX); //

    DECLARE_MESSAGE_MAP()
public:
    CEdit m_ctrlEdit;
};
```

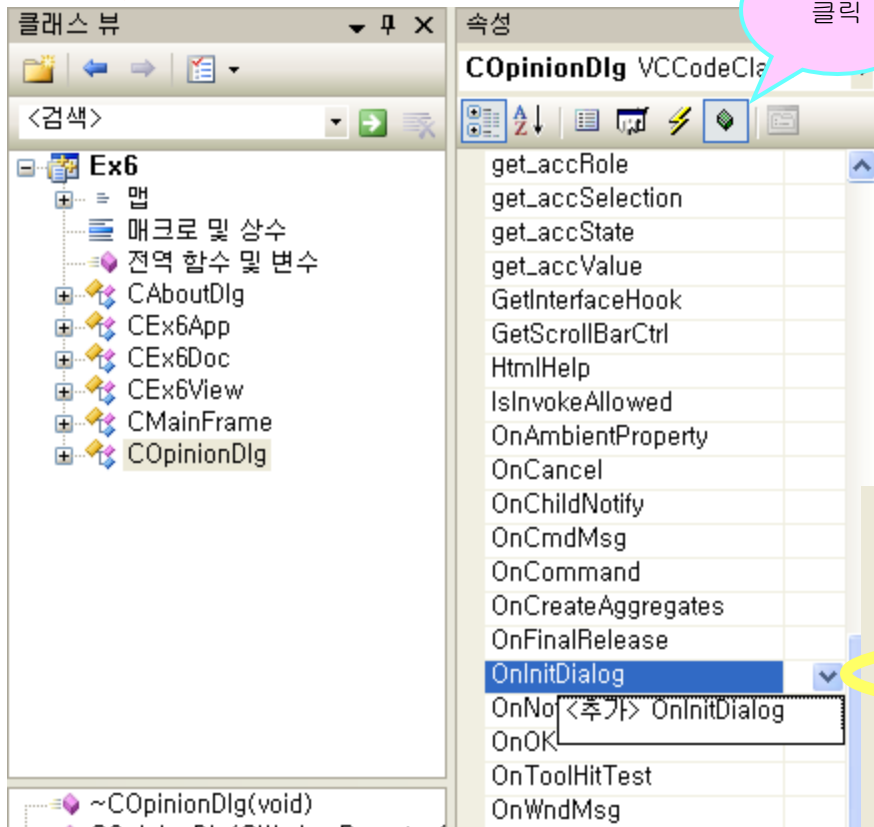
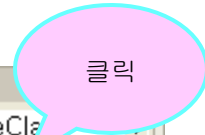
Source File

```
void COpinionDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    DDX_Control(pDX, IDC_EDIT1, m_ctrlEdit);
}
```

실습

OnInitDialog() Function Override

◆ Edit Control을 비활성화



```

BOOL COpinionDlg::OnInitDialog()
{
    CDialog::OnInitDialog();

    // TODO: 여기에 추가 초기화 작업을 추가합니다.
    m_ctrlEdit.EnableWindow(FALSE);


    return TRUE; // return TRUE unless you set the focus to a control
    // 예외: OXC 속성 페이지는 FALSE를 반환해야 합니다.
}
    
```

F5 눌러서 실행

실습

CheckBox를 위한 Value Type Member Variable 생성

◆ CheckBox 선택 후에 ctrl+DoubleClick



```

#pragma once
#include "afxwin.h"

// COpinionDlg 대화 상자입니다.
class COpinionDlg : public CDialog
{
    DECLARE_DYNAMIC(COpinionDlg)

public:
    COpinionDlg(CWnd* pParent = NULL); // 표준 생성자
    virtual ~COpinionDlg();

// 대화 상자 데이터입니다.
    enum { IDD = IDD_DIALOG1 };

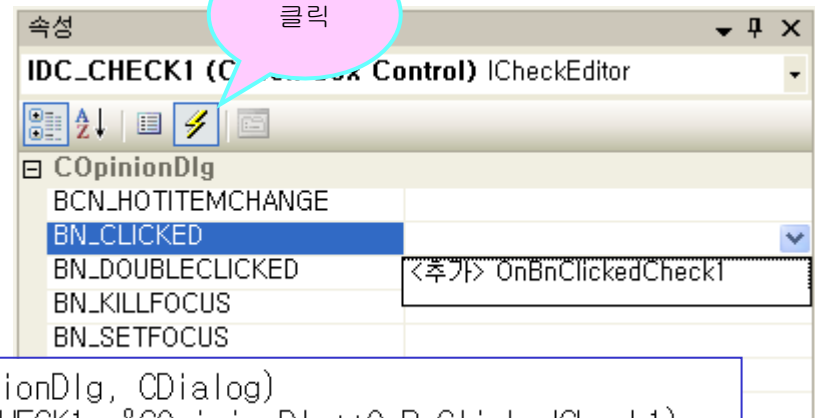
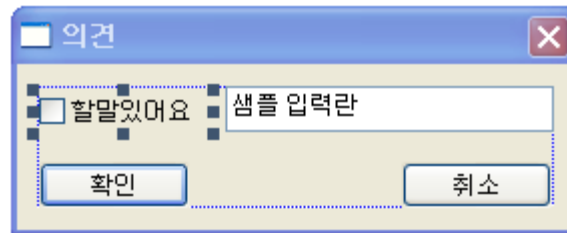
protected:
    virtual void DoDataExchange(CDataExchange* pDX);

    DECLARE_MESSAGE_MAP()
public:
    CEdit m_ctrlEdit;
public:
    BOOL m_Check;
};
        
```

실습

Control Notification Message Handler

◆ BN_CLICKED 이벤트를 처리할 함수 추가



생성된 코드 확인

```
COpinionDlg.cpp
BEGIN_MESSAGE_MAP(COpinionDlg, CDialog)
    ON_BN_CLICKED(IDC_CHECK1, &COpinionDlg::OnBnClickedCheck1)
END_MESSAGE_MAP()
```

화면(Layout)으로 부터 Control의 정보를 소스코드로 가져옴

```
void COpinionDlg::OnBnClickedCheck1()
{
    // TODO: 여기에 컨트롤 알림 처리기 코
    UpdateData(TRUE);

    m_ctrlEdit.EnableWindow(m_Check);
}
```

화면으로부터 가져온 m_Check값이 TRUE이면 Edit 활성화, FALSE이면 비활성화

F5 눌러서 실행

Summary

Relationship Diagram

