

11장

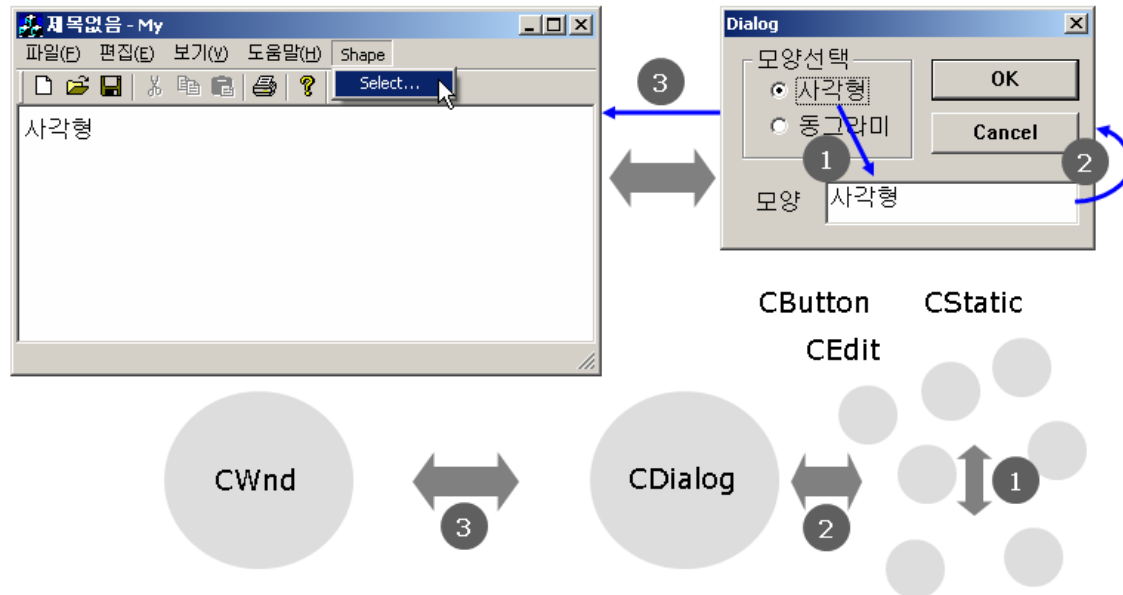
대화상자

김성영교수
금오공과대학교
컴퓨터공학부

대화상자 기본

선택사항이 많은 경우 대화상자가 메뉴 보다 유용

일반적인 대화상자 사용시 세 가지 유형의 데이터 흐름



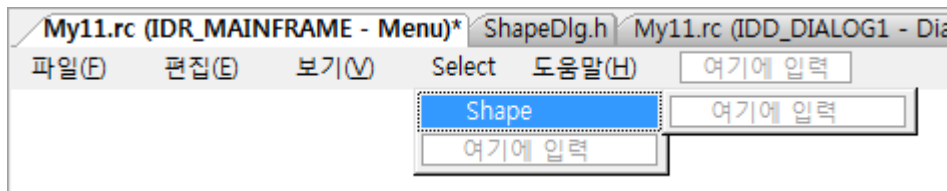
- 표면적으로 부모 윈도우와 대화상자 윈도우 사이의 데이터 흐름
- **내부적**으로 다음 상황에서 데이터 흐름 존재
 - ① 컨트롤 윈도우 객체간
 - ② 컨트롤 윈도우 객체와 대화상자 객체간
 - ③ 대화상자 객체와 부모 윈도우 객체간
- 에디트 컨트롤의 문자열을 부모 윈도우에 출력하는 과정
 - ① 사각형 모양에 대한 라디오 버튼 컨트롤을 선택하면
에디트 컨트롤에 “사각형”출력
 - ② 에디트 컨트롤에 쓰여진 문자열을 대화상자 객체에 넘김
 - ③ OK 버튼을 누르면 대화상자 객체에 저장된 문자열을
부모윈도우 객체에 넘기고 부모윈도우 객체는 화면에 출력

실습 11.1

앞에서 예시한 대화상자를 사용하는 프로그램을 작성하자.

• 메뉴

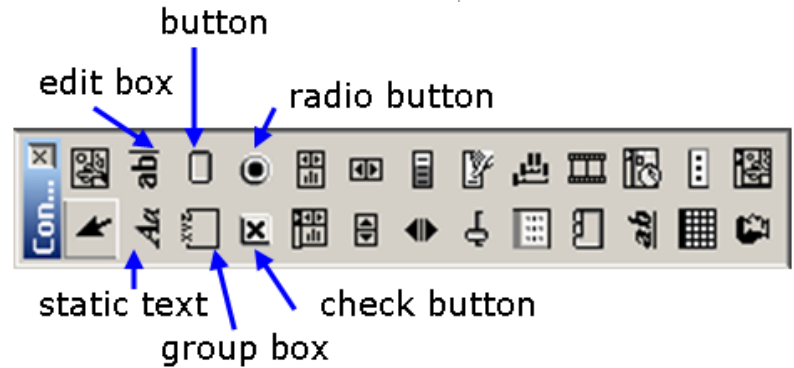
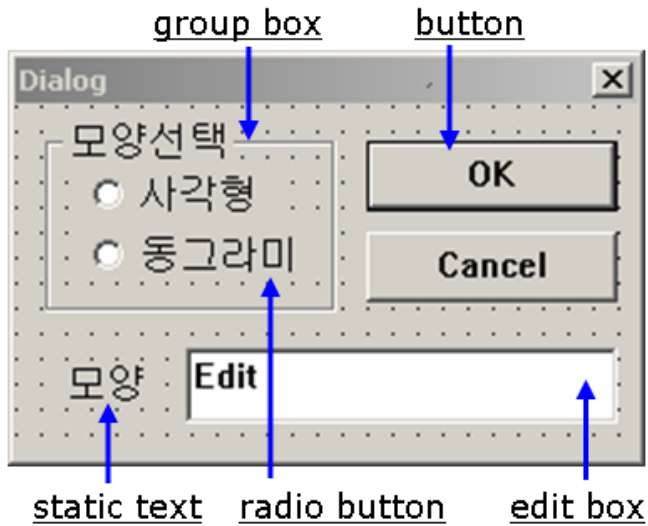
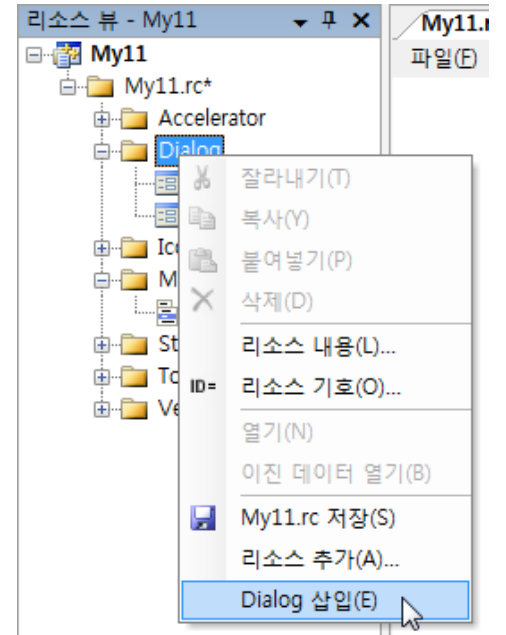
- IDR_MAINFRAME에 Select>Shape 메뉴항목 추가
- ID_SELECT_SHAPE ID 부여

A screenshot of the '속성' (Properties) window for the 'Shape' menu item. The window title is '속성' and the object is '메뉴 편집기 IMenuEd'. The properties are organized into sections: '기타' (General), '등작' (Appearance), and '모양' (Appearance).

속성	
메뉴 편집기 IMenuEd	
기타	
(Name)	메뉴 편집기
Help	False
ID	ID_SELECT_SHAPE
Prompt	
Separator	False
등작	
Break	None
Right Justify	False
Right Order	False
모양	
Caption	Shape
Checked	False
Enabled	True
Grayed	False
PopUp	False

- 대화상자

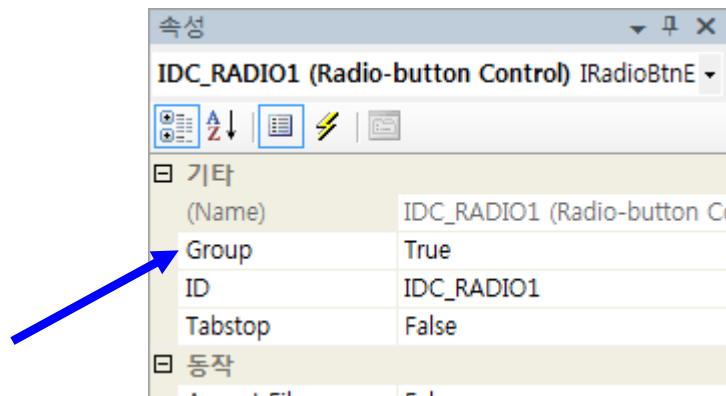
- 컨텍스트 메뉴의 Dialog 삽입을 클릭
- IDD_DIALOG1을 추가한후 컨트롤 배치



컨트롤과 ID 설정

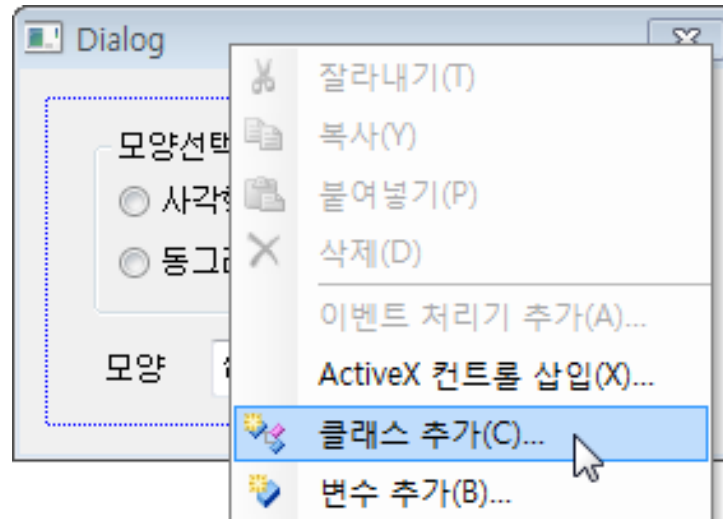
컨트롤	관련 클래스	이름	ID
radio button	CButton	사각형	IDC_RADIO_RECTANGLE
radio button	CButton	동그라미	IDC_RADIO_CIRCLE
edit box	CEdit	-	IDC_EDIT_SHAPE
button	CButton	OK	IDOK
button	CButton	Cancel	IDCANCEL

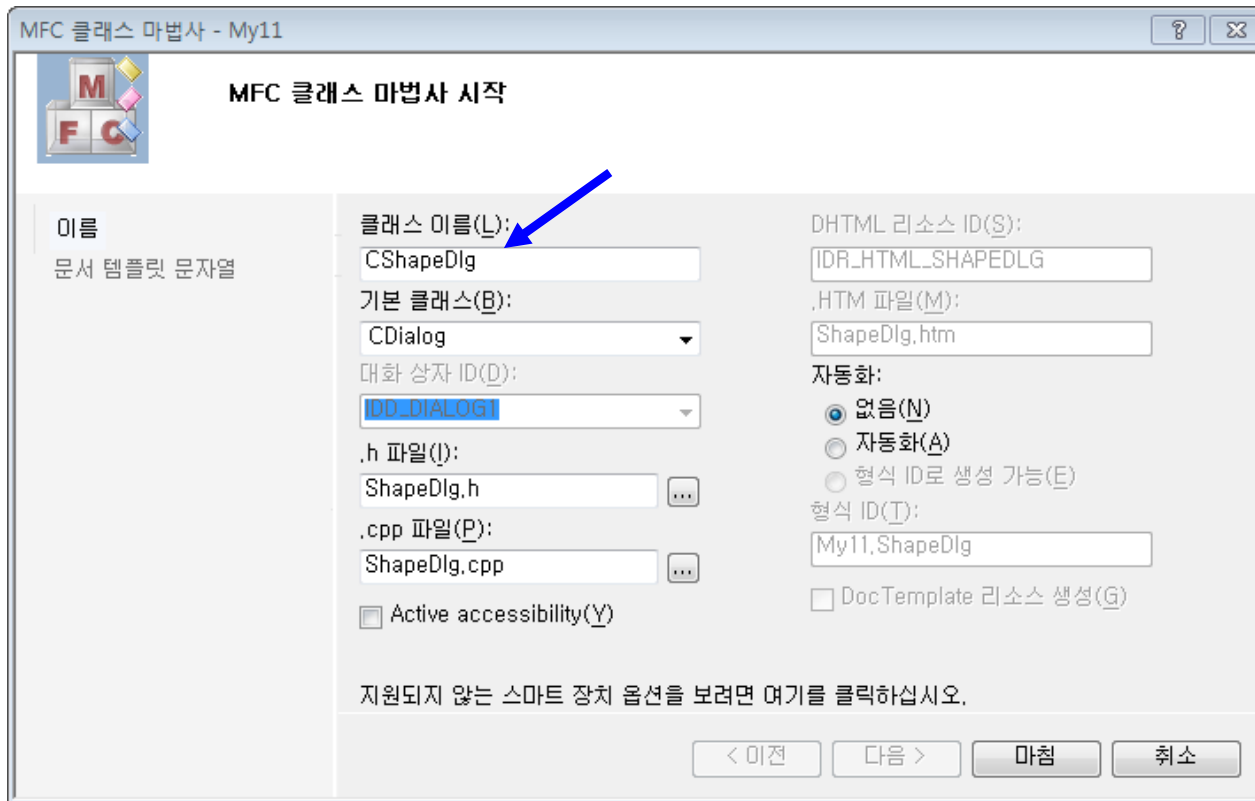
라디오 버튼은 동일 그룹 내의 버튼들과 상호배타적으로 선택
- 사각형 라디오 버튼에 대해 Group 속성 설정



대화상자 클래스 생성

대화상자 자원 마련 후 이를 이용해 `CDialog` 클래스를 만들
편집한 대화상자 자원에서 팝업 메뉴를 선택 후 위저드 실행





클래스 이름에 CShapeDlg라고 클래스 명 입력(File name 창에 ShapeDlg.cpp)후 OK → 프로젝트에 CShapeDlg 클래스가 기본 틀을 갖춰 추가

대화상자 호출 및 종료: 데이터흐름 ③

- 위저드를 이용해 뷰클래스의 **ID_SELECT_SHAPE** 핸들러 설정
- 대화상자를 화면에 출력하기 위해 **DoModal()** 함수 호출

MyView.cpp

```
#include "shapedlg.h"    //대화상자클래스 헤더파일 포함
...
void CMyView::OnShapeSelect( )
{
    CShapeDlg dlg( this ); //대화상자 객체생성
    dlg.DoModal(); //대화상자 화면출력
}
```

버튼의 기능

버튼	호출함수	기능	DoModal() 반환값
OK	CDialog::OnOK()	대화상자 닫음	IDOK
Cancel	CDialog::OnCancel()	대화상자 닫음	IDCANCEL

- 대화상자의 종료는 `CDialog::OnOK()`를 호출하여 처리
- 호출한 쪽에서 어느 버튼이 눌렸는지는 `DoModal()` 함수 반환값으로 확인

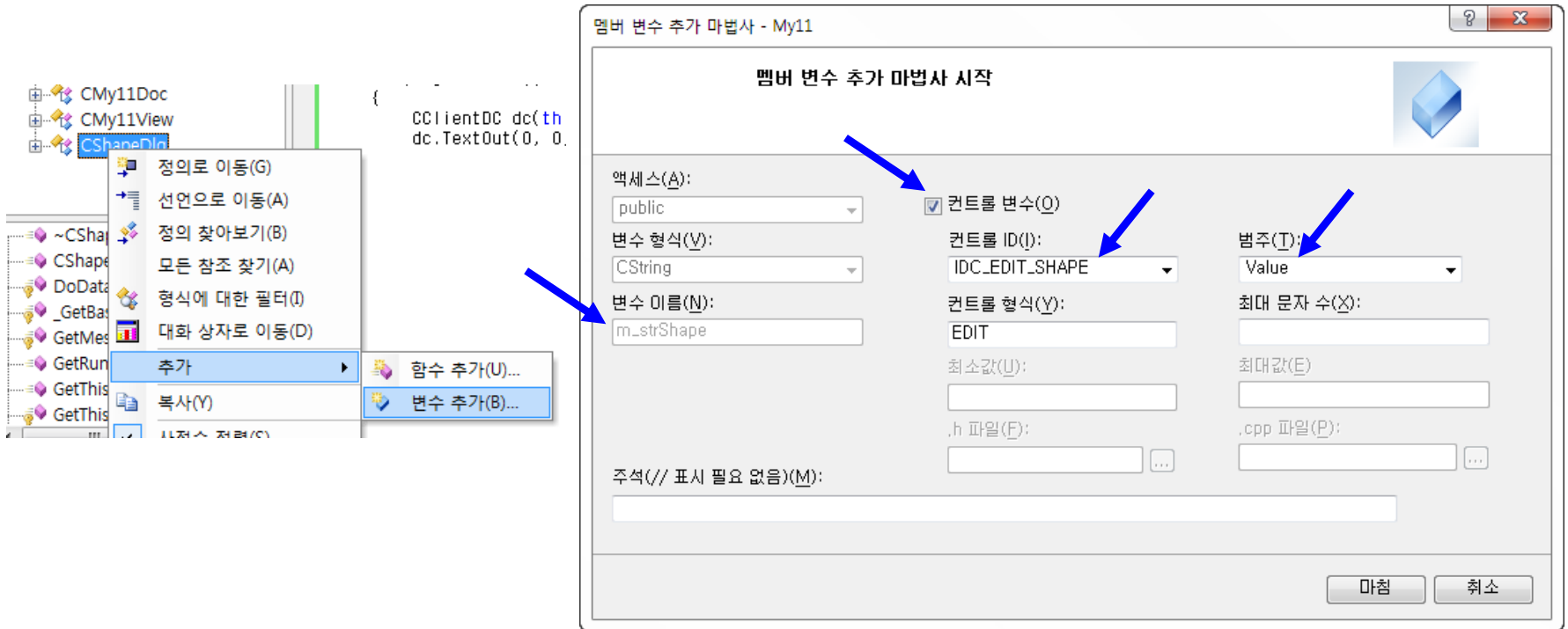
MyView.cpp

```
#include "shapedlg.h"
...
void CMyView::OnShapeSelect( )
{
    CShapeDlg dlg( this );
    if ( dlg.DoModal()==IDOK ) // 대화상자 반환값 확인
    {
        CClientDC dc( this );
        dc.TextOut( 0, 0, dlg.m_strShape );
    }
}
```

대화상자 호출 및 종료: 데이터흐름②

에디트 컨트롤에 입력된 내용을 CShapeDlg 클래스의 m_strShape에 전달
⇒ 대화상자 데이터 교환(Dialog Data eXchange: DDX) 기능 사용

CShapeDlg 클래스의 멤버 변수 추가 - 위저드 사용



ShapeDlg.h

```
class CShapeDlg: public CDialog
{
public:
    CString m_strShape;    // 에디트 컨트롤과 연동된 멤버변수(위저드가 생성)
}
```

ShapeDlg.cpp

```
CShapeDlg::CShapeDlg(CWnd* pParent /*=NULL*/)
    : CDialog(CShapeDlg::IDD, pParent)
    , m_strShape(_T(""))
{
    // m_strShape = _T( "" );
}

void CShapeDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    DDX_Text( pDX, IDC_EDIT_SHAPE, m_strShape );
}
```

- **DDX_Text()** 매크로
⇒ **IDC_EDIT_SHAPE** 컨트롤과 **m_strShape** 변수 연결
- 상호 연동 함수

```
BOOL CWnd::UpdateData( BOOL bSaveAndValidate = TRUE )
```

- **IDOK**에 대한 핸들러를 추가하여 에디트 컨트롤에 있는 정보를 **m_strShape**에 전달

ShapeDlg.cpp

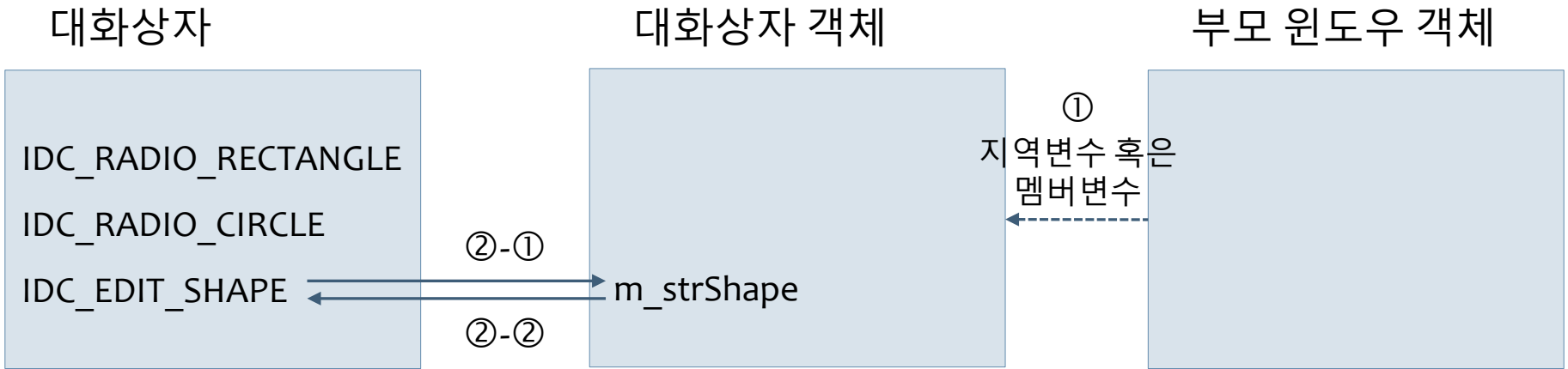
```
void CShapeDlg::OnBnClickedOk( )  
{  
    UpdateData( TRUE ); //에디트 컨트롤에 있는 값을 m_strShape로 전달  
  
    OnOK(); //대화상자 종료  
}
```

컨트롤간의 정보교환: 데이터흐름①

라디오 버튼의 선택에 따라 에디트 컨트롤에 "사각형" 또는 "동그라미" 출력

ShapeDlg.cpp

```
void CShapeDlg::OnBnClickedRadioCircle()  
{  
    m_strShape = "동그라미";  
    UpdateData( FALSE ); //m_strShape의 내용을 에디트 컨트롤에 전달  
}  
  
void CShapeDlg::OnBnClickedRadioRectangle( )  
{  
    m_strShape = "사각형";  
    UpdateData( FALSE ); //m_strShape의 내용을 에디트 컨트롤에 전달  
}
```



②-① 과정 처리: `UpdateData(true)`

②-② 과정 처리: `UpdateData(false)`

변수명이 `m_strText`인 다른 컨트롤에 연결된 변수에서 데이터 획득

```
void CShapeDlg::OnRadioCircle( )
{
    UpdateData( TRUE ); //어떤 컨트롤의 내용을 m_strText에 전달
    m_strShape = m_strText;
    UpdateData( FALSE ); //m_strShape의 내용을 에디트 컨트롤에 전달
}
```

Lab.

사각형 라디오 버튼이 초기에 선택된 상태로 표시되도록 하자.

- `InitDialog()` 함수를 재정의
- `CButton::setCheck()` 함수 사용하여 라디오 버튼의 상태 변경
 - `BST_CHECKED`, `BST_UNCHECKED`

`InitDialog()`: [http://msdn.microsoft.com/ko-kr/library/fwz35s59\(VS.80\).aspx](http://msdn.microsoft.com/ko-kr/library/fwz35s59(VS.80).aspx)

`setCheck()`: [http://msdn.microsoft.com/ko-kr/library/31deb72x\(VS.80\).aspx](http://msdn.microsoft.com/ko-kr/library/31deb72x(VS.80).aspx)


```
CShapeDlg::CShapeDlg(CWnd* pParent /*=NULL*/)
    : CDialog(CShapeDlg::IDD, pParent)
    , m_strShape(_T("사각형"))
{
    //m_strShape = _T("");
}

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

    CButton *pBtn;
    pBtn = (CButton*)GetDlgItem( IDC_RADIO_RECTANGLE );
    pBtn->SetCheck( BST_CHECKED );

    return TRUE; // return TRUE unless you set the focus to a control
    // 예외: OCX 속성페이지는FALSE를 반환해야 합니다.
}
```

GetDlgItem() 함수

- UpdateDate() 함수가 특정 컨트롤과 변수 사이의 정보교환만 처리 하는 것은 아님 → 불필요한 정보교환 발생 가능
- 컨트롤간 정보교환은 DDX를 사용하지 않고도 가능
→ 윈도우 객체의 주소사용
- 지정한 ID를 가진 컨트롤 윈도우 객체 반환 함수

```
CWnd* CWnd::GetDlgItem( int nID );
```

- 대화상자 위에 있는 컨트롤 윈도우를 다룰 수 있음
- 라디오 버튼의 핸들러에서 UpdateData() 함수 사용없이 IDC_EDIT_SHAPE 에디트 컨트롤에 값 출력 가능
- 컨트롤 윈도우 객체 확보 → CWnd 클래스가 제공하는 다양한 처리 가능

```
void CShapeDlg::OnRadioCircle()
{
    //m_strShape = "동그라미";
    CButton* pBtn;
    pBtn = (CButton*)GetDlgItem( IDC_RADIO_CIRCLE );
    pBtn->GetWindowText( m_strShape ); //라디오버튼의 이름에서 "동그라미" 획득

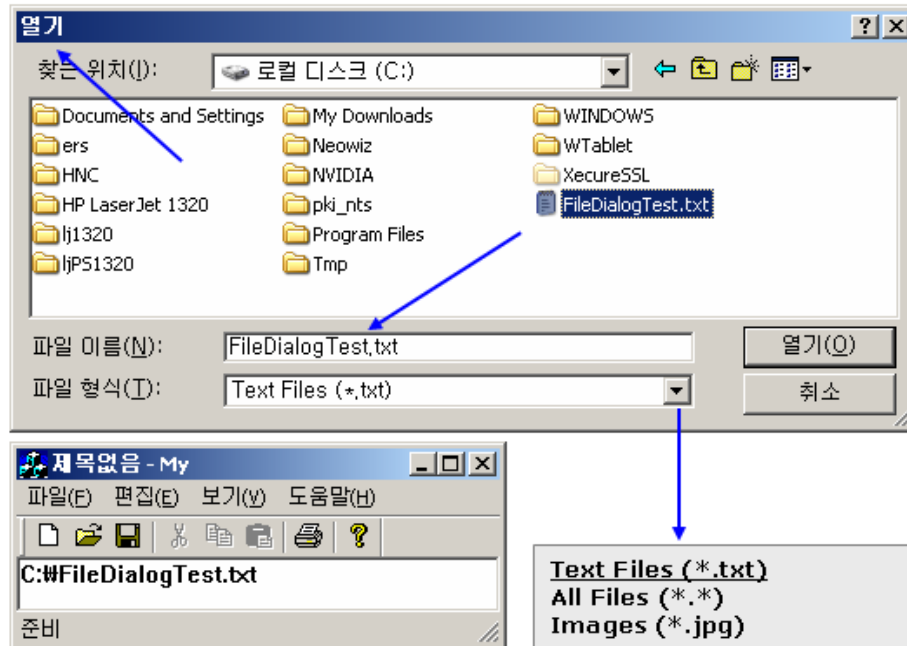
    //UpdateData( FALSE );
    CEdit* pEdit;
    pEdit = (CEdit*)GetDlgItem( IDC_EDIT_SHAPE ); //에디트 컨트롤 객체 획득
    pEdit->SetWindowText( m_strShape ); //동그라미 출력
}
```

공용 대화상자

윈도우는 사용 빈도가 높은 대화상자를 미리 작성
→ 다음과 같은 공용 대화상자 클래스를 제공

클래스	관련 기능제공 API함수	기능
CFileDialog	GetOpenFileName, GetSaveFileName	파일열기와 닫기 기능의 대화상자
CPrintDialog	PrintDlg	프린트와 프린트 설정 대화상자
CPageSetupDialog	PageSetupDlg	페이지 설정 대화상자
CFindReplaeDialog	FindText, ReplaceText	문자열 검색 및 변경 대화상자
CColorDialog	ChooseColor	컬러 설정 대화상자
CFontDialogot	ChooseFont	폰트 설정 대화상자

실습 11.2



- 파일 대화상자의 사용법을 알아보자.
- IDR_MAINFRAME 메뉴 자원의 ID_FILE_OPEN 핸들러에서 처리

```
void CMyView::OnFileOpen( )
{
    char szFilters[] = "Text Files (*.txt)|*.txt|
        All Files(*.*)|*.*|
        Images (*.jpg)|*.jpg|"; //2개씩 쌍으로 사용
    CFileDialog dlg(
        TRUE, //파일열기; FALSE=파일저장
        "txt", "*.txt", //기본 파일 확장자 설정
        OFN_FILEMUSTEXIST| //존재하는 파일만 선택 가능하게 함
        OFN_HIDEREADONLY, //읽기전용 체크표시 없앰
        szFilters ); //파일종류지정

    if( dlg.DoModal()==IDOK )
    {
        CString strPath;
        strPath = dlg.GetPathName( ); //선택한 파일의 경로 반환

        CClientDC dc( this );
        dc.TextOut( 0, 0, strPath );
    }
}
```