

객체 지향 프로그래밍 응용

Chap 5. 사용자 인터페이스 (#1)



2012.10.11.

오 병 우

컴퓨터공학과
금오공과대학교

Contents

- Menu
- ToolBar
- DialogBar
- StatusBar
- MessageBox
- Mouse Cursor
- Program Icon
- CFormView

Menu 개요

□ 응용 프로그램의 처리 제어

□ 종류

◆ Main Menu

– View 위에 위치한 메뉴 표시줄

◆ PopUp Menu: 평소에는 보이지 않음

– DropDown Menu: Main Menu를 누르면 나타남

– Context Menu: 마우스 오른쪽 버튼 클릭하면 나타남

◆ System Menu

– Window의 가장 왼쪽 위의 아이콘 클릭

◆ (Dialog Menu)

– Dialog도 Menu 가질 수 있음

일반적인 Menu 작성 및 사용 방법

■ 작성

◆ VC++의 리소스 편집기를 사용하여 작성

- ID는 주로 IDM_으로 시작
- String Table
- *.rc 파일에 저장
- Resource.h
- View|Resource Symbols

```

BOOL CEx6App::InitInstance()
{
    ...
    CSingleDocTemplate* pDocTemplate;
    pDocTemplate = new CSingleDocTemplate(
        IDR_MAINFRAME,
        RUNTIME_CLASS(CEx6Doc),
        RUNTIME_CLASS(CMainFrame),
        RUNTIME_CLASS(CEx6View));
    AddDocTemplate(pDocTemplate);

    ...
    return TRUE;
}
    
```

■ 사용

◆ ON_COMMAND, ON_COMMAND_RANGE

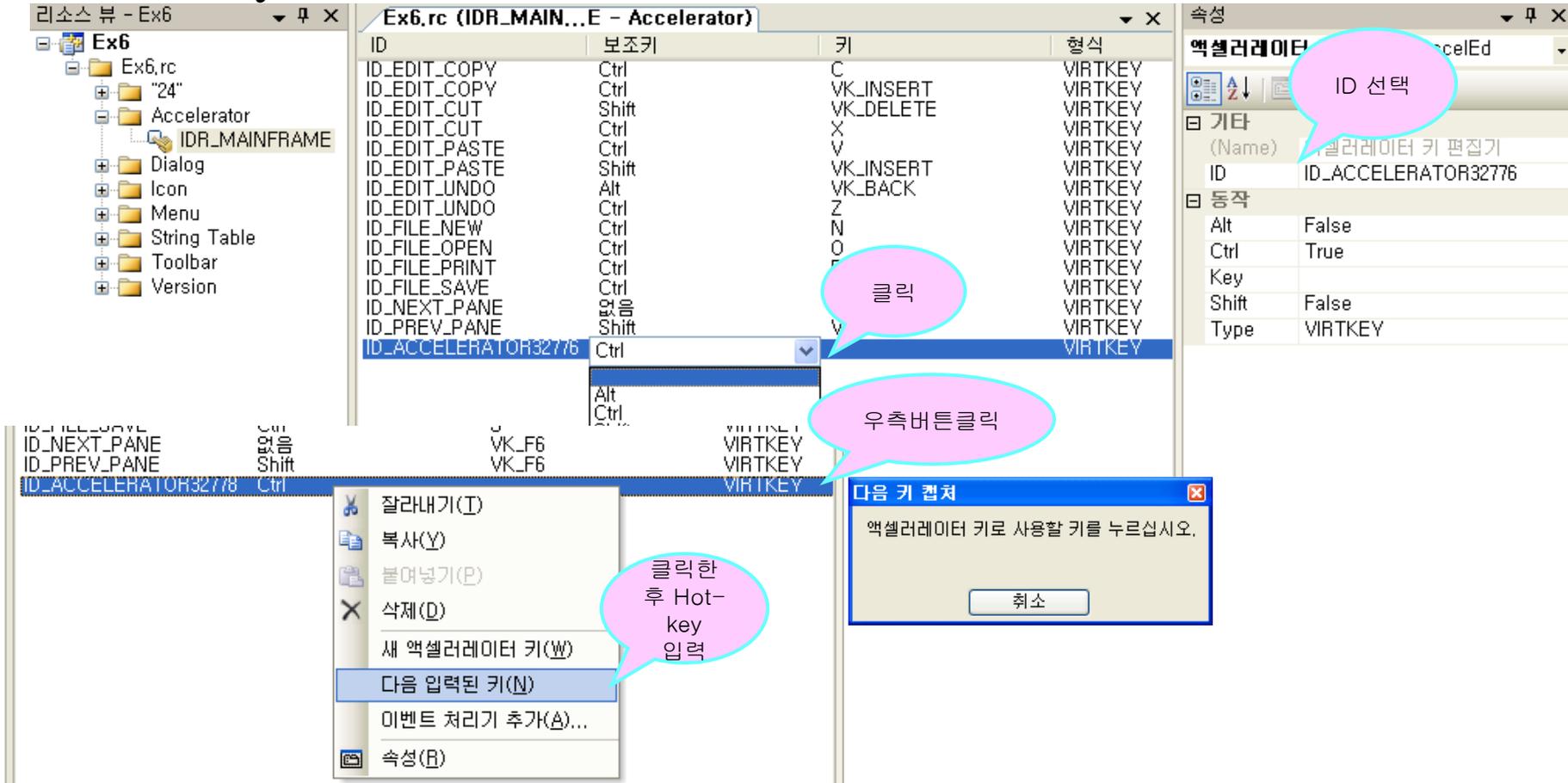
- WM_COMMAND 메시지 처리
- Menu 실행 처리

◆ ON_UPDATE_COMMAND_UI, ON_UPDATE_COMMAND_UI_RANGE

- Callback 함수 처리
- 메뉴 아이템이 화면에 나타나기 직전에 호출되어 어떻게 표시할지 결정
- 메뉴의 인터페이스 처리 (Enable(), SetCheck(), SetRadio(), SetText())
 - e.g., CCmdUI::Enable(FALSE)

Accelerator

Hot-Key 설정

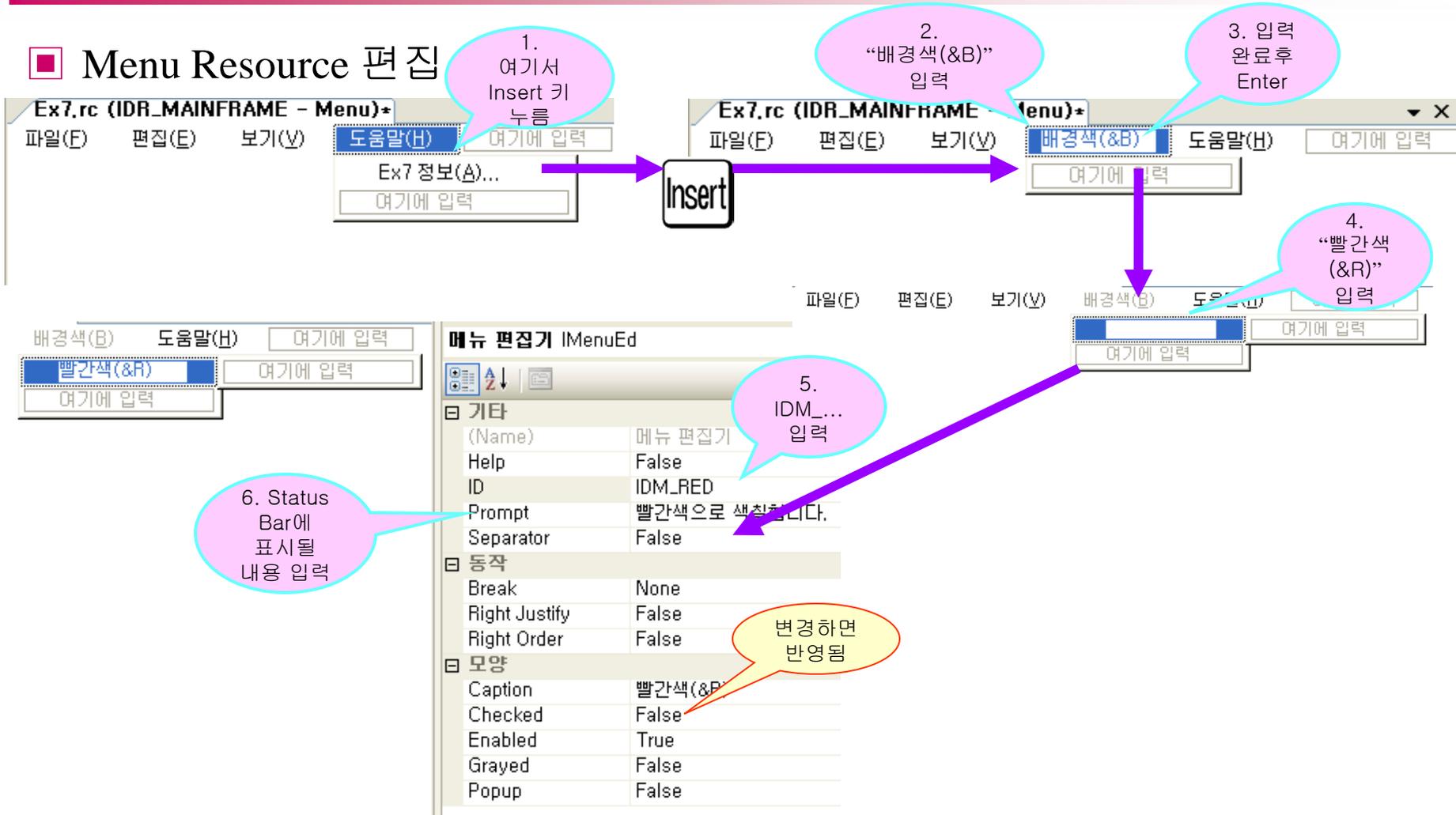


The screenshot shows the 'Accelerator' dialog box in Visual Studio. The main window displays a list of menu items with their corresponding accelerators. The item 'ID_ACCELERATOR32776' is selected, and its context menu is open, showing options like '잘라내기(I)', '복사(Y)', and '다음 입력된 키(N)'. A callout bubble points to the '다음 입력된 키(N)' option with the text '클릭한 후 Hot-key 입력'. Another callout bubble points to the 'Ctrl' modifier key in the dialog with the text '클릭'. A third callout bubble points to the '우측버튼클릭' (Right-click) option in the context menu. A fourth callout bubble points to the 'ID 선택' (ID selection) field in the '속성' (Properties) window on the right. The '속성' window shows the 'ID' property set to 'ID_ACCELERATOR32776' and the 'Type' property set to 'VIRTKEY'. A '다음 키 캡처' (Next Key Capture) dialog box is also visible, with the text '엑셀러레이터 키로 사용할 키를 누르십시오.' (Press the key to use as the accelerator key.) and a '취소' (Cancel) button.

ID	보조키	키	형식
ID_EDIT_COPY	Ctrl	C	VIRTKEY
ID_EDIT_COPY	Ctrl	VK_INSERT	VIRTKEY
ID_EDIT_CUT	Shift	VK_DELETE	VIRTKEY
ID_EDIT_CUT	Ctrl	X	VIRTKEY
ID_EDIT_PASTE	Ctrl	V	VIRTKEY
ID_EDIT_PASTE	Shift	VK_INSERT	VIRTKEY
ID_EDIT_UNDO	Alt	VK_BACK	VIRTKEY
ID_EDIT_UNDO	Ctrl	Z	VIRTKEY
ID_FILE_NEW	Ctrl	N	VIRTKEY
ID_FILE_OPEN	Ctrl	O	VIRTKEY
ID_FILE_PRINT	Ctrl	P	VIRTKEY
ID_FILE_SAVE	Ctrl	S	VIRTKEY
ID_NEXT_PANE	없음	O	VIRTKEY
ID_PREV_PANE	Shift	V	VIRTKEY
ID_ACCELERATOR32776	Ctrl		VIRTKEY

실습

Menu Resource 편집



1. 여기서 Insert 키 누름
여기에 입력

2. "배경색(&B)" 입력

3. 입력 완료 후 Enter

4. "빨간색 (&R)" 입력

5. IDM_... 입력

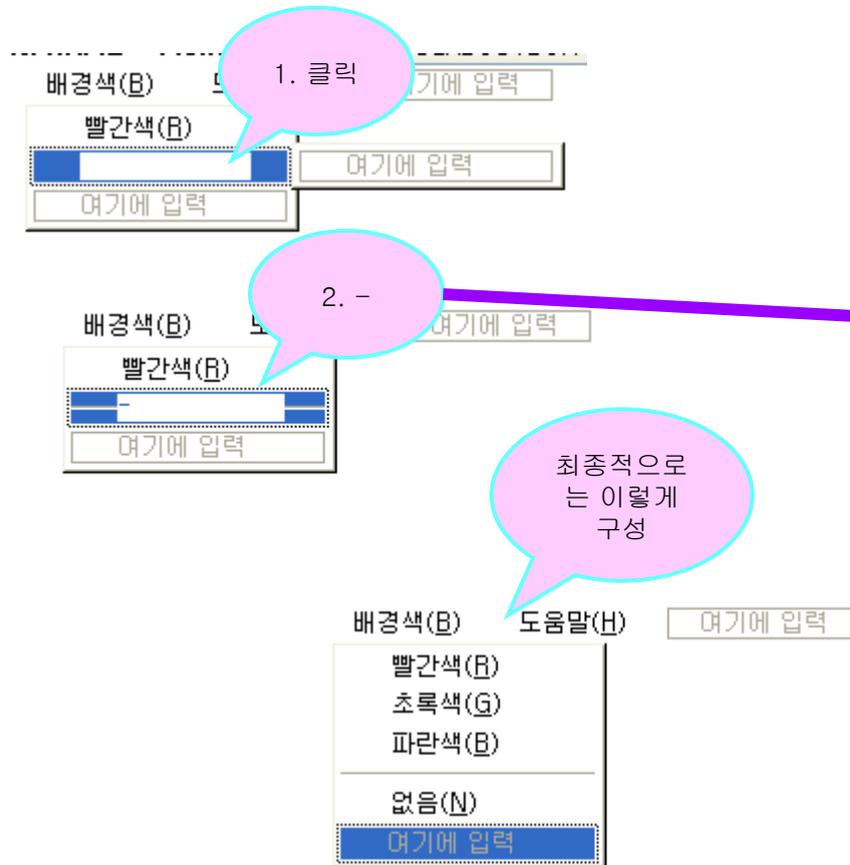
6. Status Bar에 표시될 내용 입력

변경하면 반영됨

메뉴 편집기 IMenuEd	
기타	
(Name)	메뉴 편집기
Help	False
ID	IDM_RED
Prompt	빨간색으로 색칠합니다.
Separator	False
동작	
Break	None
Right Justify	False
Right Order	False
모양	
Caption	빨간색(&R)
Checked	False
Enabled	True
Grayed	False
Popup	False

실습

Seperator 삽입



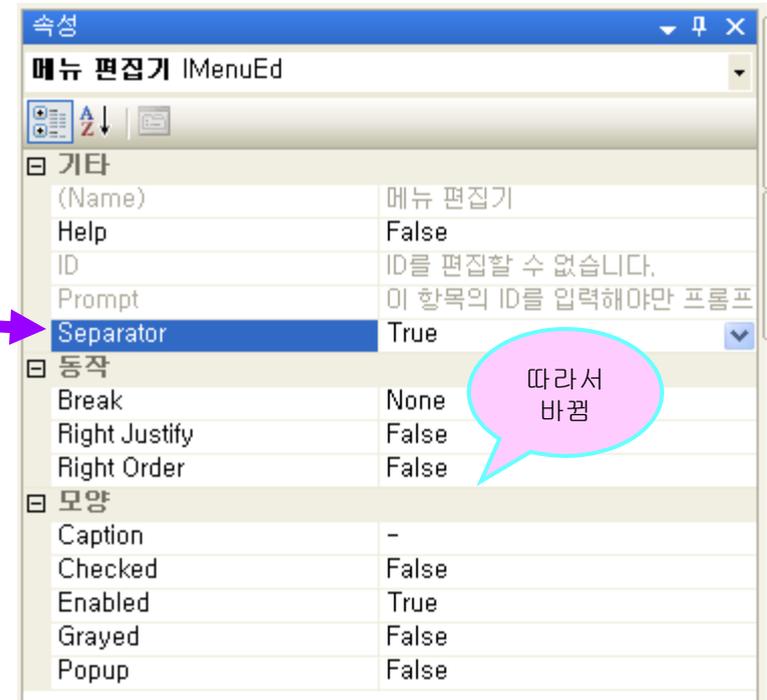
1. 클릭

2. -

최종적으로
는 이렇게
구성

배경색(B) 도움말(H) 여기에 입력

- 빨간색(R)
- 초록색(G)
- 파란색(B)
- 없음(N)
- 여기에 입력



속성	
메뉴 편집기 IMenuEd	
기타	
(Name)	메뉴 편집기
Help	False
ID	ID를 편집할 수 없습니다.
Prompt	이 항목의 ID를 입력해야만 프롬프트
Separator	True
동작	
Break	None
Right Justify	False
Right Order	False
모양	
Caption	-
Checked	False
Enabled	True
Grayed	False
Popup	False

따라서
바뀜

실습

Accelerator

리소스 뷰 - Ex7

Ex7.rc (IDR_MAIN...E - Accelerator)

ID	보조키	키	형식
ID_EDIT_COPY	Ctrl	C	VIRTKEY
ID_EDIT_COPY	Ctrl	VK_INSERT	VIRTKEY
ID_EDIT_CUT	Shift	VK_DELETE	VIRTKEY
ID_EDIT_CUT	Ctrl	X	VIRTKEY
ID_EDIT_PASTE	Ctrl	V	VIRTKEY
ID_EDIT_PASTE	Shift	VK_INSERT	VIRTKEY
ID_EDIT_UNDO	Alt	VK_BACK	VIRTKEY
ID_EDIT_UNDO	Ctrl	Z	VIRTKEY
ID_FILE_NEW	Ctrl	N	VIRTKEY
ID_FILE_OPEN	Ctrl	O	VIRTKEY
ID_FILE_PRINT	Ctrl	P	VIRTKEY
ID_FILE_SAVE	Ctrl	S	VIRTKEY
ID_NEXT_PANE	Ctrl	VK_F6	VIRTKEY
ID_PREV_PANE	Shift	VK_F6	VIRTKEY

속성

액셀러레이터 키 편집기 | AccelEd

기타

(Name) 액셀러레이터 키 편집기

ID IDM_NONE

동작

Alt	False	IDM_BLUE
Ctrl	True	IDM_GREEN
Key		IDM_NONE
Shift	False	IDM_RED
Type		VIRTKEY

다음 키 캡처

액셀러레이터 키로 사용할 키를 누르십시오

취소

3. 콤보박스의 버튼 클릭

4. ID 선택

5. 마우스 우측버튼 클릭

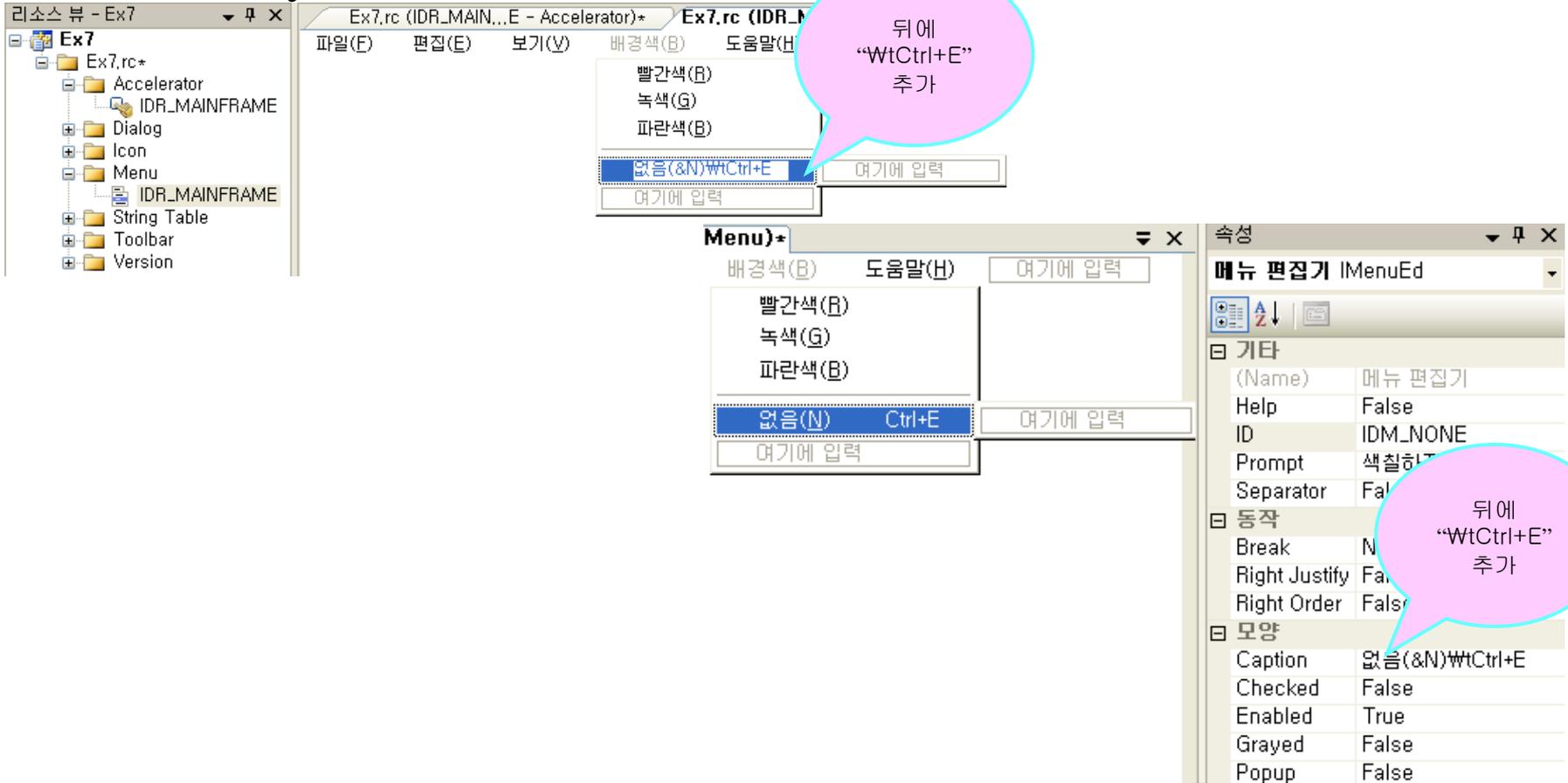
6. 클릭

ID	보조키	키	형식
ID_EDIT_PASTE	Shift	VK_INSERT	VIRTKEY
ID_EDIT_CUT	Ctrl	X	VIRTKEY
ID_EDIT_UNDO	Ctrl	Z	VIRTKEY
IDM_NONE	Ctrl	E	VIRTKEY

7. Ctrl을 누른 상태에서 e (^e)

실습

Hot-key를 메뉴에 반영



뒤에 "WtCtrl+E" 추가

뒤에 "WtCtrl+E" 추가

속성	값
메뉴 편집기 (Menu Editor)	IMenuEd
기타 (Other)	
(Name)	메뉴 편집기
Help	False
ID	IDM_NONE
Prompt	색칠하기
Separator	False
동작 (Action)	
Break	None
Right Justify	False
Right Order	False
모양 (Appearance)	
Caption	없음(&N)WtCtrl+E
Checked	False
Enabled	True
Grayed	False
Popup	False

실습

View 영역 색칠

Header File (Ex7View.h)

```
class CEx7View : public CView
{
protected:
    COLORREF m_Color;
```

Color Reference

변수 만들면 반드시 생성자에서 초기화!!!

Document Class에서 관리하는 변수는 OnNewDocument()에서 초기화

Source File (Ex7View.cpp)

```
// CEx7View 생성/소멸
CEx7View::CEx7View()
{
    // TODO: 여기에 생성 코드를 추가합니다.
    m_Color = RGB(0, 0, 0);
}

void CEx7View::OnDraw(CDC* pDC)
{
    CEx7Doc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    if (!pDoc)
        return;

    // TODO: 여기에 원시 데이터에

    CRect rect;
    GetClientRect(&rect);

    CBrush brush(m_Color);

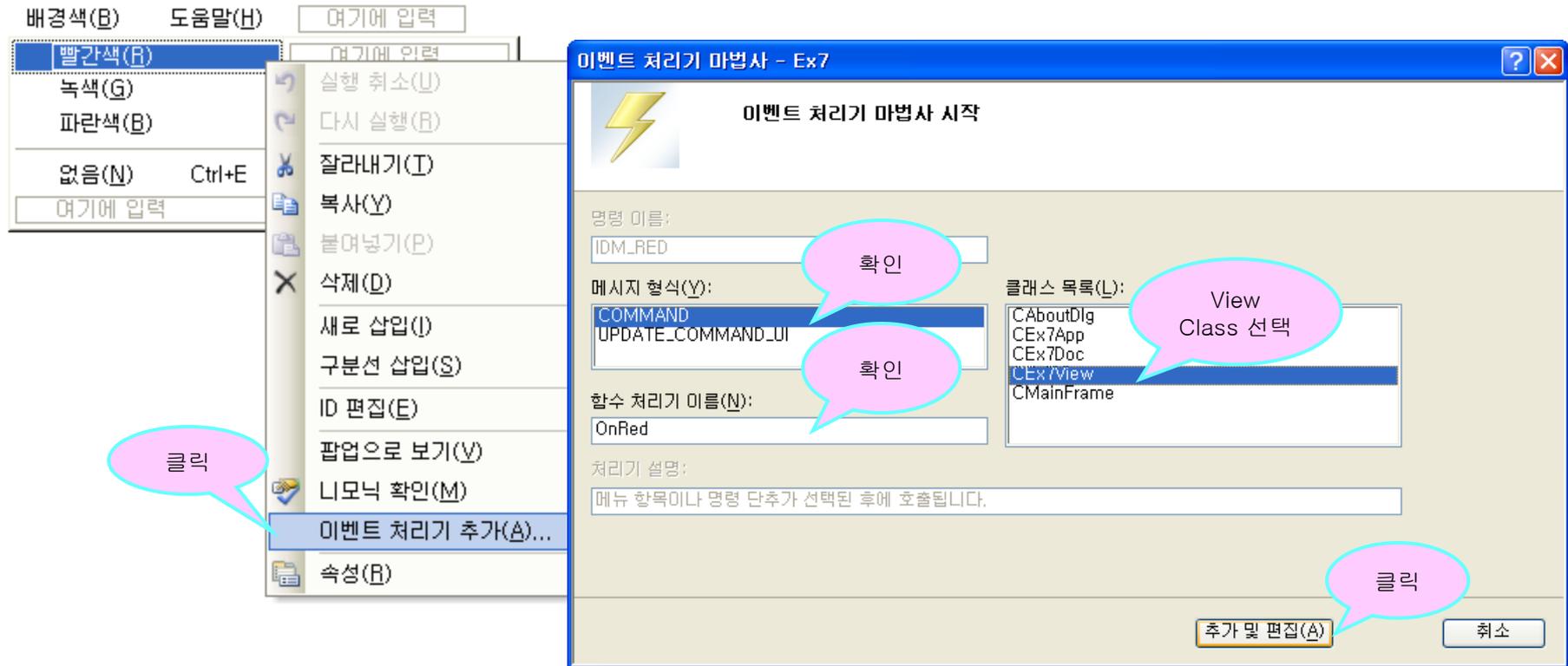
    pDC->FillRect(rect, &brush);
}
```

검은색

FillRect 함수는 SelectObject() 필요 없음 (그러므로, SelectObject(oldBrush) 또는 SaveDC()/RestoreDC()도 필요 없음)

실습

Message Handler 등록



배경색(B) 도움말(H) 여기에 입력

빨간색(R) 여기에 입력

- 녹색(G)
- 파란색(B)
- 없음(N) Ctrl+E
- 여기에 입력

실행 취소(U)

다시 실행(B)

잘라내기(I)

복사(Y)

붙여넣기(P)

삭제(D)

새로 삽입(I)

구분선 삽입(S)

ID 편집(E)

팝업으로 보기(V)

니모닉 확인(M)

이벤트 처리기 추가(A)...

속성(R)

클릭

이벤트 처리기 마법사 - Ex7

이벤트 처리기 마법사 시작

명령 이름: IDM_RED 확인

메시지 형식(Y): COMMAND 확인

함수 처리기 이름(N): OnRed

클래스 목록(L): View Class 선택

- CAboutDlg
- CEx7App
- CEx7Doc
- CEx7View**
- CMainFrame

처리기 설명: 메뉴 항목이나 명령 단추가 선택된 후에 호출됩니다.

추가 및 편집(A) 취소

클릭

실습

Message Handler Code 입력

```

// CEx7View 메시지 처리기
void CEx7View::OnRed()
{
    // TODO: 여기에 명령 처리기 코드를 추가합니다.
    m_Color = RGB(255, 0, 0);
}
    
```

◆ F5 눌러서 실행

◆ 색이 바뀌지 않음 (Resize하면 바뀜)

Invalidate()를 해주어야 다시 그림

```

// CEx7View 메시지 처리기
void CEx7View::OnRed()
{
    // TODO: 여기에 명령 처리기 코드를 추가합니다.
    m_Color = RGB(255, 0, 0);
    Invalidate();
}
    
```

◆ F5 눌러서 실행

실습

직접 코딩

◆ CEx7View.h의 Class Declaration에 함수 원형 입력

- Return Type: void, 매개변수 없음

```

// 생성된 메시지 맵 함수
protected:
    DECLARE_MESSAGE_MAP()
public:
    afx_msg void OnRed();
    afx_msg void OnGreen();
    afx_msg void OnBlue();
};

BEGIN_MESSAGE_MAP(CEx7View, CView)
    // 표준 인쇄 명령입니다.
    ON_COMMAND(ID_FILE_PRINT, &CView::OnFilePrint)
    ON_COMMAND(ID_FILE_PRINT_DIRECT, &CView::OnFilePrint)
    ON_COMMAND(ID_FILE_PRINT_PREVIEW, &CView::OnFilePrintPreview)
    ON_COMMAND(IDM_RED, &CEx7View::OnRed)
    ON_COMMAND(IDM_GREEN, &CEx7View::OnGreen)
    ON_COMMAND(IDM_BLUE, &CEx7View::OnBlue)
END_MESSAGE_MAP()

void CEx7View::OnRed()
{
    // TODO: 여기에 명령 처리기 코드를 추가합니다.
    m_Color = RGB(255, 0, 0);
    Invalidate();
}

void CEx7View::OnGreen()
{
    // TODO: 여기에 명령 처리기 코드를 추가합니다.
    m_Color = RGB(0, 255, 0);
    Invalidate();
}

void CEx7View::OnBlue()
{
    // TODO: 여기에 명령 처리기 코드를 추가합니다.
    m_Color = RGB(0, 0, 255);
    Invalidate();
}

```

◆ CEx7View.cpp 파일

- 맨 위의 Message Map 부분에 추가
- 맨 아래의 OnRed() Copy&Paste 후 수정

◆ F5 눌러서 실행

- 가끔 안되면 Build|Rebuild All 수행

실습

Menu Item의 속성 변경 (화면에 표시될 때 호출됨)

IDM_NONE의 비활성화 (F5 눌러서 확인)



이벤트 처리기 마법사 - Ex7

이벤트 처리기 마법사 시작

명령 이름: IDM_NONE

메시지 형식(Y): UPDATE_COMMAND_UI

함수 처리기 이름(N): OnUpdateNone

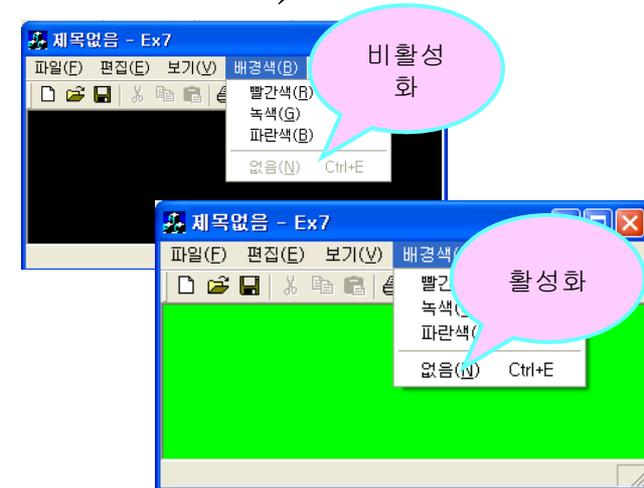
클래스 목록(L): CAboutDlg, CEx7App, CEx7Doc, CEx7View, CMainFrame

UPDATE_COMMAND_UI 클릭

확인

확인

추가 및 편집(A)



제목없음 - Ex7

비활성화

제목없음 - Ex7

활성화

```
void CEx7View::OnUpdateNone(CCmdUI* pCmdUI)
{
    // TODO: 여기에 명령 업데이트 로직을 추가합니다.
    /**
    if(m_Color == RGB(0, 0, 0))
        pCmdUI->Enable(FALSE);
    else
        pCmdUI->Enable(TRUE);
    ***/
    pCmdUI->Enable(m_Color !=RGB(0, 0, 0));
}

```

If 문 처리해도 되고... 로직을 추가합니다.

이렇게 간단히도 됨

실습

IDM_NONE의 Message Handler

- ◆ 앞 페이지의 UPDATE_COMMAND_UI는 메뉴를 화면에 표시할 때만 처리됨
- ◆ Menu가 선택되었을 때의 동작은 COMMAND 처리 해주어야 함



이벤트 처리기 마법사 - Ex7

이벤트 처리기 마법사 시작

명령 이름: IDM_NONE

메시지 형식(Y): COMMAND
UPDATE_COMMAND_UI

함수 처리기 이름(N): OnNone

처리 설명: 메뉴 항목이나 명령 단추가 선택된 후에 호출됩니다.

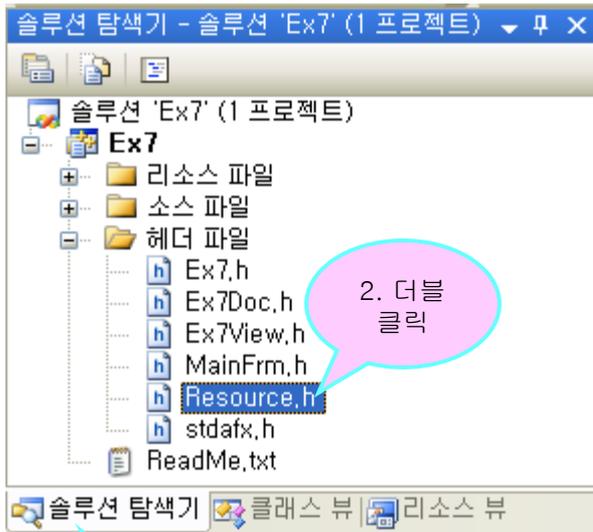
클래스 목록(L): CAboutDlg, CEx7App, CEx7Doc, CEx7View, CMainFrame

void CEx7View::OnNone()
{
// TODO: 여기에 명령 처리기 코드를 추가합니다.
m_Color = RGB(0, 0, 0);
Invalidate();
}

F5 눌러서 확인

실습

Resource.h



```

//{{NO_DEPENDENCIES}}
// Microsoft Visual C++ generated include file.
// Used by Ex7.rc
//
#define IDD_ABOUTBOX 100
#define IDP_OLE_INIT_FAILED 100
#define IDR_MAINFRAME 128
#define IDR_Ex7TYPE 129
#define IDM_RED 32771
#define IDM_GREEN 32772
#define IDM_BLUE 32773
#define IDM_NONE 32775

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE 130
#define _APS_NEXT_COMMAND_VALUE 32776
#define _APS_NEXT_CONTROL_VALUE 1000
#define _APS_NEXT_SYMED_VALUE 101
#endif
#endif
    
```

실습 (계속)

□ ID 값 변경

- ◆ COMMAND_RANGE 또는 UPDATE_COMMAND_UI_RANGE 매크로를 사용하려면 ID들이 연속된 값이어야 함 (순서 중요)

```

#define IDD_ABOUTBOX          100
#define IDR_MAINFRAME         128
#define IDR_EX7TYPE           129
#define IDM_RED                40000
#define IDM_GREEN              40001
#define IDM_BLUE               40002
#define IDM_NONE               40003

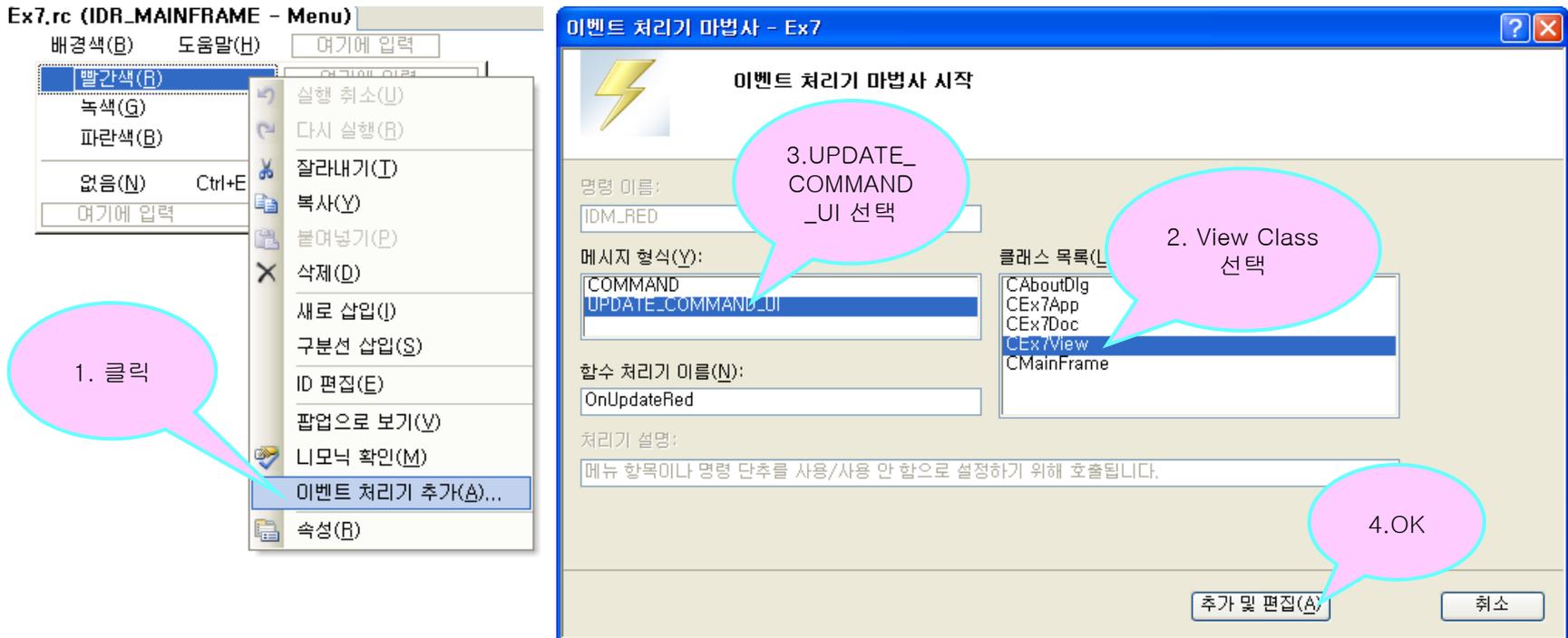
// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_3D_CONTROLS        1
#define _APS_NEXT_RESOURCE_VALUE 129
#define _APS_NEXT_COMMAND_VALUE 40004
#define _APS_NEXT_CONTROL_VALUE 1000
#define _APS_NEXT_SYMED_VALUE  101
#endif
#endif

```

실습

UPDATE_COMMAND_UI_RANGE는 Class Wizard에서 지원 안됨

◆ IDM_RED에 대한 UPDATE_COMMAND_UI를 Class Wizard를 통해 생성하고 변경하는 실습



1. 클릭

2. View Class 선택

3. UPDATE_COMMAND_UI 선택

4. OK

실습

- IDM_RED를 위한 Callback 함수를 IDM_GREEN 및 IDM_BLUE 도 처리할 수 있도록 변경

	변경전	변경후
Header File	<pre>afx_msg void OnUpdateRed(CCmdUI *pCmdUI);</pre>	<pre>// 생성된 메시지 맵 함수 protected: DECLARE_MESSAGE_MAP() public: afx_msg void OnRed(); afx_msg void OnGreen(); afx_msg void OnBlue(); afx_msg void OnUpdateNone(CCmdUI *pCmdUI); afx_msg void OnNone(); afx_msg void OnUpdateColor(CCmdUI *pCmdUI);</pre> <p>함수 이름 변경</p>
Source File	<pre>ON_UPDATE_COMMAND_UI(IDM_RED, &CEX7View::OnUpdateRed) void CEX7View::OnUpdateRed(CCmdUI *pCmdUI) { // TODO: 여기에 명령 업데이트 UI 처리기 코드를 추가합니다. }</pre>	<pre>BEGIN_MESSAGE_MAP(CEX7View, CView) // 표준 인쇄 명령입니다. ON_COMMAND(ID_FILE_PRINT, &CView::OnFilePrint) ON_COMMAND(ID_FILE_PRINT_DIRECT, &CView::OnFilePrint) ON_COMMAND(ID_FILE_PRINT_PREVIEW, &CView::OnFilePrintPreview) ON_COMMAND(IDM_RED, &CEX7View::OnUpdateRed) ON_COMMAND(IDM_GREEN, &CEX7View::OnUpdateColor) ON_COMMAND(IDM_BLUE, &CEX7View::OnUpdateColor) ON_UPDATE_COMMAND_UI(IDM_NONE, &CEX7View::OnUpdateNone) ON_UPDATE_COMMAND_UI(IDM_NONE, &CEX7View::OnUpdateNone) ON_UPDATE_COMMAND_UI_RANGE(IDM_RED, IDM_BLUE, &CEX7View::OnUpdateColor) void CEX7View::OnUpdateColor(CCmdUI *pCmdUI) { // TODO: 여기에 명령 업데이트 UI 처리기 코드를 추가합니다. }</pre> <p>IDM_RED에서 IDM_BLUE까지</p>

실습

UPDATE_COMMAND_UI_RANGE를 위한 Callback 함수

◆ 메뉴를 표시하기 위해 3번 호출됨

- IDM_RED, IDM_GREEN, IDM_BLUE 각각을 표시하기 위해 호출됨
- 3개를 한꺼번에 처리하지 않는다는 것에 주의할 것

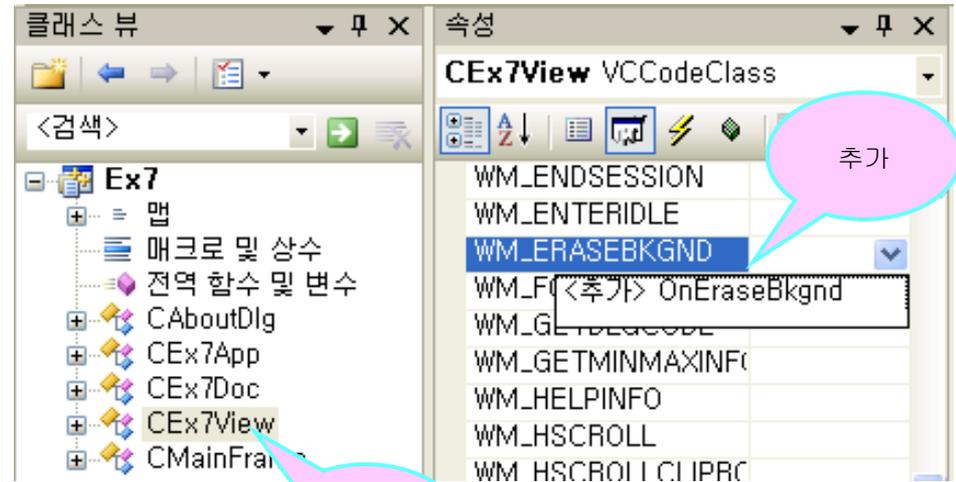
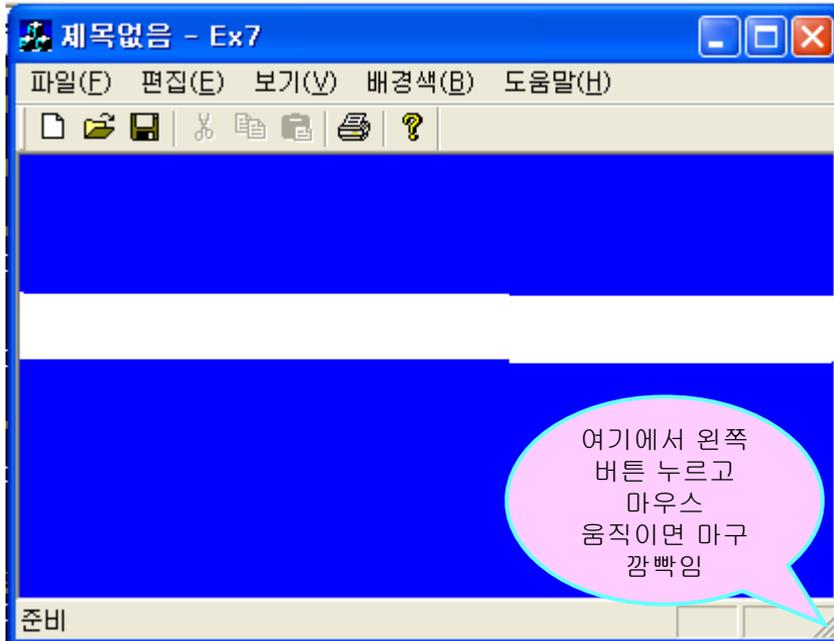
```
//같은 함수가 3번 호출됨
void CEx7View::OnUpdateColor(CCmdUI *pCmdUI)
{
    // TODO: 여기에 명령 업데이트 UI 처리기 코드를 추가합니다.
    if((m_Color==RGB(255, 0, 0) && pCmdUI->m_nID == IDM_RED) ||
        (m_Color==RGB(0, 255, 0) && pCmdUI->m_nID == IDM_GREEN) ||
        (m_Color == RGB(0, 0, 255) && pCmdUI->m_nID == IDM_BLUE))
        pCmdUI->SetRadio(TRUE);
    else
        pCmdUI->SetRadio(FALSE);
}
```

◆ F5 눌러서 실행 확인

화면의 깜빡임 제거

화면 전체를 다시 그릴 때 유용함

WM_ERASEBKGD



OnEraseBkgnd가 먼저 실행됨

- OnDraw와 OnEraseBkgnd에 Break Point (F9) 잡고 실행 (F5)

```

BOOL CEx7View::OnEraseBkgnd(CDC* pDC)
{
    // TODO: 여기에 메시지 처리기 코드를 입력하십시오.
    //return CView::OnEraseBkgnd(pDC);
    return TRUE;
}
    
```

아무것도 하지 않도록 주석 처리

기본값을 호출합니다.

ToolBar 개요

- 메뉴의 기능 중 자주 사용되는 것을 버튼 형태로 만든 것
 - ◆ 모양은 다르지만 ID를 Menu와 같게 지정하여 동일한 이벤트 발생
 - ◆ CFrameWnd의 파생클래스(AppWizard는 CMainFrame이라는 이름으로 만들어 줌)에서 초기화
 - OnCreate()

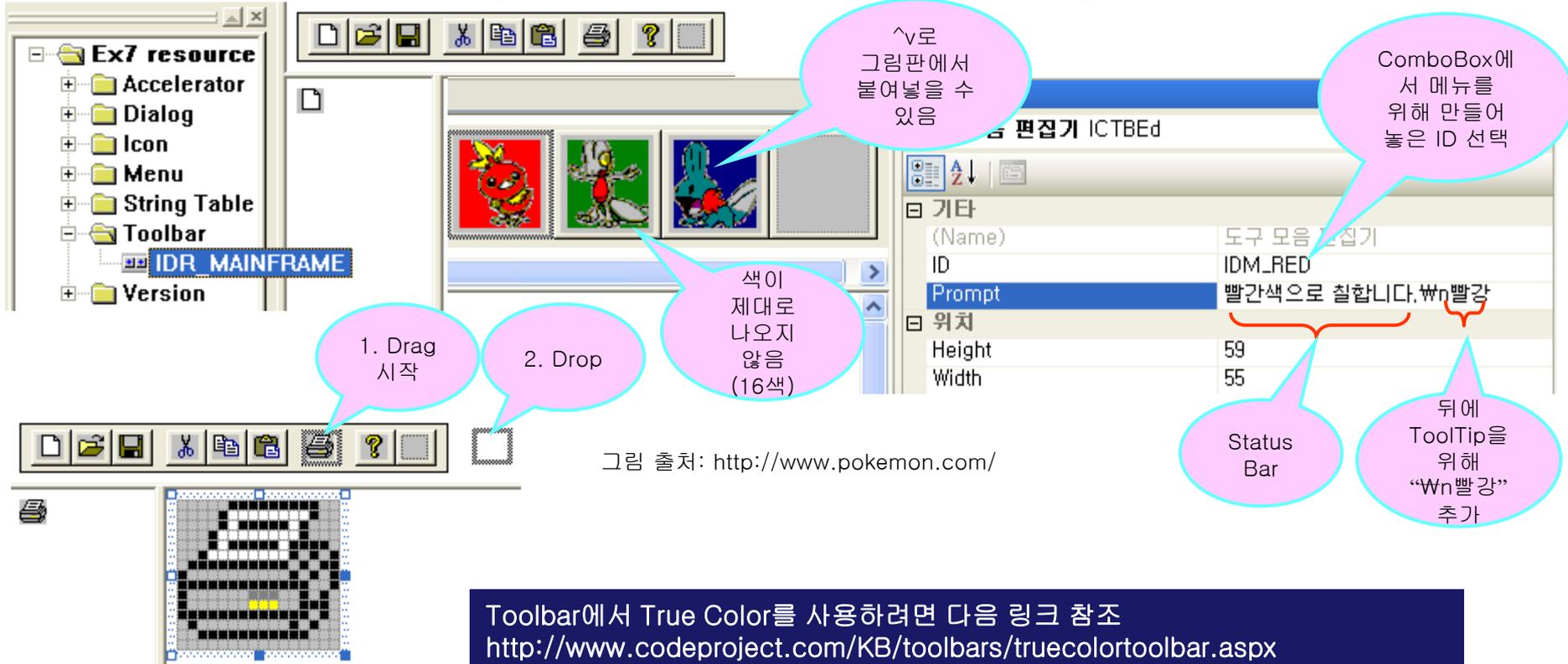
- 작성
 - ◆ VC++의 리소스 편집기를 사용하여 작성
 - Tool Tip 문자열은 Prompt에서 “\n”뒤에 지정
 - ◆ CreateEx() 또는 SetBarStyle()에서 ToolBar의 스타일 지정
 - CBRS_TOOLTIPS (툴팁), CBRS_FLYBY(상태표시줄에 문자열 출력), etc.
 - ◆ EnableDocking()으로 ToolBar를 Frame Window에 붙이는 Docking 지정
 - CBRS_ALIGN_TOP, CBRS_ALIGN_ANY, etc.

- 사용
 - ◆ ID를 Menu와 일치시키기만 하면 자동으로 메뉴에서 연결한 메시지 핸들러 함수 호출됨

ToolBar

ToolBar 편집

◆ 삭제할 때는 Drag 상태에서 도구모음 밖에 Drop



1. Drag 시작

2. Drop

색이 제대로 나오지 않음 (16색)

^v로 그림판에서 붙여넣을 수 있음

ComboBox에서 메뉴를 위해 만들어 놓은 ID 선택

기타	
(Name)	도구 모음 편집기
ID	IDM_RED
Prompt	빨간색으로 칠합니다. Wn 빨강
위치	
Height	59
Width	55

Status Bar

뒤에 ToolTip을 위해 "Wn빨강" 추가

그림 출처: <http://www.pokemon.com/>

ToolBar에서 True Color를 사용하려면 다음 링크 참조
<http://www.codeproject.com/KB/toolbars/truecolortoolbar.aspx>

ToolBar에 Text 넣기

□ ToolBar에 Text 넣기



```
int CMainFrame::OnCreate(LPCREATESTRUCT lpCreateStruct)
{
    if (CFrameWnd::OnCreate(lpCreateStruct) == -1)
        return -1;

    if (!m_wndToolBar.CreateEx(this, TBSTYLE_FLAT, WS_CHILD | WS_VISIBLE | CBRS_TOP
        | CBRS_GRIPPER | CBRS_TOOLTIPS | CBRS_FLYBY | CBRS_SIZE_DYNAMIC) ||
        !m_wndToolBar.LoadToolBar(IDR_MAINFRAME))
    {
        TRACE0("Failed to create toolbar\n");
        return -1;    // fail to create
    }

    if (!m_wndStatusBar.Create(this) ||
        !m_wndStatusBar.SetIndicators(indicators,
            sizeof(indicators)/sizeof(UINT)))
    {
        TRACE0("Failed to create status bar\n");
        return -1;    // fail to create
    }

    // TODO: Delete these three lines if you don't want the toolbar to
    // be dockable
    m_wndToolBar.EnableDocking(CBRS_ALIGN_ANY);
    EnableDocking(CBRS_ALIGN_ANY);
    DockControlBar(&m_wndToolBar);

    m_wndToolBar.SetButtonText(1, "빨간색칠");
    m_wndToolBar.SetButtonText(2, "초록색칠");
    m_wndToolBar.SetButtonText(3, "파랑색칠");

    m_wndToolBar.SetSizes(CSize(70, 80), CSize(55, 59));
}
```

CFrameWnd
의
OnCreate()

```
void CToolBar::SetSizes(SIZE sizeButton, SIZE sizeImage)
{
    ASSERT_VALID(this);

    // sizes must be non-zero and positive
    ASSERT(sizeButton.cx > 0 && sizeButton.cy > 0);
    ASSERT(sizeImage.cx > 0 && sizeImage.cy > 0);

    // button must be big enough to hold image
    // + 7 pixels on x
    // + 6 pixels on y
    ASSERT(sizeButton.cx >= sizeImage.cx + 7);
    ASSERT(sizeButton.cy >= sizeImage.cy + 6);
}
```

Bitmap 크기

버튼 크기
(Bitmap
크기보다
크게)

-Debug-

Summary

Relationship Diagram for Menu

