

공용 컨트롤 I

(4주차)

학습개요

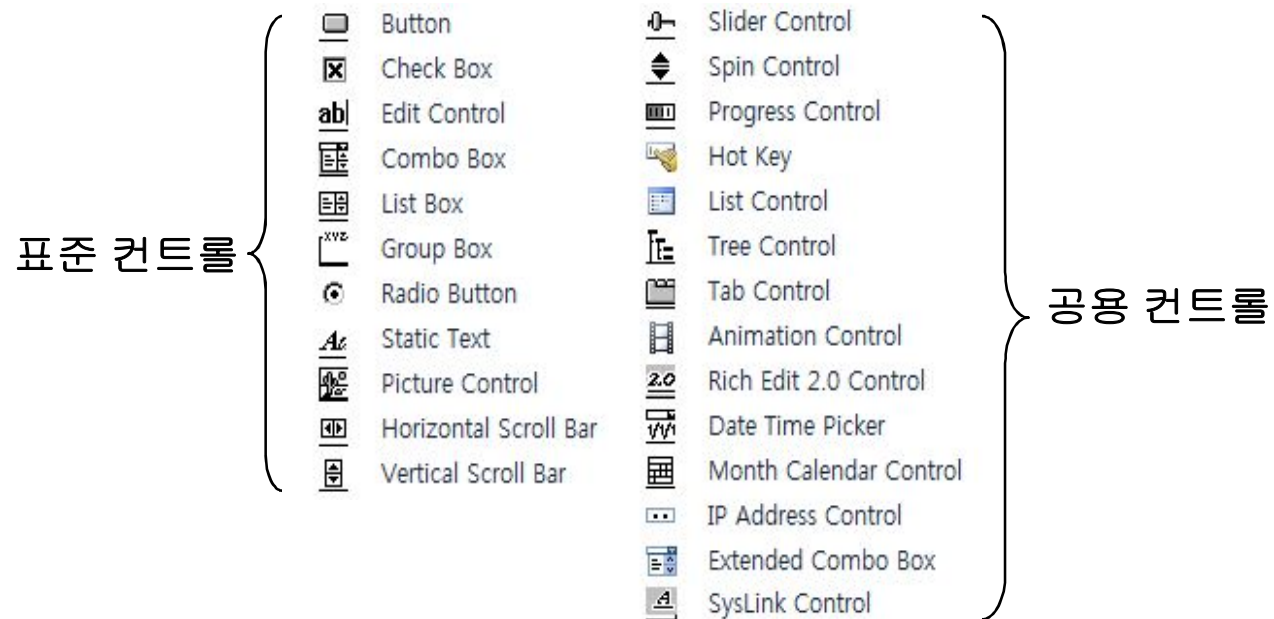
- 학습 목표

- 공용 컨트롤의 종류와 특징을 이해한다.
- 공용 컨트롤의 프로그래밍 방법을 익힌다.

- 학습 내용

- 공용 컨트롤 기초
- 이미지 리스트
- 실습

도구상자



비주얼 C++ 도구 상자를 살펴보면 위 그림에 표시되지 않은 컨트롤이 있다. 이들은 윈도우 운영체제가 기본으로 제공하지 않은 커스텀 컨트롤이다.

표준 컨트롤 vs 공용 컨트롤

- 표준 컨트롤은 USER32.DLL에 구현되어 있지만 공용 컨트롤은 COMCTL32.DLL에 구현되어 있다.
- COMCTL32.DLL의 버전에 따라 사용할 수 있는 공용 컨트롤의 종류와 기능에 제약이 있으므로 주의해야 한다.
- 사용 전에 반드시 ::InitCommonControls() 또는 ::InitCommonControlsEx() 함수를 호출해야 한다.
 - MFC를 사용하는 경우 이 과정은 자동으로 이루어진다.
- 통지 메시지의 종류와 형태가 다르다.
 - 일반적으로 표준 컨트롤은 WM_COMMAND 메시지를 사용하지만 공용 컨트롤은 WM_NOTIFY 메시지를 사용한다.

공용 컨트롤(Common Control) 종류 (1)

이름	MFC 클래스	기능
애니메이션(Animation)	CAnimateCtrl	AVI 비디오 클립 재생
확장 콤보 박스 (Extended ComboBox)	CComboBoxEx	텍스트와 이미지를 표시할 수 있는 콤보 박스
날짜 시간 선택기 (Date and Time Picker)	CDateTimeCtrl	날짜와 시간 선택
◆드래그 리스트 박스 (Drag List Box)	CDragListBox	항목을 드래그하여 옮길 수 있는 리스트 박스
◆평면 스크롤바(Flat Scroll Bar)	없음	형태 변경이 가능한 스크롤바 (윈도우 XP부터 사용 불가)
◆헤더(Header)	CHeaderCtrl	여러 열(Column)에 표시되는 텍스트의 폭 조절
핫키(Hotkey)	CHotKeyCtrl	핫키 입력과 생성
◆이미지 리스트(Image List)	CImageList	여러 개의 이미지 관리

공용 컨트롤(Common Control) 종류 (2)

이름	MFC 클래스	기능
IP 주소(IP address)	CIPAddressCtrl	IP 주소 입력
리스트(List or List-View)	CListCtrl	텍스트와 아이콘을 다양한 방법으로 표시
달력(Month Calendar)	CMonthCalCtrl	날짜 정보 표시와 선택
◆페이저(Pager)	CPagerCtrl	좁은 화면에 자식 윈도우를 스크롤하여 표시
프로그레스 (Progress or Progress Bar)	CProgressCtrl	작업 진행 상황 표시
◆속성 시트(Property Sheet)	CpropertySheet	항목 속성 보기와 편집
◆리바(Rebar)	CRebarCtrl	다른 컨트롤을 자식 윈도우로 포함하는 툴바
리치 에디트(Rich Edit)	CRichEditCtrl	다양한 기능을 제공하는 편집 컨트롤

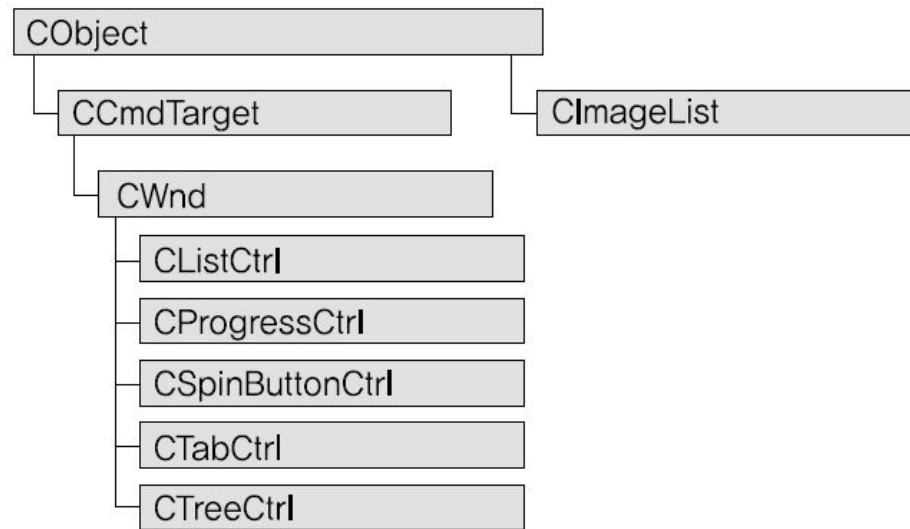
공용 컨트롤(Common Control) 종류 (3)

이름	MFC 클래스	기능
◆상태바(Status Bar)	CStatusBarCtrl	프로그램 상태 표시
시스링크(SysLink)	CLinkCtrl (MFC 7.0이상)	내장된 하이퍼링크 제공 (윈도우 XP부터 사용 가능)
탭(Tab)	CTabCtrl	동일한 화면 영역에 서로 다른 페이지를 표시 및 전환
◆태스크 대화 상자(Task Dialog)	CTaskDialog(MFC 10.0 이상)	풍부한 기능을 제공하는 일종의 메시지 상자 (윈도우 비스타부터 사용 가능)
◆툴바(Toolbar)	CToolBarCtrl	명령 메시지를 발생시키는 비트맵 버튼의 집합
◆툴팁(Tool Tip)	CToolTipCtrl	툴바 버튼 등의 컨트롤 위에 조그만 크기의 팝업 윈도우 표시

공용 컨트롤(Common Control) 종류 (4)

이름	MFC 클래스	기능
슬라이더 (Slider or Trackbar)	CSliderCtrl	눈금 표시를 가질 수 있는 일종의 스크롤 바
트리 (Tree or Tree-View)	CTreeCtrl	항목을 계층적으로 표시
스핀 버튼 (Spin Button or Up-Down)	CSpinButtonCtrl	두 개의 화살 표시로 값 증감

MFC 클래스 계층도



이미지 리스트(Image List)

- 크기가 동일한 이미지의 집합
- 각각의 이미지를 배열 항목처럼 취급하여 0부터 시작하는 인덱스로 참조
 - 예) 트리 컨트롤과 리스트 컨트롤에 사용되는 비트맵

이미지 리스트 생성 (1)

- 이미지 리스트 생성 방법

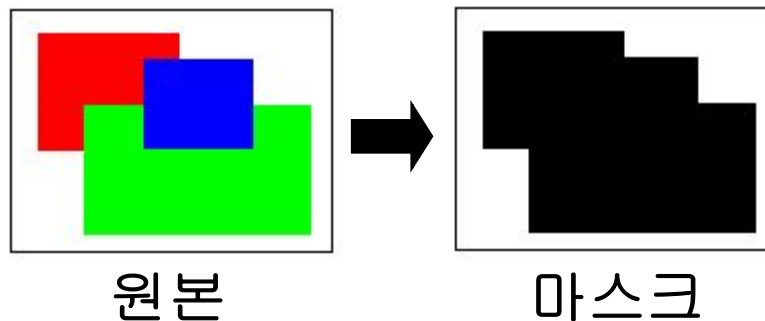
```
CImageList m_il;  
m_il.Create(...);
```

- 이미지 리스트 생성 함수

```
BOOL CImageList::Create(int cx, int cy, UINT nFlags, int nInitial, int nGrow);  
                        ①      ②      ③      ④      ⑤  
BOOL CImageList::Create(UINT nBitmapID, int cx, int nGrow, COLORREF crMask);  
                        ⑥      ①      ⑤      ⑦
```

이미지 리스트 생성 (2)

- cx, cy: 이미지의 폭과 높이(픽셀 단위)
- nFlags: 이미지 리스트의 타입을 지정하는 플래그 값들
- nInitial: 초기에 포함할 이미지 개수
- nGrow: 이미지 리스트가 가득 찬 상태에서 새 이미지를 추가할 때 내부의 여유 공간을 얼마나 키울지를 나타냄
- nBitmapID: 비트맵 리소스 ID
- crMask: 투명색으로 사용할 색상

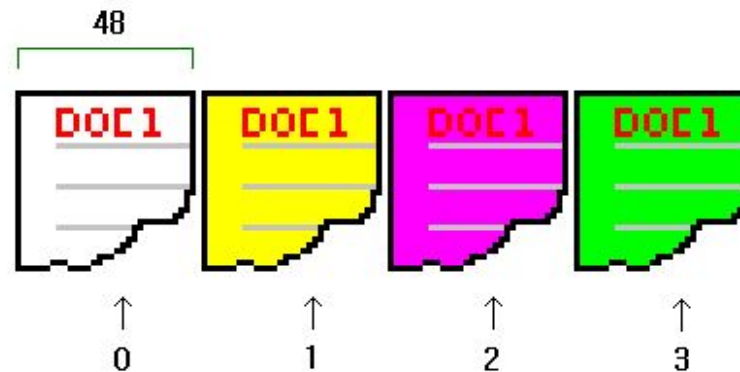


이미지 리스트 생성(3)

- 이미지 리스트 생성 예

```
CImageList m_il1, m_il2;  
①m_il1.Create(32, 32, ILC_COLOR4, 2, 1);  
②m_il2.Create(IDB_BITMAP1, 48, 1, RGB(255, 255, 255));
```

- IDB_BITMAP1



이미지 추가 (1)

- 이미지 추가 함수

```
int CImageList::Add(HICON hIcon);
```

①

```
int CImageList::Add(CBitmap* pbmImage, COLORREF crMask);
```

②

③

- hIcon: 아이콘 리소스를 가리키는 핸들값
- pbmImage: 비트맵 리소스를 담고 있는 CBitmap 객체
- crMask: 투명색으로 사용할 색상

이미지 추가 (2)

- 이미지 추가 예

```
// 두 개의 아이콘을 이미지 리스트에 추가한다.  
m_i11.Add(AfxGetApp()->LoadIcon(IDI_ICON1));  
m_i11.Add(AfxGetApp()->LoadIcon(IDI_ICON2));  
// 비트맵을 이미지 리스트에 추가한다.  
CBitmap bitmap;  
bitmap.LoadBitmap(IDB_BITMAP1);  
m_i12.Add(&bitmap, RGB(255, 255, 255));
```

화면 출력 (1)

- 화면 출력 함수

```
BOOL CImageList::Draw(CDC* pDC, int nImage, POINT pt, UINT nStyle);
```

① ② ③ ④

- pDC: 출력할 대상을 나타내는 디바이스 컨텍스트
- nImage: 출력할 이미지의 인덱스
- pt: 출력 좌표
- nStyle: 그리기 스타일

화면 출력 (2)

- 그리기 스타일

값	의미	
ILD_BLEND25 또는 ILD_FOCUS	시스템 하이라이트(Highlight) 색상을 25% 섞어서 이미지를 출력한다. (이미지 리스트에 마스크가 있을 때만 의미 있다.)	
ILD_BLEND50 또는 ILD_BLEND	시스템 하이라이트(Highlight) 색상을 50% 섞어서 이미지를 출력한다. (이미지 리스트에 마스크가 있을 때만 의미 있다.)	
ILD_MASK	마스크를 출력한다.	
ILD_NORMAL	이미지 리스트의 배경색을 이용하여 이미지를 출력한다. 배경색이 CLR_NONE인 경우에는 마스크를 이용하여 이미지의 배경을 투명하게 출력한다.	
ILD_TRANSPARENT	이미지 리스트의 배경색과 무관하게, 마스크를 이용하여 이미지의 배경을 투명하게 출력한다.	

화면 출력 (3)

- 화면 출력 예

```
void CImageView::OnDraw(CDC* pDC)
{
    // 배경색을 CLR_NONE으로 설정
    m_il.SetBkColor(CLR_NONE);

    // 배경을 투명하게 출력
    m_il.Draw(pDC, 0, CPoint(100, 200), ILD_NORMAL);

    // 하이라이트 색상을 25% 섞음
    m_il.Draw(pDC, 0, CPoint(150, 200), ILD_BLEND25);

    // 하이라이트 색상을 50% 섞음
    m_il.Draw(pDC, 0, CPoint(200, 200), ILD_BLEND50);

    // 마스크만 출력
    m_il.Draw(pDC, 0, CPoint(250, 200), ILD_MASK);
}
```

학습정리

- 공용 컨트롤은 COMCTL32.DLL에 구현되어 있으며, COMCTL32.DLL의 버전에 따라 사용할 수 있는 공용 컨트롤의 종류와 기능에 제약이 있다.
- 표준 컨트롤은 WM_COMMAND 메시지를 주로 사용하지만, 공용 컨트롤은 WM_NOTIFY 메시지를 주로 사용한다.
- 이미지 리스트 컨트롤은 크기가 동일한 이미지의 집합으로 각각의 이미지를 배열 항목처럼 취급하여 0부터 시작하는 인덱스로 참조할 수 있다.
- 이미지 리스트 컨트롤은 CImageList 클래스의 객체를 생성해 사용한다.