



Window API Programming- 입문 3

afewhee@gmail.com



- 1. 메뉴
- 2. 표준 Control
- 3. Dialog





- Resource를 이용해서 추가
- WNDCLASS 구조체 멤버 변수에서 Resource ID로 등록
 - ◆ Ex) MAKEINTRESOURCE(MENU_ID)
- Message Receive
 - ◆ WM_COMMAND에서 처리
 - ◆ 메뉴 ID: LOWORD(wParam)

Ex)

```
case WM_COMMAND:
{
    switch(LOWORD(wParam))
    {
        case ID_FILE_MENU1:
            ...
            break;
    }
}
...
```





- 표준 Control
 - ◆ Window에서 미리 정해 놓은 윈도우
 - ◆ class name으로 생성
 - ◆ Ex) button, edit, static, listbox, combobox, scrollbar...
- 생성: CreateWindow()
- 해제: DestroyWindow()
- 메시지
 - ◆ WM_COMMAND에서 처리
 - ◆ 컨트롤 ID: LOWORD(wParam)
 - ◆ 컨트롤 이벤트: HIWORD(wParam)





2. 표준 Control - button

- 버튼, 체크 박스, 라디오 버튼 등은 클래스 이름을 "button" 으로 생성.
- 옵션에서 버튼, 체크 박스, 라디오 버튼 등을 지정
- SendMessage() 함수를 이용 상태 변경

```
Ex) m_hWndBtn1 = CreateWindow("button", "Title"  
    , WS_CHILD | WS_VISIBLE | BS_PUSHBUTTON  
    , 20,20,100,25  
    , m_hWnd  
    , (HMENU) (ID_BUTTON1)  
    , m_hInst, NULL);
```

...

```
case WM_COMMAND:  
{  
    switch(LOWORD(wParam))  
    {  
        case ID_BUTTON1:  
        {  
            ...  
        }  
    }  
}
```





2. 표준 Control - combobox

- 클래스 이름을 "combobox" 으로 생성.
- SendMessage() 함수를 이용 값을 가져오거나 수정
- 이벤트: HIWORD(wParam)를 이용

```
Ex) m_hWndCombo = CreateWindow("combobox", NULL
                               WS_CHILD | WS_VISIBLE | CBS_DROPDOWN
                               , 20, 100, 100, 200
                               , m_hWnd
                               , (HMENU)ID_COMBO
                               , m_hInst, NULL);
```

```
...
```

```
// 리스트 추가하기
```

```
SendMessage(m_hWndCombo, CB_ADDSTRING, 0, (LPARAM)string);
```

```
...
```

```
case WM_COMMAND:
```

```
{
```

```
    switch(LOWORD(wParam))
```

```
    {
```

```
        case ID_COMBO:
```

```
        {
```

```
            switch (HIWORD(wParam))
```

```
            {
```

```
                case CBN_SELCHANGE: ...
```

```
                case CBN_EDITCHANGE: ...
```

```
            }
```

```
        }
```

```
}
```





2. 표준 Control - edit

- 클래스 이름을 "edit" 으로 생성.
- SendMessage() 함수를 이용 값을 가져오거나 수정.
- 텍스트의 경우 GetWindowText() / SetWindowText()
- 이벤트: HIWORD(wParam)를 이용해서 얻음

Ex)

```
m_hWndEdit=CreateWindow("edit", NULL
    , WS_CHILD | WS_VISIBLE | WS_BORDER | ES_AUTOHSCROLL
    , 10, 10, 200, 25
    , m_hWnd
    , (HMENU)ID_EDIT
    , m_hInst, NULL);
```

...

```
case WM_COMMAND:
{
    switch(LOWORD(wParam))
    {
        case ID_EDIT:
        {
            switch (HIWORD(wParam))
            {
                case EN_CHANGE: ...
```





2. 표준 Control - listbox

- 클래스 이름을 "listbox" 으로 생성.
- SendMessage() 함수를 이용 값을 가져오거나 수정. Combobox와 비슷
- 이벤트: HIWORD(wParam)를 이용해서 얻음

Ex)

```
m_hWndList=CreateWindow("listbox",NULL
                        , WS_CHILD | WS_VISIBLE | WS_BORDER | LBS_NOTIFY
                        , 300,10,100,200
                        , m_hWnd
                        , (HMENU)ID_LIST
                        ,m_hInst, NULL);
```

...

// 리스트 추가

```
SendMessage(m_hWndList,LB_ADDSTRING,0,(LPARAM)string);
```

```
case WM_COMMAND:
```

```
{
    switch(LOWORD(wParam))
    {
        case ID_EDIT:
        {
            switch (HIWORD(wParam))
            {
                case LBN_SELCHANGE : ...
```





2. 표준 Control - static

- 클래스 이름을 "static" 으로 생성. 단순 문자열 생성
- SendMessage() 함수를 이용 값을 가져오거나 수정.
- 텍스트의 경우 GetWindowText() / SetWindowText()

Ex)

```
HWND hStatic = CreateWindow("static", "Title", WS_CHILD | WS_VISIBLE  
    , 500,500,200,25  
    , m_hWnd  
    , (HMENU)-1  
    , m_hInst,NULL);
```

```
SetWindowText(hStatic, "Static is very Simple!!!");
```





- 클래스 이름을 "scrollbar" 으로 생성.
- {Set|Get}ScrollRange(), {Set|Get}ScrollPos() 함수를 이용
- Message: WM_HSCROLL / WM_VSCROLL 에서 처리
- IParam 값은 해당 스크롤 핸들
- 이벤트: LOWORD(wParam)를 이용해서 얻음

Ex)

```
m_hWndScrol[0]=CreateWindow("scrollbar", NULL  
                           , WS_CHILD | WS_VISIBLE | SBS_HORZ, ...);
```

```
m_hWndScrol[2]=CreateWindow("scrollbar", NULL  
                           , WS_CHILD | WS_VISIBLE | SBS_VERT, ...);
```

```
SetScrollRange(...);
```

```
SetScrollPos(...);
```

```
case WM_VSCROLL:
```

```
{
```

```
    case WM_HSCROLL:
```

```
    {
```

```
        hScrol = (HWND)IParam;
```

```
        ...
```

```
        switch (LOWORD(wParam))
```

```
        {
```

```
            case SB_LINELEFT:...
```

```
            case SB_LINERIGHT:...
```

```
            case SB_PAGELEFT:...
```

```
            case SB_PAGERIGHT:...
```

```
            case SB_THUMBTRACK:...
```





3. Dialog – 리소스를 간단한 다이얼로그 생성

```
BOOL CALLBACK MainDlgProc(...)  
{  
    switch( msg )  
    {  
        case WM_COMMAND:  
            ...  
            return FALSE;  
  
        case WM_CLOSE:  
        case WM_DESTROY:  
            PostQuitMessage( 0 );  
            return 0;  
    }  
  
    return FALSE;  
}  
  
INT WINAPI WinMain( HINSTANCE hInst, HINSTANCE, LPSTR, INT )  
{  
    return DialogBox(hInst, MAKEINTRESOURCE(IDD_DIALOG1), HWND_DESKTOP, MainDlgProc);  
  
    //return CreateDialog (hInst, MAKEINTRESOURCE(IDD_DIALOG1), HWND_DESKTOP, MainDlgProc);  
}
```





- **독점적 다이얼로그. 다이얼로그가 종료될 때까지 Blocking.**
- **DialogBox() 함수로 생성**
- **MessageBox 도 Modal Dialog**

Ex) 앞의 리소스를 이용한 윈도우 생성 예제



- 비 독점적 다이얼로그
- 생성: CreateDialog()
- 소멸: DestroyWindow()

Ex)

```
hDlg=CreateDialog(m_hInst,MAKEINTRESOURCE(IDD_DIALOG1)  
                , hParent,MlessDlgProc);
```

// 반드시 필요!!!

```
ShowWindow(hDlg, SW_SHOW);
```





- 다이얼로그다음 그림과 같이 3개의 화면을 구성하시오.
- 메인 윈도우가 움직여도 나머지 두 윈도우가 같이 움직이도록 구성하시오.
- 이미지를 편집해서 아이콘, 커서를 구현하시오.

