|  |  |
| --- | --- |
| sym01_s | **윈도우프로그래밍**  **LAB 06** |
| **분반** | **001** |
| **학번** | **2010044** |
| **이름** | **박진희** |

**# 실습 문제 1**

#1 tictactoe.cpp 프로그램 소스 및 설명

#include <afxwin.h>

#define IDB\_BUTTON0 100

#define IDB\_BUTTON1 101

#define IDB\_BUTTON2 102

#define IDB\_BUTTON3 103

#define IDB\_BUTTON4 104

#define IDB\_BUTTON5 105

#define IDB\_BUTTON6 106

#define IDB\_BUTTON7 107

#define IDB\_BUTTON8 108 //사용해야 할 버튼들의 ID를 정하는데 9개의 버튼이므로, 100~108의 값을 의미하도록 매크로를 정의한다.

enum { OFF, ON };

// Declare the application class

class CToggleApp : public CWinApp {

public:

virtual BOOL InitInstance();

};

// Create an instance of the application class

CToggleApp ToggleApp;

// Declare the main window class

class CToggleWindow : public CFrameWnd {

CButton\* button[9]; //버튼은 9개이므로 포인터 배열 크기가 8이면 딱 맞지만, 크기는 9로 설정한다.

int status[9]; //버튼 포인터배열 수에 맞춰 버튼이 눌렸는지, 아닌지의 상태를 나타내기 위한 배열 status 를 선언한다.

void init();

public:

CToggleWindow();

afx\_msg void HandleButton0();

afx\_msg void HandleButton1();

afx\_msg void HandleButton2();

afx\_msg void HandleButton3();

afx\_msg void HandleButton4();

afx\_msg void HandleButton5();

afx\_msg void HandleButton6();

afx\_msg void HandleButton7();

afx\_msg void HandleButton8(); //각 버튼별로 눌렀을 시 실행되는 함수를 정의한다.

void HandleButton(int \_iK);

DECLARE\_MESSAGE\_MAP();

};

// the InitInstance function is called once

// when the application first executes

BOOL CToggleApp::InitInstance()

{

m\_pMainWnd = new CToggleWindow();

m\_pMainWnd->ShowWindow(m\_nCmdShow);

m\_pMainWnd->UpdateWindow();

return TRUE;

}

// the message map

BEGIN\_MESSAGE\_MAP(CToggleWindow, CFrameWnd)

ON\_BN\_CLICKED(IDB\_BUTTON0, HandleButton0)

ON\_BN\_CLICKED(IDB\_BUTTON1, HandleButton1)

ON\_BN\_CLICKED(IDB\_BUTTON2, HandleButton2)

ON\_BN\_CLICKED(IDB\_BUTTON3, HandleButton3)

ON\_BN\_CLICKED(IDB\_BUTTON4, HandleButton4)

ON\_BN\_CLICKED(IDB\_BUTTON5, HandleButton5)

ON\_BN\_CLICKED(IDB\_BUTTON6, HandleButton6)

ON\_BN\_CLICKED(IDB\_BUTTON7, HandleButton7)

ON\_BN\_CLICKED(IDB\_BUTTON8, HandleButton8) //각 버튼이 클릭되었을때, 어떤 함수가 실행될 것인지 ON\_BN\_CLIKED 매크로 함수에 넣는다.

END\_MESSAGE\_MAP()

void CToggleWindow::init()

{ for(int i=0;i<9;i++) //처음 버튼의 상태를 각 -1로 설정하여 , 표시가 없다는 상태의 status로 이니셜라이즈해준다.

status[i] = -1;

}

// the contructor for the window class

CToggleWindow::CToggleWindow()

{

//Create the window itself

Create(NULL, \_T("Tic-Tac-Toe"),//상단바의 글씨를 Tic-Tac\_Toe로 설정한다.

WS\_OVERLAPPEDWINDOW,

CRect(0, 0, 300, 200));

init();

int k = 0;// button의 index를 설정하기 위한 정수형 변수 k. 초기값 0

for (int i = 0; i <3; i++)//CRect(a,b,c,d)라 한다면, a,c의 위치값을 설정하기 위한 loop

{

for (int j = 0; j < 3; j++)//b,d값을 설정하기 위한 innerloop

{

// Create button

button[k] = new CButton();

button[k]->Create(\_T(""), //button의 초기 텍스트 상태를 빈 상태로 만든다.

WS\_CHILD | WS\_VISIBLE |

BS\_PUSHBUTTON,

CRect(6 + i \* 30, 6 + j \* 30, 34 + i \* 30, 34 + j \* 30),

this,

100+k); //button의 id를 100,101..로 설정하였으므로 100+k로 입력하여 버튼을 연결한다.

k++;

}

}

}

void CToggleWindow::HandleButton(int \_iK)

{

if (status[\_iK] == ON) { //버튼이 X 상태라면

button[\_iK]->SetWindowText(\_T("O")); //O로 텍스트를 바꾼다

status[\_iK] = OFF;

MessageBeep(-1); //경고음 발생(버튼에 이미 텍스트가 넣어져있는 상태이기 때문

}

else if(status[\_iK]==OFF) //버튼이 O 상태라면

{

button[\_iK]->SetWindowText(\_T("X")); //X로 텍스트를 바꾼다.

status[\_iK] = ON;

MessageBeep(-1);

}

else //버튼에 아무 text도 입력되있지 않는 상태라면

{

button[\_iK]->SetWindowText(\_T("X"));

status[\_iK] = ON;

}

}

// message handler function for number buttons

void CToggleWindow::HandleButton0() //각 버튼별로 눌렀을 시 실행되는 함수

{

HandleButton(0);

}

void CToggleWindow::HandleButton1()

{

HandleButton(1);

}

void CToggleWindow::HandleButton2()

{

HandleButton(2);

}

void CToggleWindow::HandleButton3()

{

HandleButton(3);

}

void CToggleWindow::HandleButton4()

{

HandleButton(4);

}

void CToggleWindow::HandleButton5()

{

HandleButton(5);

}

void CToggleWindow::HandleButton6()

{

HandleButton(6);

}

void CToggleWindow::HandleButton7()

{

HandleButton(7);

}

void CToggleWindow::HandleButton8()

{

HandleButton(8);

}

#2 실행 화면

