

ใบงานที่ 3 การตรวจจับ message จากเมาส์และคีย์บอร์ด

วัตถุประสงค์

1. เพื่อให้รู้วิธีการเขียนโปรแกรมบน Windows ด้วยฟังก์ชัน Win32 API
2. เพื่อให้มีความคุ้นเคยกับการใช้เครื่องมือต่างๆ ในการสร้าง Application บน IDE

ลำดับการทดลอง

1. สร้าง Project ใหม่ เป็นชนิด Win32 Project แบบ empty project ตามใบงานที่ 1
2. เพิ่มไฟล์ .cpp ใหม่ เข้าไปใน Project
3. คัดลอกไฟล์ .cpp จากการทดลองที่ 2
4. แก้ไข code ในส่วนของการ switch (message) ในไฟล์ .cpp ให้เป็นไปดังต่อไปนี้

```
60     switch (message) {
61
62     case WM_PAINT:
63         hdc = BeginPaint (hwnd, &ps);
64         // Ellipse (hdc, 10, 10, 200, 100);
65         EndPaint (hwnd, &ps);
66         return 0;
67
68     case WM_LBUTTONDOWN:
69         char str[256];
70         POINT pt;    // point of mouse clicked, received via message
71         pt.x = LOWORD(lParam);
72         pt.y = HIWORD(lParam);
73         wsprintf(str,"WM_LBUTTONDOWN\nCo-ordinate are\n (%i, %i)",pt.x, pt.y);
74         MessageBox(hwnd, str, "Left Button Clicked", MB_OK);
75         return 0;
76
77     case WM_RBUTTONDOWN:
78         pt.x = LOWORD(lParam);
79         pt.y = HIWORD(lParam);
80         wsprintf(str,"WM_RBUTTONDOWN\nCo-ordinate are\n (%i, %i)",pt.x, pt.y);
81         MessageBox(hwnd, str, "Right Button Clicked", MB_OK);
82         return 0;
83
84     case WM_CHAR:
85         hdc = GetDC(hwnd);
86         TextOut(hdc, 1,1, " ",3);
87         wsprintf(str,"%c", (char) wParam);
88         TextOut(hdc, 1,1, str,strlen(str));
89         ReleaseDC(hwnd, hdc);
90         return 0;
91
92     case WM_DESTROY:
93         PostQuitMessage (0);
94         return 0;
95     }
```

5. กดปุ่ม F5 เพื่อดูผลการทำงานของโปรแกรม

Reference manual

โปรแกรมทั้งหมดในการทดลองนี้

```
#include <windows.h>
```

```
LONG WINAPI WndProc (HWND, UINT, WPARAM, LPARAM);
```

```
int WINAPI WinMain (HINSTANCE hInstance, HINSTANCE hPrevInstance,
LPSTR lpszCmdLine, int nCmdShow)
{
    WNDCLASS wc;
    HWND hwnd;
    MSG msg;
    /***** 1. Define Windows class *****/
    wc.style = 0; // Class style
    wc.lpfnWndProc = (WNDPROC) WndProc; // Window procedure address
    wc.cbClsExtra = 0; // Class extra bytes
    wc.cbWndExtra = 0; // Window extra bytes
    wc.hInstance = hInstance; // Instance handle
    wc.hIcon = LoadIcon (NULL, IDI_WINLOGO); // Icon handle
    wc.hCursor = LoadCursor (NULL, IDC_ARROW); // Cursor handle
    wc.hbrBackground = (HBRUSH) (COLOR_WINDOW + 1); // Background color
    wc.lpszMenuName = NULL; // Menu name
    wc.lpszClassName = "MyWndClass"; // WNDCLASS name

    /***** 2. Register the Windows class *****/

    RegisterClass (&wc);

    /***** 3. Create window *****/
    hwnd = CreateWindow (
        "MyWndClass", // WNDCLASS name
        "SDK Application", // Window title
        WS_OVERLAPPEDWINDOW, // Window style
        CW_USEDEFAULT, // Horizontal position
        CW_USEDEFAULT, // Vertical position
        CW_USEDEFAULT, // Initial width
        CW_USEDEFAULT, // Initial height
        HWND_DESKTOP, // Handle of parent window
        NULL, // Menu handle
        hInstance, // Application's instance handle
        NULL // Window-creation data
    );

    /***** 4. Display the window *****/
    ShowWindow (hwnd, nCmdShow);
    UpdateWindow (hwnd);

    /***** 5. Message loop *****/
    while (GetMessage (&msg, NULL, 0, 0)) {
        TranslateMessage (&msg);
        DispatchMessage (&msg);
    }
    return msg.wParam;
}

LRESULT CALLBACK WndProc (HWND hwnd, UINT message, WPARAM wParam,
LPARAM lParam)
{
    PAINTSTRUCT ps;
```

```

HDC hdc;

switch (message) {

case WM_PAINT:
    hdc = BeginPaint (hwnd, &ps);
//    Ellipse (hdc, 10, 10, 200, 100);
    EndPaint (hwnd, &ps);
    return 0;

case WM_LBUTTONDOWN:
    char str[256];
    POINT pt;    // point of mouse clicked, received via message
    pt.x = LOWORD(lParam);
    pt.y = HIWORD(lParam);
    wsprintf(str, "WM_LBUTTONDOWN\nCo-ordinate are\n (%i, %i)", pt.x,
        pt.y);
    MessageBox(hwnd, str, "Left Button Clicked", MB_OK);
    return 0;

case WM_RBUTTONDOWN:
    pt.x = LOWORD(lParam);
    pt.y = HIWORD(lParam);
    wsprintf(str, "WM_RBUTTONDOWN\nCo-ordinate are\n (%i, %i)", pt.x,
        pt.y);
    MessageBox(hwnd, str, "Right Button Clicked", MB_OK);
    return 0;

case WM_CHAR:
    hdc = GetDC(hwnd);
    TextOut(hdc, 1, 1, " ", 3);
    wsprintf(str, "%c", (char) wParam);
    TextOut(hdc, 1, 1, str, strlen(str));
    ReleaseDC(hwnd, hdc);
    return 0;

case WM_DESTROY:
    PostQuitMessage (0);
    return 0;
}
return DefWindowProc (hwnd, message, wParam, lParam);
}

```