

Ferestrele de dialog reprezinta niste ferestre Windows cu proprietati specifice pentru a permite introducerea datelor de la echipamentele de intrare iesire.

Crearea unei ferestre de dialog presupune mai intii crearea unei resurse " dialog". Un fisier de resurse pentru crearea unei ferestre dialog este prezentat in continuare:

```
IDD_ABOUT_DIALOG DISCARDABLE 0, 0, 239, 66
STYLE DS_MODALFRAME | WS_POPUP | WS_CAPTION | WS_SYSMENU
CAPTION "My About Box"
FONT 8, "MS Sans Serif"
BEGIN
    DEFPUSHBUTTON "&OK", IDOK, 174, 18, 50, 14
    PUSHBUTTON "&Cancel", IDCANCEL, 174, 35, 50, 14
    GROUPBOX "About this program...", IDC_STATIC, 7, 7, 225, 52
    CTEXT "An example program ",
        IDC_STATIC, 16, 18, 144, 33
END
```

Al doilea pas este acela de a crea o procedura Dialog care sa proceseze mesajele . Aceasta procedura este similara cu procedura principala a unui program Windows (dar nu si identica!)

```
BOOL CALLBACK AboutDlgProc(HWND hwnd, UINT Message, WPARAM wParam,
LPARAM lParam)
{
    switch(Message)
    {
        case WM_INITDIALOG:

            return TRUE;
        case WM_COMMAND:
            switch(LOWORD(wParam))
            {
                case IDOK:
                    EndDialog(hwnd, IDOK);
                    break;
                case IDCANCEL:
                    EndDialog(hwnd, IDCANCEL);
                    break;
            }
            break;
        default:
            return FALSE;
    }
    return TRUE;
}
```

Exista citeva diferente importante intre procedura Dialog si procedura principala:

- nu trebuie apelata procedura DefWindowProc() pentru mesajele care nu sint de interes pentru programul de aplicatie. Aceste mesaje vor fi tratate in mod automat.

- nu trebuie apalata procedura `DestroyWindow()` pentru a inchide fereastra de dialog; pentru acest lucru se va apela `EndDialog()`.

- se va trata mesajul `WM_INITDIALOG` in locul mesajului `WM_CREATE`

Crearea unei fereste dialog este ilustrata in continuare:

```
case WM_COMMAND:
    switch(LOWORD(wParam))
    {
        case ID_HELP_ABOUT:
        {
            int ret = DialogBox(GetModuleHandle(NULL),
                MAKEINTRESOURCE(IDD_ABOUT), hwnd, AboutDlgProc);
            if(ret == IDOK){
                MessageBox(hwnd, "Dialog exited with IDOK.", "Notice",
                    MB_OK | MB_ICONINFORMATION);
            }
            else if(ret == IDCANCEL){
                MessageBox(hwnd, "Dialog exited with IDCANCEL.",
                    "Notice",
                    MB_OK | MB_ICONINFORMATION);
            }
            else if(ret == -1){
                MessageBox(hwnd, "Dialog failed!", "Error",
                    MB_OK | MB_ICONINFORMATION);
            }
        }
        break;
        // Other menu commands...
    }
break;
```

Revenirea din fereastra dialog se va face numai dupa apasarea unuia dintre butoanele definite in fereastra de dialog. Acest mod de lucru se numeste modul “modal”.

Utilizarea functiei `CreateDialog()`. Diferenta fata de procedura `DialogBox()` este aceea ca revenirea in fereastra principala se poate face si fara inchiderea feretrei de dialog. Acest mod de lucru se numeste “modeless”.

Se creaza un fisier de resurse astfel:

```
IDD_TOOLBAR_DIALOGEX 0, 0, 98, 52
STYLE DS_MODALFRAME | WS_POPUP | WS_CAPTION
EXSTYLE WS_EX_TOOLWINDOW
CAPTION "My Dialog Toolbar"
FONT 8, "MS Sans Serif"
BEGIN
    PUSHBUTTON        "&Press This Button", IDC_PRESS, 7, 7, 84, 14
    PUSHBUTTON        "&Or This One", IDC_OTHER, 7, 31, 84, 14
END
```

Programul care creeaza un dialog pe modul modeless este:

```
HWND g_hToolbar = NULL;
case WM_CREATE:
    g_hToolbar = CreateDialog(GetModuleHandle(NULL),
MAKEINTRESOURCE(IDD_TOOLBAR),
    hwnd, ToolDlgProc);
    if(g_hToolbar != NULL)
    {
        ShowWindow(g_hToolbar, SW_SHOW);
    }
    else
    {
        MessageBox(hwnd, "CreateDialog returned NULL", "Warning!",
            MB_OK | MB_ICONINFORMATION);
    }
break;
```

```
BOOL CALLBACK ToolDlgProc(HWND hwnd, UINT Message, WPARAM wParam,
LPARAM lParam)
{
    switch(Message)
    {
        case WM_COMMAND:
            switch(LOWORD(wParam))
            {
                case IDC_PRESS:
                    MessageBox(hwnd, "Hi!", "This is a message",
                        MB_OK | MB_ICONEXCLAMATION);
                    break;
                case IDC_OTHER:
                    MessageBox(hwnd, "Bye!", "This is also a message",
                        MB_OK | MB_ICONEXCLAMATION);
                    break;
            }
            break;
        default:
            return FALSE;
    }
    return TRUE;
}
```

Pentru acest mod de lucru se poate utiliza functia DestroyWindow() pentru distrugerea ferestrei de dialog

```
case WM_DESTROY:
    DestroyWindow(g_hToolbar);
    PostQuitMessage(0);
break;
```

De asemenea fereastra de dialog poate fi ascunsa sau vizualizata:

```
case WM_COMMAND:
    switch(LOWORD(wParam))
    {
        case ID_DIALOG_SHOW:
            ShowWindow(g_hToolbar, SW_SHOW);
            break;
        case ID_DIALOG_HIDE:
            ShowWindow(g_hToolbar, SW_HIDE);
            break;
        //... other command handlers
    }
break;
```

Bucula de tratare a mesajelor trebuie modificata astfel:

```
while(GetMessage(&Msg, NULL, 0, 0))
{
    if(!IsDialogMessage(g_hToolbar, &Msg))
    {
        TranslateMessage(&Msg);
        DispatchMessage(&Msg);
    }
}
```

Se separa astfel mesajele destinate ferestrei de dialog de mesajele destinate ferestrei principale.