Тема № 10. Диалоговые окна и элементы управления

10.1. Классификация диалоговых окон

Диалоговые окна:

Признак модальности:

- модальные;
- немодальные.

Назначение:

- окна сообщений;
- стандартные (выбор файла, выбор цвета, выбор шрифта и др.);
- специальные (создаются программистом для решения своих задач).

10.2. Окна сообщений

Функция для создания окна сообщений

```
int MessageBoxA(HWND hWnd ,
   LPCSTR lpText, // Текст в окне
    LPCSTR lpCaption,  // Заголовок
    UINT uType);
```

Возможные значения параметра uType:

MB_OK

MB_OKCANCEL

MB_ABORTRETRYIGNORE

MB_YESNOCANCEL

MB_YESNO

MB_RETRYCANCEL

MB_ICONHAND

MB_ICONQUESTION

MB_ICONEXCLAMATION

MB_ICONASTERISK

Числовое значение, возвращаемое функцией:

Соответствует нажатой кнопке

Ok – 1, Cancel – 2, Abort – 3, Retry – 4, Ignore – 5, Yes – 6, No – 7

- #define IDOK 1
- #define IDCANCEL 2
- #define IDABORT 3
- #define IDRETRY 4
- #define IDIGNORE 5
- #define IDYES 6
- #define IDNO 7

10.3. Создание модального диалогового окна

Удобнее всего диалоговое окно вместе со своими элементами управления описывать в файле ресурсов.

Функция для создания окна

DialogBox(hInstance, //Хэндл приложения
lpTemplate, //Строка-название ресурса окна
hWndParent, //Хэндл родительского окна
lpDialogFunc) //Указатель на функцию окна

Функция для закрытия окна

BOOL EndDialog(HWND hDlg, int nResult);

Для инициализации диалогового окна (задания начального состояния элементов управления, присвоения значений переменным и др.) необходимо обработать сообщение

WM_INITDIALOG

10.4. Элементы управления диалогового окна

Элементы управления:

Основные:

– кнопка, контрольный переключатель, радиокнопка, текстовое поле, список,…

Общие:

– строка состояния, спин, регулятор, индикатор процесса, ...

Функция для получения хэндла элемента управления:

```
HWND GetDlgItem(HWND hDlg,
    //Хэндл род.окна
  int nIDDlgItem);//Идентиф. элемента
```

Функции для посылки сообщений элементам управления:

LRESULT SendDlgItemMessage(HWND hDlg, int nIDDlgItem, UINT Msg, WPARAM wParam, LPARAM lParam);

LRESULT SendMessage(HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam);

10.5. Кнопки, контр. переключатель, радио кнопка

Сообщение:

- WM_COMMAND

Младшее слово wParam содержит значение идентификатора элемента управления

Старшее слово wParam определяет действия с кнопкой (нотификационные сообщения):

- #define BN_CLICKED 0
- #define BN_PAINT 1
- #define BN_DOUBLECLICKED 5
- #define BN_SETFOCUS 6
- #define BN_KILLFOCUS 7

Сообщения, которые можно посылать:

- BM_GETCHECK BM_SETCHECK
  BM_GETSTATE BM_SETSTATE
  BM_SETSTYLE BM_CLICK
  BM_GETIMAGE BM_SETIMAGE

Пример обработки сообщений от кнопок

```
case WM_COMMAND:
    switch(LOWORD(wParam))
    {
    case IDOK: // Нажатие кнопок Ok и Cancel
    case IDCANCEL:
        EndDialog(hwnd, LOWORD(wParam));
        return TRUE;
    case ID_B1: // Нажатие кнопки с идентификатором ID_B1
        .......
    }
```

10.5. Текстовое поле

Нотификационные сообщения:

```
#define EN_SETFOCUS      0x0100
#define EN_KILLFOCUS     0x0200
#define EN_CHANGE        0x0300
#define EN_UPDATE        0x0400
#define EN_ERRSPACE      0x0500
#define EN_MAXTEXT       0x0501
#define EN_HSCROLL       0x0601
#define EN_VSCROLL       0x0602
```

Основные сообщения, которые можно посылать текстовым полям:

- WM_SETTEXT
- WM_GETTEXT

Поместить текст в текстовое поле:

SendDlgItemMessage(hwnd, IDC_EDIT1, WM_SETTEXT, 0, (LPARAM)Text);

Получить текст из текстового поля:

SendDlgItemMessage(hwnd, IDC_EDIT2,
WM_GETTEXT, 100, (LPARAM)Text);

10.6. Списки

Сообщения, посылаемые спискам:

LB_ADDSTRING
LB_INSERTSTRING
LB_DELETESTRING
LB_SETCURSEL
LB_GETCURSEL
LB_GETTEXTLEN
LB_GETCOUNT
LB_SELECTSTRING

LB_SETSEL
LB_GETSEL
LB_GETTEXT

Добавить строку текста в список:

```
SendDlgItemMessage(hwnd, IDC_LIST1,
   LB_ADDSTRING, 0, (LPARAM)Text);
```

Получить индекс выделенного элемента:

```
int i=SendDlgItemMessage(hwnd, IDC_LIST1,
   LB_GETCURSEL, 0, 0);
```

Извлечь строку из элемента с индексом i:

```
SendDlgItemMessage(hwnd, IDC_LIST1,
   LB_GETTEXT, i, (LPARAM)Text);
```

Удалить элемент с индексом i:

```
SendDlgItemMessage(hwnd, IDC_LIST1,
   LB_DELETESTRING, i, 0);
```

Коды нотификационных сообщений:

- LBN_ERRSPACE
  LBN_SELCHANGE
- LBN_DBLCLK
- LBN_SETFOCUS
- LBN_KILLFOCUS

Пример обработки нотификационных сообщений:

```
case WM_COMMAND:
    switch(LOWORD(wParam))
    {
        case IDC_LIST1: // Сообщение от списка
        if (HIWORD(wParam)==LBN_SELCHANGE)
        {……….
        }
        if (HIWORD(wParam)==LBN_DBLCLK)
        {……….
        }
        return TRUE;
        ……….
```

10.7. Общие элементы управления

Необходимо загрузить библиотеку, вызвать функцию

void InitCommonControls();

для этого подключить заголовочный файл

#include <commctrl.h>

В свойствах проекта для компоновщика установить необходимость использования библиотеки

comctl32.lib

10.8. Строка состояния

Чтобы определить число панелей необходимо послать сообщение SB_SETPARTS, при этом wParam определяет число панелей, а lParam должен содержать указатель на массив целых чисел, каждый элемент которого должен определять позицию правой границы соответствующей части, если элемент равен –1, то границей панели считается правая граница строки состояния.

```
case WM_INITDIALOG:
    InitCommonControls();
// Создать строку состояния
    hSB=CreateStatusWindow(WS_CHILD | WS_VISIBLE |
    SBARS_SIZEGRIP, "Simple", hwnd, 555);
// Добавить текст в строку состояния
    SendDlgItemMessage(hwnd, 555, SB_SETTEXT, 0, (LPARAM)"
    Текст в строке состояния");
```

10.9. Спин (стрелки)

Сообщения спина WM_VSCROLL, WM_HSCROLL

```
case WM_INITDIALOG:
// Получить хэндл текстового поля
    hE=GetDlgItem(hwnd, IDC_EDIT1);
// Установить для стрелок приятельское окно (текст. поле)
    SendDlgItemMessage(hwnd, IDC_SPIN1,
   UDM_SETBUDDY, (WPARAM)hE, 0);
// Установить диапазон стрелок
    SendDlgItemMessage(hwnd, IDC_SPIN1,
   UDM_SETRANGE, 0, 200);
// Установить начальное состояние
    SendDlgItemMessage(hwnd, IDC_SPIN1,
   UDM_SETPOS, 0, 100);
```

10.10. Регулятор и индикатор процесса

```
case WM_INITDIALOG:
// Установить диапазон и начальное состояние для регулятора и
// /индикатора
    SendDlgItemMessage(hwnd, IDC_SLIDER1, TBM_SETRANGE,
   (WPARAM)TRUE, (LPARAM)MAKELONG(0, 200));
    SendDlgItemMessage(hwnd, IDC_SLIDER1, TBM_SETPOS,
   (WPARAM)TRUE, 100);
    SendDlgItemMessage(hwnd, IDC_PROGRESS1, PBM_SETRANGE,
   (WPARAM)0, (LPARAM)MAKELONG(0, 200));
    SendDlgItemMessage(hwnd, IDC_PROGRESS1, PBM_SETPOS, 100, 0);
case WM_HSCROLL:
// Обработка сообщения получаем значение регулятора и устанавливаем
//  такое же  значение индикатора процесса
    SendDlgItemMessage(hwnd, IDC_PROGRESS1, PBM_SETPOS,
   SendDlgItemMessage(hwnd, IDC_SLIDER1, TBM_GETPOS, 0, 0) , 0);
```

10.11. Немодальные диалоговые окна

1. Вместо функции DialogBox используется функция

CreateDialog(hInst, (LPCTSTR)IDD_DIALOG2, hWnd, DialogFun2);

2. В свойства окна установить Visible – True.

3. При закрытии окна вместо функции EndDialog используется функция

DestroyWindow(hwnd);