***Министерство образования Республики Беларусь***

***Учреждение образования***

***«Брестский государственный технический университет»***

***Кафедра ИИТ***

**Лабораторная работа №4**

**По дисциплине ППвИС за IV семестр**

**Тема: «Создание интерфейсов. Диалоговые окна. Меню. Каркас Hello»**

**Выполнил:**

Студент группы ИИ-15 (1)

2-го курса

Волк И. А.

**Проверил:**

Муравьев Г.Л.

Брест 2019

Цель:

* Изучить использование типовых ЭУ (кнопок, окон редактирования, списков)
* Изучить управление меню, технологию создание интерфейсов с использованием меню и диалоговых окон
* Изучить типовые диалоговые окна
* Изучить создание приложений на базе каркаса Hello

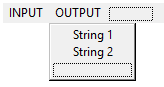
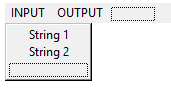
Состав отчета:

* Диаграммы состояний
* Листинги обработчиков
* Изображения результатов

Ход работы:

**Меню.**

1. Изучим теоретический материал по созданию и использованию пользовательского меню в составе главного окна.
2. Разработаем приложение на базе ТКП с окном с рамкой в качестве главного, содержащее простейшее пользовательское меню.



1. Внесем изменения в меню

Изменим “String 1” в “Input” на “Exit”. Пункт “String 1” в “Output” сделаем POPUP с пунктами “Additional 1” и “Additional 2”.

Обработчик сообщений от элементов управления:

case WM\_COMMAND:

{

WORD id = LOWORD(wParam);

switch(id)

{

case ID\_INPUT\_EXIT:

MessageBox(hWnd, "Do you really want to exit?",

"Confirmation", MB\_OKCANCEL);

break;

case ID\_INPUT\_STRING1:

MessageBox(hWnd, "String1","", MB\_OK);

break;

case ID\_OUTPUT\_STRING1\_ADDITIONAL1:

MessageBox(hWnd, "INPUT->String1->Additional1", "", MB\_OK);

break;

case ID\_OUTPUT\_STRING1\_ADDITIONAL2:

MessageBox(hWnd, "INPUT->String1->Additional2", "", MB\_OK);

break;

case ID\_OUTPUT\_STRING2:

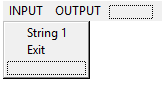
MessageBox(hWnd, "OUTPUT->String2", "", MB\_OK);

break;

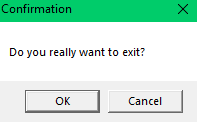
}

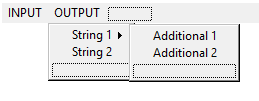
}

break;

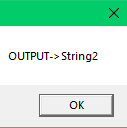


При нажатии INPUT->Exit



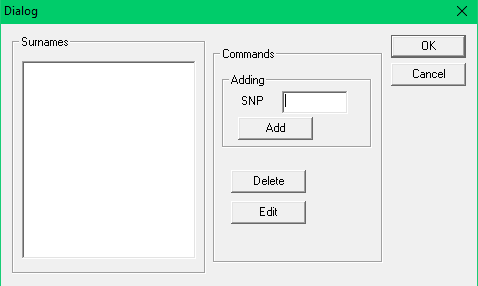


Примет работы нажатия на String 2:

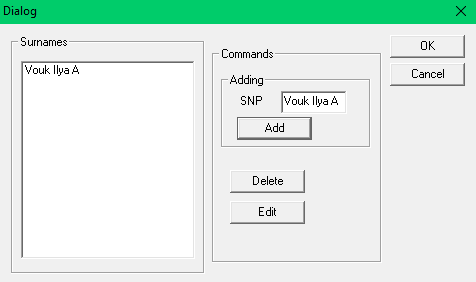


1. Использование диалогового окна со списком и окошком редактирования в качестве главного.

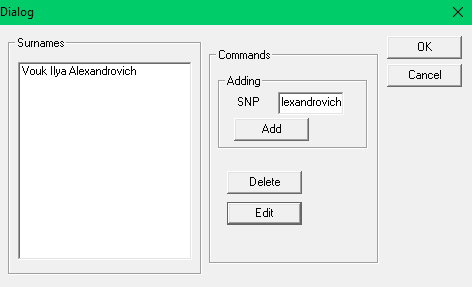
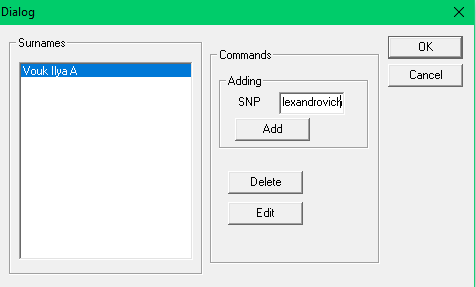
Вид окна:



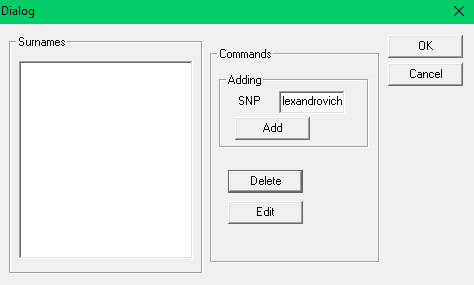
Добавление элемента:



Редактирование выбранного элемента:



Удаление:

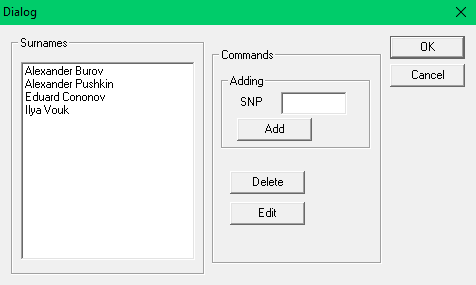


1. Модифицируем приложение для работы с заранее заданными строками.

Строки будем загружать из файла db.txt:



Получаем:



Листинг программы:

LRESULT CALLBACK DlgSurnameList(HWND hWnd, UINT msg, WPARAM wParam, LPARAM lParam)

{

switch(msg)

{

case WM\_KEYDOWN:

switch(LOWORD(wParam))

{

case VK\_RETURN:

SendMessage(hWnd, WM\_COMMAND, IDC\_ADD, 0);

break;

default:

return DefWindowProc(hWnd, msg, wParam, lParam);

}

case WM\_INITDIALOG:

{

std::ifstream fin("db.txt");

if(!fin)

{

MessageBox(hWnd, "Cannot open db", "", MB\_OK);

PostQuitMessage(0);

return FALSE;

}

std::string sBuffer;

while(std::getline(fin, sBuffer))

SendDlgItemMessage(hWnd, IDC\_SURNAMES, LB\_ADDSTRING, 0, (LPARAM)sBuffer.c\_str());

return TRUE;

}

case WM\_COMMAND:

switch(LOWORD(wParam))

{

case IDOK:

PostQuitMessage(0);

break;

case IDCANCEL:

PostQuitMessage(0);

break;

case IDC\_ADD:

{

char szBuffer[STR\_SIZE] = "";

GetDlgItemText(hWnd, IDC\_EDIT, szBuffer, STR\_SIZE);

SendDlgItemMessage(hWnd, IDC\_SURNAMES, LB\_ADDSTRING, 0, (LPARAM)szBuffer);

}

break;

case IDC\_DELETE:

{

int nSelected = SendDlgItemMessage(hWnd, IDC\_SURNAMES, LB\_GETCURSEL, 0, 0);

if(nSelected == LB\_ERR)

{

MessageBox(hWnd, "There are no selected items", "", MB\_OK);

break;

}

SendDlgItemMessage(hWnd, IDC\_SURNAMES, LB\_DELETESTRING, nSelected, 0);

}

break;

case IDC\_CHANGE:

{

int nSelected = SendDlgItemMessage(hWnd, IDC\_SURNAMES, LB\_GETCURSEL, 0, 0);

if(nSelected == LB\_ERR)

{

MessageBox(hWnd, "There are no selected items", "", MB\_OK);

break;

}

char szBuffer[STR\_SIZE] = "";

GetDlgItemText(hWnd, IDC\_EDIT, szBuffer, STR\_SIZE);

SendDlgItemMessage(hWnd, IDC\_SURNAMES, LB\_DELETESTRING, nSelected, 0);

SendDlgItemMessage(hWnd, IDC\_SURNAMES, LB\_INSERTSTRING, nSelected, (LPARAM)szBuffer);

}

break;

}

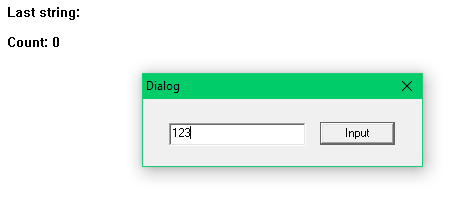
}

return 0;

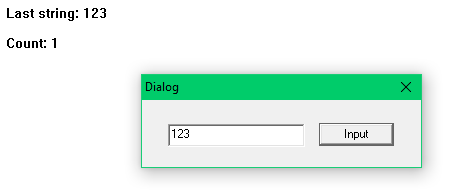
}

1. Напишем приложение для ввода-вывода строк с фиксацией количества введенных строк.

До нажатия на Input:



После нажатия на Input:



Листинг программы:

#define DO\_INC 12345

LRESULT CALLBACK WndProc(HWND hWnd, UINT msg, WPARAM wParam, LPARAM lParam)

{

HDC hdc;

PAINTSTRUCT ps;

static std::string sLastString;

static int count = 0;

WORD wmId = LOWORD(wParam);

WORD wmEvent = HIWORD(wParam);

switch(msg)

{

case DO\_INC:

sLastString = (char\*)lParam;

count++;

InvalidateRect(hWnd, NULL, TRUE);

break;

case WM\_COMMAND:

switch(wmId)

{

case ID\_INPUT:

DialogBox(0, MAKEINTRESOURCE(IDD\_INPUT), hWnd, (DLGPROC)InputProc);

break;

}

case WM\_PAINT:

{

hdc = BeginPaint(hWnd, &ps);

std::string sLine1 = "Last string: " + sLastString;

char szLine2[STR\_SIZE];

memset(szLine2, 0, STR\_SIZE);

wsprintf(szLine2, "Count: %d", count);

TextOut(hdc, 10, 10, sLine1.c\_str(), sLine1.length());

TextOut(hdc, 10, 40, szLine2, STR\_SIZE);

EndPaint(hWnd, &ps);

}

break;

case WM\_DESTROY:

PostQuitMessage(0);

break;

default:

return DefWindowProc(hWnd, msg, wParam, lParam);

}

return 0;

}

LRESULT CALLBACK InputProc(HWND hDlg, UINT msg, WPARAM wParam, LPARAM lParam)

{

switch(msg)

{

case WM\_INITDIALOG:

return TRUE;

case WM\_COMMAND:

{

switch(LOWORD(wParam))

{

case IDCANCEL:

EndDialog(hDlg, 0);

break;

case ID\_INPUT:

{

char szBuffer[STR\_SIZE] = "";

GetDlgItemText(hDlg, IDC\_EDIT1, szBuffer, STR\_SIZE);

if (strlen(szBuffer) == 0)

{

MessageBox(hDlg, "No text", "", MB\_OK);

break;

}

SendMessage(g\_hWnd, DO\_INC, 0, (LPARAM)szBuffer);

break;

}

}

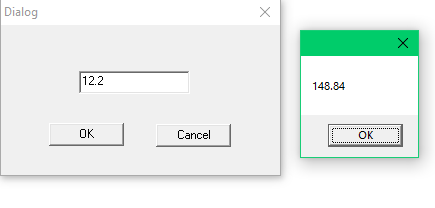
}

}

return 0;

}

11. Создадим проект на базе шаблона Hello. Модифицируем меню. По выбору пункта Enter активируем диалоговое окно для ввода числа. Число будем возводить в квадрат вы выводить через окно сообщения.



Листинг программы:

LRESULT CALLBACK Dlg(HWND hDlg, UINT message, WPARAM wParam, LPARAM lParam)

{

switch (message)

{

case WM\_INITDIALOG:

return TRUE;

case WM\_COMMAND:

if (LOWORD(wParam) == IDOK)

{

char szNumber[MAX\_LOADSTRING] = "";

GetDlgItemText(hDlg, IDC\_EDIT1, szNumber, MAX\_LOADSTRING);

double fNumber = atof(szNumber);

fNumber \*= fNumber;

std::stringstream ss;

ss << fNumber;

std::string sMsg (ss.str());

char szMsg[MAX\_LOADSTRING];

wsprintf(szMsg, "Result - %f", fNumber);

MessageBox(hDlg, sMsg.c\_str(), "", MB\_OK);

}

else if (LOWORD(wParam) == IDOK || LOWORD(wParam) == IDCANCEL)

{

EndDialog(hDlg, LOWORD(wParam));

return TRUE;

}

break;

}

return FALSE;

}

Вывод: по ходу данной лабораторной работы научились работать с меню и элементами управления на диалоговых окнах.