Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут”

Кафедра ОТ

ЗВІТ

про виконання комп’ютерного практикуму № 6

з дисципліни

“Об’єктно орієнтоване програмування ”

Тема: «Робота з табличними даними»

|  |  |  |
| --- | --- | --- |
| Прийняв: |  | Виконав: |
| Порєв В.М. |  | студент 2-го курсу  гр. ІП-54 ФІОТ  Макаренко Антон Олександрович |

Київ 2016

**Мета:** Мета роботи – отримати вміння та навички програмувати табличні дані на С++, запрограмувавши простий графічний редактор в об’єктно-орієнтованому стилі.

**Завдання:**

1. Створити у середовищі MS Visual Studio C++ проект Win32 з ім’ям **Lab6**.
2. Написати вихідний текст програми згідно варіанту завдання.
3. Скомпілювати вихідний текст і отримати виконуваний файл програми.
4. Перевірити роботу програми. Налагодити програму.
5. Проаналізувати та прокоментувати результати та вихідний текст програми.
6. Оформити звіт.

Лістинг

DlgTable.h

#include "ShapeEditor.h"

INT\_PTR CALLBACK DlgTableProc(HWND hDlg, UINT message, WPARAM wParam, LPARAM lParam);

void OnInit(HWND hDlg);

void OnPaint(HWND hdlg);

void AddItem(ShapeInfo info);

DlgTable.cpp

#include "stdafx.h"

#include "DlgTable.h"

HWND hTable;

int countItems = 0;

int curentItem = 0;

SCROLLINFO VscrollInfo;

HWND hDlgT;

//21

INT\_PTR CALLBACK DlgTableProc(HWND hDlg, UINT message, WPARAM wParam, LPARAM lParam)

{

int yCurrentScroll;

int yDelta;

int yNewPos;

//SetFocus(hDlg);

UNREFERENCED\_PARAMETER(lParam);

switch (message)

{

//case WM\_CREATE:

// OnInit(hDlg);

// break;

case WM\_INITDIALOG:

hDlgT = hDlg;

OnInit(hDlg);

return (INT\_PTR)TRUE;

case WM\_VSCROLL:

/\*VscrollInfo.nPos = 20;

SetScrollInfo(hDlg, SB\_VERT, &VscrollInfo, false);\*/

yCurrentScroll = GetScrollPos(hDlg, SB\_VERT);

switch (LOWORD(wParam))

{

case SB\_PAGEDOWN:

yNewPos = min(yCurrentScroll + 5, VscrollInfo.nMax);

break;

case SB\_PAGEUP:

yNewPos = max(yCurrentScroll - 5, VscrollInfo.nMin);

break;

case SB\_LINEDOWN:

yNewPos = min(yCurrentScroll + 1, VscrollInfo.nMax);

break;

case SB\_LINEUP:

yNewPos = max(yCurrentScroll - 1, VscrollInfo.nMin);

break;

case SB\_THUMBTRACK:

yNewPos = HIWORD(wParam);

break;

default:

yNewPos = yCurrentScroll;

}

//SetScrollPos(hWnd, SB\_VERT, yNewPos, false);

if (yNewPos > VscrollInfo.nPos)

ListView\_EnsureVisible(hTable, yNewPos + 21, false);

else

ListView\_EnsureVisible(hTable, yNewPos, false);

VscrollInfo.nPos = yNewPos;

SetScrollInfo(hDlg, SB\_VERT, &VscrollInfo, false);

//SetScrollsPropertys(hWnd);

InvalidateRect(hDlg, 0, false);

break;

case WM\_KEYDOWN:

switch (wParam)

{

case VK\_ESCAPE:

//SendMessage(hTable, WM\_VSCROLL, 10, 10);

ListView\_EnsureVisible(hTable, min(curentItem - 21, 0), false);

curentItem = max(curentItem - 1, 0);

break;

case VK\_RETURN:

ListView\_EnsureVisible(hTable, max(curentItem + 21, 0), false);

curentItem = min(curentItem + 1, 0);

break;

default:

break;

}

break;

case WM\_COMMAND:

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

{

EndDialog(hDlg, LOWORD(wParam));

return (INT\_PTR)TRUE;

}

break;

default:

break;

}

return (INT\_PTR)FALSE;

}

void OnInit(HWND hDlg)

{

VscrollInfo.cbSize = sizeof(SCROLLINFO);

VscrollInfo.nMin = 0;

VscrollInfo.fMask = SIF\_ALL;

VscrollInfo.nPos = 0;

VscrollInfo.nMax = 1;

VscrollInfo.nPage = 21;

SetScrollInfo(hDlg, SB\_VERT, &VscrollInfo, false);

RECT rcClient;

GetClientRect(hDlg, &rcClient);

hTable = CreateWindow(WC\_LISTVIEW,

"",

WS\_VISIBLE | WS\_BORDER | WS\_CHILD | LVS\_REPORT | LVS\_EDITLABELS | LVS\_NOCOLUMNHEADER,

0, 0,

rcClient.right + 20,

rcClient.bottom + 10,

hDlg,

(HMENU)1008,

NULL,

NULL);

ListView\_SetExtendedListViewStyleEx(hTable,

LVS\_EX\_GRIDLINES, LVS\_EX\_GRIDLINES);

ListView\_SetExtendedListViewStyleEx(hTable,

LVS\_EX\_FULLROWSELECT, LVS\_EX\_FULLROWSELECT);

LVCOLUMN lvc;

memset(&lvc, 0, sizeof(LV\_COLUMN));

lvc.mask = LVCFMT\_LINE\_BREAK | LVCF\_WIDTH | LVCF\_TEXT;

lvc.iSubItem = 0;

lvc.cx = 100;

lvc.fmt = LVCFMT\_LEFT;

ListView\_InsertColumn(hTable, 0, &lvc);

lvc.cx = 100;

lvc.iSubItem = 1;

ListView\_InsertColumn(hTable, 1, &lvc);

lvc.cx = 100;

lvc.iSubItem = 2;

ListView\_InsertColumn(hTable, 2, &lvc);

lvc.cx = 100;

lvc.iSubItem = 1;

ListView\_InsertColumn(hTable, 3, &lvc);

lvc.cx = 100;

lvc.iSubItem = 2;

ListView\_InsertColumn(hTable, 4, &lvc);

LV\_ITEM lvI;

memset(&lvI, 0, sizeof(LVITEM));

lvI.mask = LVIF\_TEXT | LVIF\_STATE | LVIF\_COLUMNS | LVIF\_COLFMT;

lvI.stateMask = 0;

lvI.iSubItem = 0;

lvI.state = 0;

lvI.iItem = 0;

ListView\_InsertItem(hTable, &lvI);

lvI.iSubItem = 1;

lvI.iItem = 1;

ListView\_InsertItem(hTable, &lvI);

ListView\_SetItemText(hTable, 0, 0, TEXT("Назва"));

ListView\_SetItemText(hTable, 0, 1, TEXT("x start"));

ListView\_SetItemText(hTable, 0, 2, TEXT("y start"));

ListView\_SetItemText(hTable, 0, 3, TEXT("x end"));

ListView\_SetItemText(hTable, 0, 4, TEXT("y end"));

}

void OnPaint(HWND hDlg)

{

}

void AddItem(ShapeInfo info)

{

char\* \_DstBuf = new char[10];

countItems++;

VscrollInfo.nMax = max(countItems, 0);

SetScrollInfo(hDlgT, SB\_VERT, &VscrollInfo, false);

LV\_ITEM lvI;

memset(&lvI, 0, sizeof(LVITEM));

lvI.pszText = info.name;

lvI.mask = LVIF\_TEXT | LVIF\_STATE | LVIF\_COLUMNS | LVIF\_COLFMT;

lvI.stateMask = 0;

lvI.iSubItem = 0;

lvI.state = 0;

lvI.iItem = countItems;

ListView\_InsertItem(hTable, &lvI);

ListView\_SetItemText(hTable, countItems, 0, TEXT(info.name));

\_itoa\_s(info.start\_pt.x, \_DstBuf, 10, 10);

ListView\_SetItemText(hTable, countItems, 1, TEXT(\_DstBuf));

\_itoa\_s(info.start\_pt.y, \_DstBuf, 10, 10);

ListView\_SetItemText(hTable, countItems, 2, TEXT(\_DstBuf));

\_itoa\_s(info.end\_pt.x, \_DstBuf, 10, 10);

ListView\_SetItemText(hTable, countItems, 3, TEXT(\_DstBuf));

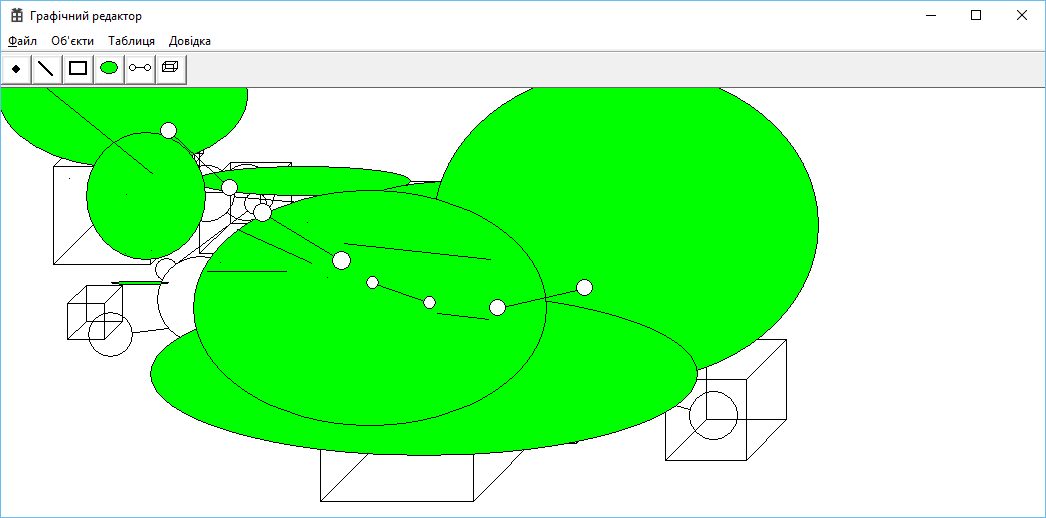
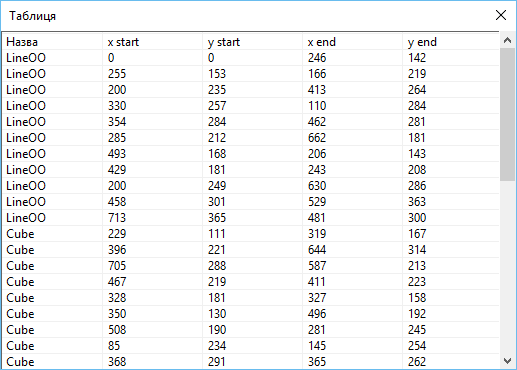
\_itoa\_s(info.end\_pt.y, \_DstBuf, 10, 10);

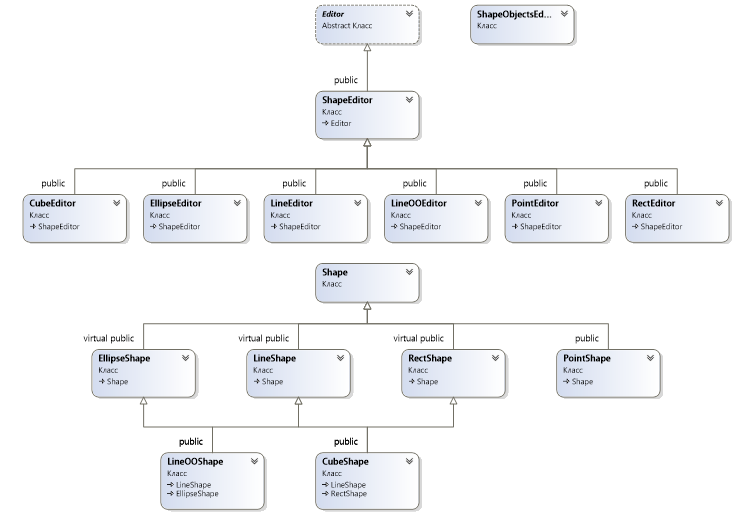
ListView\_SetItemText(hTable, countItems, 4, TEXT(\_DstBuf));

//ListView\_EnsureVisible(hTable, countItems, false);

}

Відеокопія результату

Діаграма класів 

Висновок

В ході даної лабораторної роботи я отримав вміння та навички програмувати табличні дані на С++, запрограмувавши простий графічний редактор в об’єктно-орієнтованому стилі.