#include "stdafx.h"

#include "Victorina.h"

#include"Resource.h"

#include "mmSystem.h"

#define MAX\_LOADSTRING 100

//////////////////////////////

//////////////////////////////

// Глобальные переменные:

HINSTANCE hInst;

HINSTANCE hInstance;// текущий экземпляр

WCHAR szTitle[MAX\_LOADSTRING]; // Текст строки заголовка

WCHAR szWindowClass[MAX\_LOADSTRING]; // имя класса главного окна

PAINTSTRUCT ps;

HDC hdc;

TEXTMETRIC tm;

HWND hwndButtonChose;

static HWND hwndButton[10];

HWND hwndButtonVictorina[50];

HWND hwndButton\_Victorina\_Guestion[50];

HWND hwndButtonVictorinaPodskazki[10];

HWND hWnd;

int ReloadGame = 0;

int CountWin = 0;

int count =0;

int counterGame = 0;

int counter = 0;

int counterGameGG = 0;

int r[5], g[5], b[5];

static int cxChar, cyChar;

int rt;//переменная для подсчета квадратов

HBRUSH R, G, B;//кисточки для работы с прямоугольниками

// Отправить объявления функций, включенных в этот модуль кода:

ATOM MyRegisterClass(HINSTANCE hInstance);

BOOL InitInstance(HINSTANCE, int);

LRESULT CALLBACK WndProc(HWND, UINT, WPARAM, LPARAM);

////////////////////////////////////////////////////////////////

INT\_PTR CALLBACK FistGuestion(HWND, UINT, WPARAM, LPARAM);

INT\_PTR CALLBACK NumbTwo(HWND, UINT, WPARAM, LPARAM);

INT\_PTR CALLBACK NumbTree(HWND, UINT, WPARAM, LPARAM);

INT\_PTR CALLBACK NumbFour(HWND, UINT, WPARAM, LPARAM);

INT\_PTR CALLBACK NumbFive(HWND, UINT, WPARAM, LPARAM);

INT\_PTR CALLBACK NumbSix(HWND, UINT, WPARAM, LPARAM);

INT\_PTR CALLBACK NumbSeven(HWND, UINT, WPARAM, LPARAM);

INT\_PTR CALLBACK NumbEight(HWND, UINT, WPARAM, LPARAM);

INT\_PTR CALLBACK NumbNine(HWND, UINT, WPARAM, LPARAM);

INT\_PTR CALLBACK NumbTen(HWND, UINT, WPARAM, LPARAM);

//////////////////////////////////////////////////////////////

int APIENTRY wWinMain(\_In\_ HINSTANCE hInstance,

\_In\_opt\_ HINSTANCE hPrevInstance,

\_In\_ LPWSTR lpCmdLine,

\_In\_ int nCmdShow)

{

UNREFERENCED\_PARAMETER(hPrevInstance);

UNREFERENCED\_PARAMETER(lpCmdLine);

// TODO: разместите код здесь.

// Инициализация глобальных строк

LoadStringW(hInstance, IDS\_APP\_TITLE, szTitle, MAX\_LOADSTRING);

LoadStringW(hInstance, IDC\_VICTORINA, szWindowClass, MAX\_LOADSTRING);

MyRegisterClass(hInstance);

// Выполнить инициализацию приложения:

if (!InitInstance (hInstance, nCmdShow))

{

return FALSE;

}

HACCEL hAccelTable = LoadAccelerators(hInstance, MAKEINTRESOURCE(IDC\_VICTORINA));

MSG msg;

// Цикл основного сообщения:

while (GetMessage(&msg, nullptr, 0, 0))

{

if (!TranslateAccelerator(msg.hwnd, hAccelTable, &msg))

{

TranslateMessage(&msg);

DispatchMessage(&msg);

}

}

return (int) msg.wParam;

}

ATOM MyRegisterClass(HINSTANCE hInstance)

{

WNDCLASSEXW wcex;

wcex.cbSize = sizeof(WNDCLASSEX);

wcex.style = CS\_HREDRAW | CS\_VREDRAW;

wcex.lpfnWndProc = WndProc;

wcex.cbClsExtra = 0;

wcex.cbWndExtra = 0;

wcex.hInstance = hInstance;

wcex.hIcon = LoadIcon(hInstance, MAKEINTRESOURCE(IDI\_ICON1));

wcex.hCursor = LoadCursor(nullptr, IDC\_ARROW);

wcex.hbrBackground = (HBRUSH)CreateSolidBrush(RGB(20, 20, 120));

wcex.lpszMenuName = MAKEINTRESOURCEW(IDC\_VICTORINA);

wcex.lpszClassName = szWindowClass;

wcex.hIconSm = LoadIcon(wcex.hInstance, MAKEINTRESOURCE(IDI\_ICON1));

return RegisterClassExW(&wcex);

}

BOOL InitInstance(HINSTANCE hInstance, int nCmdShow)

{

hInst = hInstance; // Сохранить дескриптор экземпляра в глобальной переменной

HWND hWnd = CreateWindowW(szWindowClass, szTitle, WS\_OVERLAPPED| WS\_SYSMENU,

CW\_USEDEFAULT, 0, CW\_USEDEFAULT, 0, nullptr, nullptr, hInstance, nullptr);

if (!hWnd)

{

return FALSE;

}

ShowWindow(hWnd, nCmdShow);

UpdateWindow(hWnd);

return TRUE;

}

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

{

switch (message)

{

case WM\_COMMAND:

{

case WM\_KEYDOWN:

#define HOTKEY1 2016

//определение комбинаций клавиш

RegisterHotKey(hWnd, HOTKEY1, MOD\_CONTROL, VK\_F1);

//цикл обработки сообщений горячих клавиш

//Enter\_The\_Game

case WM\_HOTKEY:{

if (HOTKEY1 == (int)wParam) {

PlaySound(NULL, NULL, SND\_SYSTEM);

InvalidateRect(hWnd, NULL, TRUE);

for (int i = 0; i < 10; i++) {

DestroyWindow(hwndButton\_Victorina\_Guestion[i]);

}

if (MessageBox(hWnd, "Начать игру ?\n\nСо всеми вопросами смотрите в пункт меню Instruction ", "Hello", MB\_YESNO) == IDYES) {

hwndButtonVictorina[0] = CreateWindow("BUTTON", "PRES\_TO\_START\_FIRST\_GAME",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

10, // starting x position

10, // starting y position

400, // button width

50, // button height

hWnd, // parent window

(HMENU)11, // No menu

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

//////////////end button control

}

//Esit\_from\_game

else {

for (int i = 0; i < 10; i++) {

DestroyWindow(hwndButton\_Victorina\_Guestion[i]);

}

if (MessageBox(hWnd, "Exit?", "Are\_you\_want\_this", MB\_YESNO) == IDYES) {

DestroyWindow(hWnd);

}

}

}//end hotkey

////////////////////////////////////////////////////////////////////???????????????????????????????????

}//end of hotkey process

int wmId = LOWORD(wParam);// Разобрать выбор в меню:

switch (wmId) {

//start Game

case 11: {

PlaySound(TEXT("C:\\Users\\Пользователь\\Documents\\Visual Studio 2015\\Projects\\Victorina\\mainsound.wav"), NULL, SND\_LOOP | SND\_ASYNC);

////////////////////////////////////////////////

static int cxChar, cyChar;

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

/////////////////////////////////////////////////

for (int up = 0; up < 5; up++) {

R = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, R);

//////////////////////////вверхний край

r[up] = Rectangle(hdc,

(cxChar+750)+(50\*up), cyChar \*(5),

(102\*cxChar)+50\*up, (5) \* cyChar/2);

///////////////////////////НИЖНИЙ КРАЙ

r[up] = Rectangle(hdc,

(cxChar + 750) + (50 \* up), cyChar \*(5)+144,

(102 \* cxChar) + 50 \* up, ((5) \* cyChar / 2)+144);

}

/////////////////////////////////////////

for (rt = 0; rt < 4; rt++) {

//////////////////////////////////////////////

R = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, R);

r[rt] = Rectangle(hdc,

cxChar + 700, cyChar \*(5 + (3 \* rt)),

95 \* cxChar, (5 + 6 \* rt) \* cyChar / 2);

////////////////////////////////////////////

r[rt] = Rectangle(hdc,

cxChar + 1000, cyChar \*(5 + (3 \* rt)),

137 \* cxChar + 4, (5 + 6 \* rt) \* cyChar / 2);

}

ReleaseDC(hWnd, hdc);

//////////////////////////////////////////////

hwndButton\_Victorina\_Guestion[0] = CreateWindow("BUTTON", "Fist\_Guestion",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

450, // starting x position

41, // starting y position

200, // button width

30, // button height

hWnd, // parent window

(HMENU)0, // No menu

hInstance,

//(HINSTANCE)GetWindowLong(hWnd,NULL),

NULL);

///////////////////////////////////////////////////////

DestroyWindow(hwndButtonVictorina[0]);//button start1

break;

///////////////////////////////////

case ID\_INSTRUCTION://пункт instruction

{

MessageBox(hWnd, "Интрукция \n\n Для начала игры нажмите комбинацию Ctr+F1 \n\n "

"Для вывода информации о создателе программы нажмите на кнопку About в панеле меню\n\n "

"Для выхода из программы нажимите Exit \n"

,"Инструкция", MB\_OK);

break;

}

///////////////////////////////////////////////////////////////////////////////

case ID\_ABOUT: {

MessageBox(hWnd, "Victorina ver 1.0 \n\n "

"Developer : Predator \n\n "

"HAVE FAN !!!!!!!!!\n\n"

, "About", MB\_OK);

break;

}

///////////////////////////////////////////////////////////////////

case IDM\_EXIT: {

DestroyWindow(hWnd);

break;

}

///////////////////////////////////////////////////////

}//end//start

///////////////////////////////////////////

}//end of switch for chose players

int wmIdd = LOWORD(wParam);//обработчик сообщений от диалогового окна номер 2

switch (wmIdd) {

//первый вопрос

case 0: {

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

////////////////////////////////////////////////////////

hwndButtonVictorinaPodskazki[0] = CreateWindow("BUTTON", "Подсказка",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

671, // starting x position

250, // starting y position

100, // button width

50, // button height

hWnd, // parent window

(HMENU)20,

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

///////////////////////////////////////////////////////

//////////////////////////////////////////////

B = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, B);

b[0] = Rectangle(hdc,

cxChar + 750, cyChar \*(5) + 48,

102 \* cxChar, ((5) \* cyChar / 2) + 48);

DestroyWindow(hwndButton\_Victorina\_Guestion[0]);

if (counter >= 1) {

DestroyWindow(hwndButtonVictorinaPodskazki[0]);

}

////////////////////////////////////////

///////////////////////////////////////

if (counterGameGG == 0) {

hwndButton\_Victorina\_Guestion[1] = CreateWindow("BUTTON", "Second\_Guestion",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

450, // starting x position

80, // starting y position

200, // button width

30, // button height

hWnd, // parent window

(HMENU)1, // No menu

hInstance,

//(HINSTANCE)GetWindowLong(hWnd,NULL),

NULL);

}

/////////////////////////////////////////

DialogBox(hInst, MAKEINTRESOURCE(IDD\_FistGuestion), hwndButtonChose, FistGuestion);

//////////////////////////////////////////////////

DestroyWindow(hwndButtonVictorinaPodskazki[0]);

break;

case 20: {

if (MessageBox(hWnd, "Хотите использовать подсказку ?", "", MB\_YESNO) == IDYES) {

counter++;

MessageBox(hWnd, "Uncorrect:\nNope\nГермания", "Answer", MB\_OK);

DestroyWindow(hwndButtonVictorinaPodskazki[0]);

}

else {

DestroyWindow(hwndButtonVictorinaPodskazki[0]);

}

break;

}

}

////////////////////////////////////////////

//Second вопрос

case 1: {

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

/////////////////////////////////////////////////////

hwndButtonVictorinaPodskazki[1] = CreateWindow("BUTTON", "Подсказка",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

671, // starting x position

250, // starting y position

100, // button width

50, // button height

hWnd, // parent window

(HMENU)21,

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

////////////////////////////////////////////////////

B = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, B);

b[0] = Rectangle(hdc,

cxChar + 800, cyChar \*(5) + 48,

109 \* cxChar, ((5) \* cyChar / 2) + 48);

///////////////////////////////////////////

DestroyWindow(hwndButton\_Victorina\_Guestion[1]);

///////////////////////////////////////

if (counter >= 1) {

DestroyWindow(hwndButtonVictorinaPodskazki[2]);

}

//////////////////////////////////////////////

if (counterGameGG <= 1) {

hwndButton\_Victorina\_Guestion[2] = CreateWindow("BUTTON", "Thrid\_Guestion",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

450, // starting x position

120, // starting y position

200, // button width

30, // button height

hWnd, // parent window

(HMENU)2, // No menu

hInstance,

// (HINSTANCE)GetWindowLong(hWnd,NULL),

NULL);

}

/////////////////////////////////////////////

DialogBox(hInst, MAKEINTRESOURCE(IDD\_StcondGuestion), hwndButtonChose, NumbTwo);

DestroyWindow(hwndButtonVictorinaPodskazki[1]);

break;

case 21: {

if (MessageBox(hWnd, "Хотите использовать подсказку ?", "", MB\_YESNO) == IDYES) {

counter++;

MessageBox(hWnd, "Uncorrect:\nNope\nКулхард", "Answer", MB\_OK);

DestroyWindow(hwndButtonVictorinaPodskazki[1]);

}

else {

DestroyWindow(hwndButtonVictorinaPodskazki[1]);

}

break;

}

}

//////////////////////////////////

//Thrid Guestion

case 2: {

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

////////////////////////////////////////////

hwndButtonVictorinaPodskazki[2] = CreateWindow("BUTTON", "Подсказка",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

671, // starting x position

250, // starting y position

100, // button width

50, // button height

hWnd, // parent window

(HMENU)22,

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

////////////////////////////////////////////

B = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, B);

b[0] = Rectangle(hdc,

cxChar + 850, cyChar \*(5) + 48,

116 \* cxChar+1, ((5) \* cyChar / 2) + 48);

DestroyWindow(hwndButton\_Victorina\_Guestion[2]);

//////////////////////////////////////

if (counter >= 1) {

DestroyWindow(hwndButtonVictorinaPodskazki[2]);

}

///////////////////////////////////////////////////

if (counterGameGG <= 2) {

hwndButton\_Victorina\_Guestion[3] = CreateWindow("BUTTON", "Fourth\_Guestion",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

450, // starting x position

160, // starting y position

200, // button width

30, // button height

hWnd, // parent window

(HMENU)3, // No menu

hInstance,

//(HINSTANCE)GetWindowLong(hWnd,NULL),

NULL);

}

//////////////////////////////////////////////

DialogBox(hInst, MAKEINTRESOURCE(IDD\_Third), hwndButtonChose, NumbTree);

DestroyWindow(hwndButtonVictorinaPodskazki[2]);

break;

//////////////////////////

case 22: {

if (MessageBox(hWnd, "Хотите использовать подсказку ?", "", MB\_YESNO) == IDYES) {

counter++;

MessageBox(hWnd, "Uncorrect:\nNope\nТрое", "Answer", MB\_OK);

DestroyWindow(hwndButtonVictorinaPodskazki[2]);

}

else {

DestroyWindow(hwndButtonVictorinaPodskazki[2]);

}

break;

}

}

/////////////////////////////////////////

//Forth Guestion

case 3: {

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

//////////////////////////////////////////////

hwndButtonVictorinaPodskazki[3] = CreateWindow("BUTTON", "Подсказка",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

671, // starting x position

250, // starting y position

100, // button width

50, // button height

hWnd, // parent window

(HMENU)23,

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

////////////////////////////////////////////

B = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, B);

b[0] = Rectangle(hdc,

cxChar + 900, cyChar \*(5) + 48,

123 \* cxChar + 2, ((5) \* cyChar / 2) + 48);

DestroyWindow(hwndButton\_Victorina\_Guestion[3]);

///////////////////////////////////////

if (counter >= 1) {

DestroyWindow(hwndButtonVictorinaPodskazki[3]);

}

/////////////////////////////////////

if (counterGameGG <= 3) {

hwndButton\_Victorina\_Guestion[4] = CreateWindow("BUTTON", "Fifth\_Guestion",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

450, // starting x position

200, // starting y position

200, // button width

30, // button height

hWnd, // parent window

(HMENU)4, // No menu

hInstance,

//(HINSTANCE)GetWindowLong(hWnd,NULL),

NULL);

}

////////////////////////////////////////

DialogBox(hInst, MAKEINTRESOURCE(IDD\_Four), hwndButtonChose, NumbFour);

DestroyWindow(hwndButtonVictorinaPodskazki[3]);

break;

case 23: {

if (MessageBox(hWnd, "Хотите использовать подсказку ?", "", MB\_YESNO) == IDYES) {

counter++;

MessageBox(hWnd, "Uncorrect:\nNope\n300", "Answer", MB\_OK);

DestroyWindow(hwndButtonVictorinaPodskazki[3]);

}

else {

DestroyWindow(hwndButtonVictorinaPodskazki[3]);

}

break;

}

}

/////////////////////////////////////////

//five Guestion

case 4:{

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

/////////////////////////////////////////////////////

hwndButtonVictorinaPodskazki[4] = CreateWindow("BUTTON", "Подсказка",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

671, // starting x position

250, // starting y position

100, // button width

50, // button height

hWnd, // parent window

(HMENU)24,

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

//////////

/////////////////////////////////////////////

B = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, B);

b[0] = Rectangle(hdc,

cxChar + 950, cyChar \*(5) + 48,

130 \* cxChar + 3, ((5) \* cyChar / 2) + 48);

DestroyWindow(hwndButton\_Victorina\_Guestion[4]);

///////////////////////////////////

if (counter >= 1) {

DestroyWindow(hwndButtonVictorinaPodskazki[4]);

}

////////////////////////////////////

if (counterGameGG <= 4) {

hwndButton\_Victorina\_Guestion[5] = CreateWindow("BUTTON", "Sixth\_Guestion",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

1025, // starting x position

42, // starting y position

200, // button width

29, // button height

hWnd, // parent window

(HMENU)5, // No menu

hInstance,

//(HINSTANCE)GetWindowLong(hWnd,NULL),

NULL);

}

/////////////////////////////////////////

DialogBox(hInst, MAKEINTRESOURCE(IDD\_Five), hwndButtonChose, NumbFive);

DestroyWindow(hwndButtonVictorinaPodskazki[4]);

break;

case 24: {

if (MessageBox(hWnd, "Хотите использовать подсказку ?" , "", MB\_YESNO) == IDYES) {

counter++;

MessageBox(hWnd, "Uncorrect:\nКоперник\nКеплер", "Answer", MB\_OK);

DestroyWindow(hwndButtonVictorinaPodskazki[4]);

}

else {

DestroyWindow(hwndButtonVictorinaPodskazki[4]);

}

break;

}

}

/////////////////////////

//Six Guestion

case 5: {

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

////////////////////////////////////////////////////////

hwndButtonVictorinaPodskazki[5] = CreateWindow("BUTTON", "Подсказка",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

671, // starting x position

250, // starting y position

100, // button width

50, // button height

hWnd, // parent window

(HMENU)25,

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

///////////////////////////////////////////////////

B = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, B);

b[0] = Rectangle(hdc,

cxChar + 750, cyChar \*(5) + 96,

102 \* cxChar , ((5) \* cyChar / 2) + 96);

DestroyWindow(hwndButton\_Victorina\_Guestion[5]);

////////////////////////////////////////

if (counter >= 1) {

DestroyWindow(hwndButtonVictorinaPodskazki[5]);

}

//////////////////////////////////////////

if (counterGameGG <= 5) {

hwndButton\_Victorina\_Guestion[6] = CreateWindow("BUTTON", "Seventh\_Guestion",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

1025, // starting x position

80, // starting y position

200, // button width

29, // button height

hWnd, // parent window

(HMENU)6, // No menu

hInstance,

//(HINSTANCE)GetWindowLong(hWnd,NULL),

NULL);

}

/////////////////////////////////////////

DialogBox(hInst, MAKEINTRESOURCE(IDD\_Six), hwndButtonChose, NumbSix);

DestroyWindow(hwndButtonVictorinaPodskazki[5]);

break;

case 25: {

if (MessageBox(hWnd, "Хотите использовать подсказку ?", "", MB\_YESNO) == IDYES) {

counter++;

MessageBox(hWnd, "Uncorrect:\nКреп-Лизет\nКреп-Мюзет", "Answer", MB\_OK);

DestroyWindow(hwndButtonVictorinaPodskazki[5]);

}

else {

DestroyWindow(hwndButtonVictorinaPodskazki[5]);

}

break;

}

}

//////////////////////

//Seven Guestion

case 6: {

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

///////////////////////////////////////////////////////

hwndButtonVictorinaPodskazki[6] = CreateWindow("BUTTON", "Подсказка",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

671, // starting x position

250, // starting y position

100, // button width

50, // button height

hWnd, // parent window

(HMENU)26,

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

//////////////////////////////////////////////////

B = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, B);

b[0] = Rectangle(hdc,

cxChar + 800, cyChar \*(5) + 96,

109 \* cxChar, ((5) \* cyChar / 2) + 96);

DestroyWindow(hwndButton\_Victorina\_Guestion[6]);

/////////////////////////////////////////

if (counter >= 1) {

DestroyWindow(hwndButtonVictorinaPodskazki[6]);

}

///////////////////////////////////////

if (counterGameGG <= 6) {

hwndButton\_Victorina\_Guestion[7] = CreateWindow("BUTTON", "Eighth\_Guestion",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

1025, // starting x position

120, // starting y position

200, // button width

29, // button height

hWnd, // parent window

(HMENU)7, // No menu

hInstance,

//(HINSTANCE)GetWindowLong(hWnd,NULL),

NULL);

}

/////////////////////////////////////////

DialogBox(hInst, MAKEINTRESOURCE(IDD\_Seven), hwndButtonChose, NumbSeven);

DestroyWindow(hwndButtonVictorinaPodskazki[6]);

break;

case 26: {

if (MessageBox(hWnd, "Хотите использовать подсказку ?", "", MB\_YESNO) == IDYES) {

counter ++ ;

MessageBox(hWnd, "Uncorrect:\nNope\n2", "Answer", MB\_OK);

DestroyWindow(hwndButtonVictorinaPodskazki[6]);

}

else {

DestroyWindow(hwndButtonVictorinaPodskazki[6]);

}

break;

}

}

/////////////////////////////////

//Eight Guestion

case 7: {

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

/////////////////////////////////////////////////

hwndButtonVictorinaPodskazki[7] = CreateWindow("BUTTON", "Подсказка",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

671, // starting x position

250, // starting y position

100, // button width

50, // button height

hWnd, // parent window

(HMENU)27,

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

/////////////////////////////////////////

B = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, B);

b[0] = Rectangle(hdc,

cxChar + 850, cyChar \*(5) + 96,

116 \* cxChar+1, ((5) \* cyChar / 2) + 96);

DestroyWindow(hwndButton\_Victorina\_Guestion[7]);

//////////////////////////////////////////

if (counter >= 1) {

DestroyWindow(hwndButtonVictorinaPodskazki[8]);

}

//////////////////////////////////////////

if (counterGameGG <= 7) {

hwndButton\_Victorina\_Guestion[8] = CreateWindow("BUTTON", "Ninth\_Guestion",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

1025, // starting x position

160, // starting y position

200, // button width

29, // button height

hWnd, // parent window

(HMENU)8, // No menu

hInstance,

//(HINSTANCE)GetWindowLong(hWnd,NULL),

NULL);

}

/////////////////////////////////////////

DialogBox(hInst, MAKEINTRESOURCE(IDD\_Eight), hwndButtonChose, NumbEight);

DestroyWindow(hwndButtonVictorinaPodskazki[7]);

break;

case 27: {

if (MessageBox(hWnd, "Хотите использовать подсказку ?", "", MB\_YESNO) == IDYES) {

counter++;

MessageBox(hWnd, "Uncorrect:\nNope\nТумак", "Answer", MB\_OK);

DestroyWindow(hwndButtonVictorinaPodskazki[7]);

}

else {

DestroyWindow(hwndButtonVictorinaPodskazki[7]);

}

break;

}

}

/////////////////////////////////

//Nine Guestion

case 8: {

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

/////////////////////////////////////////////////

hwndButtonVictorinaPodskazki[8] = CreateWindow("BUTTON", "Подсказка",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

671, // starting x position

250, // starting y position

100, // button width

50, // button height

hWnd, // parent window

(HMENU)28,

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

//////////////////////////////////////////////

B = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, B);

b[0] = Rectangle(hdc,

cxChar + 900, cyChar \*(5) + 96,

123 \* cxChar + 2, ((5) \* cyChar / 2) + 96);

DestroyWindow(hwndButton\_Victorina\_Guestion[8]);

/////////////////////////////////////////////

if (counter >= 1) {

DestroyWindow(hwndButtonVictorinaPodskazki[8]);

}

/////////////////////////////////////////////

if (counterGameGG <= 8) {

hwndButton\_Victorina\_Guestion[9] = CreateWindow("BUTTON", "Tenth\_Guestion",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

1025, // starting x position

200, // starting y position

200, // button width

29, // button height

hWnd, // parent window

(HMENU)9, // No menu

hInstance,

//(HINSTANCE)GetWindowLong(hWnd,NULL),

NULL);

}

/////////////////////////////////////////////

DialogBox(hInst, MAKEINTRESOURCE(IDD\_Nine), hwndButtonChose, NumbNine);

DestroyWindow(hwndButtonVictorinaPodskazki[8]);

break;

case 28: {

if (MessageBox(hWnd, "Хотите использовать подсказку ?", "", MB\_YESNO) == IDYES) {

counter++;

MessageBox(hWnd, "Uncorrect:\nЗащитник\nПолузащитник", "Answer", MB\_OK);

DestroyWindow(hwndButtonVictorinaPodskazki[8]);

}

else {

DestroyWindow(hwndButtonVictorinaPodskazki[8]);

}

break;

}

}

//////////////////////////

//Ten Guestion

case 9: {

hdc = GetDC(hWnd);

GetTextMetrics(hdc, &tm);

cxChar = tm.tmAveCharWidth;

cyChar = tm.tmHeight + tm.tmExternalLeading;

////////////////////////////////////////////////////

hwndButtonVictorinaPodskazki[9] = CreateWindow("BUTTON", "Подсказка",

WS\_CHILD | WS\_VISIBLE | WS\_GROUP | WS\_TABSTOP | BS\_PUSHBUTTON,

671, // starting x position

250, // starting y position

100, // button width

50, // button height

hWnd, // parent window

(HMENU)29,

hInstance,

//(HINSTANCE)GetWindowLong(hWnd, NULL),

NULL);

//////////////////////////////////////////////////

B = CreateSolidBrush(RGB(0, 0, 255));

SelectObject(hdc, B);

b[0] = Rectangle(hdc,

cxChar + 950, cyChar \*(5) + 96,

130 \* cxChar + 3, ((5) \* cyChar / 2) + 96);

DestroyWindow(hwndButton\_Victorina\_Guestion[9]);

////////////////////////////////////////

if (counter >= 1) {

DestroyWindow(hwndButtonVictorinaPodskazki[9]);

}

/////////////////////////////////////////

DialogBox(hInst, MAKEINTRESOURCE(IDD\_Ten), hwndButtonChose, NumbTen);

DestroyWindow(hwndButtonVictorinaPodskazki[9]);

break;

/////////////////////////////////////////////////////

case 29: {

if (MessageBox(hWnd, "Хотите использовать подсказку ?Uncorrect:", "", MB\_YESNO) == IDYES) {

counter++;

MessageBox(hWnd, "Uncorrect:\nNope\nПтица", "Answer", MB\_OK);

DestroyWindow(hwndButtonVictorinaPodskazki[9]);

}

else {

DestroyWindow(hwndButtonVictorinaPodskazki[9]);

}

break;

}

}

break;

}//end of switch

////////////////////////////////////

break;

}//end of command

case WM\_PAINT:

{

HDC hdc = BeginPaint(hWnd, &ps);

// TODO: Добавьте сюда любой код прорисовки, использующий HDC...

EndPaint(hWnd, &ps);

}

break;

case WM\_DESTROY:

PostQuitMessage(0);

break;

default:

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

}

return 0;

}