Answer on Question #60303, Programming & Computer Science / C++

Question:

write code in win32 c++ to appear dialog box to enter the number and test if the number is prime or not and compute factorial and is the number is perfect or not , this dialog box contain static text name is the text "enter the number" and the side edit box name is box and below of the dialog 3 button:
the first button test is prime or not
and the second button compute the factorial of number
and the third button is perfect number or not

Solution:

```
#define _CRT_SECURE_NO_WARNINGS
#include <windows.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <tchar.h>
#include <cmath>

#define IDC_BTN_1 1
#define IDC_BTN_2 2
#define IDC_BTN_3 3
// Global variables

// The main window class name.static TCHAR szWindowClass[] = _T("win32app");

// The string that appears in the application's title bar.static TCHAR szTitle[] = _T("Win32 Guided
Tour Application");
static TCHAR szWindowClass[] = _T("win32app");
static TCHAR szTitle[] = _T("Win32 Guided Tour Application");
HINSTANCE hInst;
HWND hStatic = 0, hEdit = 0, hBtn1 = 0, hBtn2 = 0, hBtn3 = 0;

// Forward declarations of functions included in this code module:
LRESULT CALLBACK WndProc(HWND, UINT, WPARAM, LPARAM);

int WINAPI WinMain(HINSTANCE hInstance,
       HINSTANCE hPrevInstance,
       LPSTR lpCmdLine,
       int nCmdShow)
{
       WNDCLASSEX 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_APPLICATION));
       wcex.hCursor = LoadCursor(NULL, IDC_ARROW);
       wcex.hbrBackground = (HBRUSH)(COLOR_WINDOW + 1);
```

```
wcex.lpszMenuName = NULL;
wcex.lpszClassName = szWindowClass;
wcex.hIconSm = LoadIcon(wcex.hInstance, MAKEINTRESOURCE(IDI_APPLICATION));

if (!RegisterClassEx(&wcex))
{
	MessageBox(NULL,
		_T("Call to RegisterClassEx failed!"),
		_T("Win32 Guided Tour"),
		NULL);

	return 1;
}

hInst = hInstance; // Store instance handle in our global variable

                                // The parameters to CreateWindow explained:
                                // szWindowClass: the name of the application
                                // szTitle: the text that appears in the title bar
                                // WS_OVERLAPPEDWINDOW: the type of window to create
                                // CW_USEDEFAULT, CW_USEDEFAULT: initial position (x, y)
                                // 500, 100: initial size (width, length)
                                // NULL: the parent of this window
                                // NULL: this application does not have a menu bar
                                // hInstance: the first parameter from WinMain
                                // NULL: not used in this application
HWND hWnd = CreateWindow(
	szWindowClass,
	szTitle,
	WS_OVERLAPPEDWINDOW,
	CW_USEDEFAULT, CW_USEDEFAULT,
	380, 300,
	NULL,
	NULL,
	hInstance,
	NULL
	);

if (!hWnd)
{
	MessageBox(NULL,
		_T("Call to CreateWindow failed!"),
		_T("Win32 Guided Tour"),
		NULL);

	return 1;
}



ShowWindow(hWnd,
	nCmdShow);
UpdateWindow(hWnd);

// Main message loop:
MSG msg;
while (GetMessage(&msg, NULL, 0, 0))
{
	TranslateMessage(&msg);
	DispatchMessage(&msg);
}
```

```
	return (int)msg.wParam;
}

BOOL IsPrime(LONG value) {
	if (value == 2 || value == 3 || value == 5 || value == 7) return TRUE;
	if (value % 2 == 0 || value == 1) return FALSE;

	LONG n = sqrt(value) + 1;
	for (int i = 3; i <= n; i += 2) {
		if (value % i == 0) return FALSE;
	}
	return TRUE;
}

LONG64 Factorial(LONG64 value) {
	if (value < 0) return -1;
	if (value == 0 || value == 1) return 1;
	return value * Factorial(value - 1);
}

BOOL IsPerfect(LONG value) {
	if (value < 0) return FALSE;
	LONG sum = 0;
	for (int i = 1; i < value; ++i) {
		if (value % i == 0) sum += i;
	}
	return value == sum;
}

LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam)
{
	PAINTSTRUCT ps;
	HDC hdc;
	TCHAR greeting[] = _T("Hello, World!");
	HINSTANCE hInstance = hInst;


	switch (message)
	{
	case WM_CREATE:
	{
		hStatic = CreateWindow(L"Static", NULL, WS_CHILD | WS_VISIBLE | ES_LEFT, 10, 10, 120,
30, hWnd, NULL, hInstance, NULL);
		SetWindowText(hStatic, L"Enter the number: ");
		hEdit = CreateWindow(L"Edit", NULL, WS_CHILD | WS_VISIBLE | WS_BORDER | ES_LEFT, 140,
10, 140, 30, hWnd, NULL, hInstance, NULL);

		hBtn1 = CreateWindow(L"Button", NULL, WS_CHILD | WS_VISIBLE | WS_BORDER | ES_LEFT, 140,
50, 140, 30, hWnd, (HMENU)IDC_BTN_1, hInstance, (LPVOID*)IDC_BTN_1);
		hBtn2 = CreateWindow(L"Button", NULL, WS_CHILD | WS_VISIBLE | WS_BORDER | ES_LEFT, 140,
90, 140, 30, hWnd, (HMENU)IDC_BTN_2, hInstance, (LPVOID*)IDC_BTN_2);
		hBtn3 = CreateWindow(L"Button", NULL, WS_CHILD | WS_VISIBLE | WS_BORDER | ES_LEFT, 140,
130, 140, 30, hWnd, (HMENU)IDC_BTN_3, hInstance, NULL);
		SetWindowText(hBtn1, L"Prime test");
		SetWindowText(hBtn2, L"Compute factorial");
		SetWindowText(hBtn3, L"Perfect test");
	}
		break;
	case WM_COMMAND:
	{
```

```
			int ControlID = LOWORD(wParam);
			int MessageType = HIWORD(wParam);
			char buff[256] = { 0 };
			GetWindowTextA(hEdit, (LPSTR)buff, 256);
			LONG value = atoi(buff);
			if (ControlID == IDC_BTN_1) {
				if (IsPrime(value)) {
					MessageBox(hWnd, L"Is Prime", L"Prime test", MB_OK);
				}
				else {
					MessageBox(hWnd, L"Is NOT Prime", L"Prime test", MB_OK);
				}
			}
			else if (ControlID == IDC_BTN_2) {
				LONG64 result = Factorial(value);
				sprintf(buff, "%lld", result);
				MessageBoxA(hWnd, (LPSTR)buff, "Factorial", MB_OK);
			}
			else if (ControlID == IDC_BTN_3) {
				if (IsPerfect(value)) {
					MessageBox(hWnd, L"Is Perfect", L"Perfect test", MB_OK);
				}
				else {
					MessageBox(hWnd, L"Is NOT Perfect", L"Perfect test", MB_OK);
				}
			}

			break;
		}
		case WM_DESTROY:
			PostQuitMessage(0);
			break;
		default:
			return DefWindowProc(hWnd, message, wParam, lParam);
			break;
		}

		return 0;
}
```