

```
#include <windows.h>
#include <windowsx.h>
```

```
static char szWndClassName[] = "CHelloWorld";
```

```
/*
 *
 * Function:  LRESULT CALLBACK WndProc(HWND, UINT, WPARAM, LPARAM)
 *
 * Aufgabe:   Verarbeitet die Nachrichten einer Fensterklasse
 *
 */
LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam)
{
    switch (message)
    {
        case WM_PAINT:
        {
            HDC hdc;
            PAINTSTRUCT ps;
            hdc = BeginPaint(hWnd, &ps);
            TextOut(hdc, 100, 100, "Hello World", 11);
            EndPaint(hWnd, &ps);
            return 0;
        }
        case WM_DESTROY:
        {
            PostQuitMessage (0);
            return 0;
        }
        default:
            return DefWindowProc (hWnd, message, wParam, lParam) ;
    }
}
```

```
/*
 *
 * Function:  int InitApplication (HINSTANCE)
 *
 * Aufgabe:   Initialisiert eine Windowsapplikation
 *
 * Kommentar: Legt eine Fensterklasse an.
 *
 */
```

```
static int InitApplication(HINSTANCE hInstCurr)
{
    WNDCLASSEX wc; /* window class */
    wc.cbSize      = sizeof(WNDCLASSEX);
    wc.style       = CS_HREDRAW | CS_VREDRAW ;
    wc.lpfnWndProc = (WNDPROC)WndProc ;
    wc.cbClsExtra  = 0 ;
    wc.cbWndExtra  = 0 ;
    wc.hInstance   = hInstCurr ;
}
```

```

    wc.hIcon          = LoadIcon(NULL, IDI_WINLOGO);
    wc.hIconSm        = NULL;
    wc.hCursor        = LoadCursor(NULL, IDC_ARROW);
    wc.hbrBackground = (HBRUSH)(COLOR_WINDOW + 1);
    wc.lpszMenuName   = NULL;
    wc.lpszClassName = szWndClassName;
    return RegisterClassEx(&wc);
}

```

```

/*****
 *
 * Function:  int InitInstance (HINSTANCE, int)
 *
 * Aufgabe:   Initialisiert die aktuelle Instanz einer Windowsapplikation
 *
 * Kommentar: Öffnet ein Fenster
 *
 *****/

```

```

static int InitInstance(HINSTANCE hInstCurr, int nCmdShow)
{
    HWND hWnd;
    hWnd = CreateWindow(szWndClassName, // Class Name: See top of file
        "Hello World",                // Title of window
        WS_OVERLAPPEDWINDOW,          // Type of window: Standard (+Title,
Minimize, Maximize, Resizable)
        20,                            // x position
        40,                            // y position
        600,                           // Width
        400,                           // Height
        NULL,                          // No parent
        NULL,                          // No menu
        hInstCurr,                    // For this instance
        NULL);                         // No parameters
    if (hWnd)
    {
        ShowWindow(hWnd, nCmdShow);
        UpdateWindow(hWnd);
        return TRUE;
    }
    return FALSE;
}

```

```

/*****
 *
 * Function:  int WINAPI WinMain (HINSTANCE, HINSTANCE, LPSTR, int)
 *
 * Aufgabe:   Hauptprogramm einer Windowsapplikation
 *
 * Kommentar: Legt eine Fensterklasse an, öffnet ein Fenster,
 *            verarbeitet Messages.
 *
 *****/

```

```

int WINAPI WinMain (HINSTANCE hInstCurr, HINSTANCE hInstPrev, LPSTR lpszCmd, int
nCmdShow)

```

```
{
MSG msg ;
if (!hInstPrev)
    if (!InitApplication(hInstCurr))
        return 0;
if (!InitInstance(hInstCurr, nCmdShow))
    return 0;
while (GetMessage (&msg, NULL, 0, 0)>0)
{
    TranslateMessage(&msg);
    DispatchMessage(&msg);
}
return msg.wParam;
}
```