

LAMPIRAN D

Listing Program Software “Download list builder”

Form utama program (frmMain)

D-1

```

//-----
#include <vcl.h>
#pragma hdrstop

#include "code_frmMain.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TfrmMain *frmMain;
int count = 0;
int select = -1;
String file_name;
String iniPath = ExtractFilePath(Application->ExeName) + "\\list_builder.ini";
//-----
__fastcall TfrmMain::TfrmMain(TComponent* Owner)
    : TForm(Owner)
{
}
//-----
void __fastcall TfrmMain::btnAddClick(TObject *Sender)
{
    if(txtURL->Text != "") {
        // Add URL into list
        lbURL->Items->Add(txtURL->Text);
        count++;
        txtURL->Clear();
        txtURL->SetFocus();
    } else {
        MessageBox(Handle,"Please type URL before
add!","Error",MB_OK|MB_ICONEXCLAMATION);
    }
}
//-----
void __fastcall TfrmMain::btnApplyClick(TObject *Sender)
{
    if(MessageBox(Handle,"Update current item?","Ask",MB_YESNO|MB_ICONQUESTION) ==
ID_YES) {
        lbURL->Items->Strings[select] = txtURL->Text;
    }
}
//-----
void __fastcall TfrmMain::btnDeleteClick(TObject *Sender)
{
    if(lbURL->ItemIndex != -1) {
        if(MessageBox(Handle,"Delete selected item from the
list?","Ask",MB_YESNO|MB_ICONQUESTION) == ID_YES) {
            lbURL->DeleteSelected();
            count--;
        }
    } else {
        MessageBox(Handle,"You haven't choose anything!\nPick up one item.", "Choose
first",MB_OK|MB_ICONEXCLAMATION);
    }
}
//-----
void __fastcall TfrmMain::lbURLClick(TObject *Sender)
{
    select = lbURL->ItemIndex;
    txtURL->Text = lbURL->Items->Strings[select];
}
//-----
void __fastcall TfrmMain::btnOpenListClick(TObject *Sender)
{
    String URL;
    String temp;

    if(OpenDlg->Execute()) {
        lbURL->Clear();
    }
}

```

```

file_name = OpenFileDialog->FileName;
char * retStr;
retStr = (char*) calloc( 3001, sizeof(char));
memset(retStr,0,3000);

// Load selected download list file
// Get the count
GetPrivateProfileString(
"Properties","Count","0",retStr,3000,file_name.c_str());
count = StrToInt(String(retStr));

// Get the list
for(int i = 0 ; i < count ; i++) {
    memset(retStr,0,3000);
    temp = "Addr" + IntToStr(i);
    GetPrivateProfileString(
"List",temp.c_str(),"",retStr,3000,file_name.c_str());
    URL = String(retStr);
    lbURL->Items->Add(URL);
}

free(retStr);
}
}
//-----
void __fastcall TfrmMain::btnSaveListAsClick(TObject *Sender)
{
    String URL;
    String temp;

    if(SaveDlg->Execute()) {
        // Save list to selected download list file
        file_name = SaveDlg->FileName;
        if(count != 0) {

            WritePrivateProfileString(
"Properties","Count",IntToStr(count).c_str(),file_name.c_str());

            for(int i = 0 ; i < count ; i++ ) {
                temp = "Addr" + IntToStr(i);
                URL = lbURL->Items->Strings[i];

                //WritePrivateProfileString(
"List","addr1","http://",file_name.c_str());
                WritePrivateProfileString( "List", temp.c_str(), URL.c_str(),
file_name.c_str());
            }

        } else {
            MessageBox(Handle,"You haven't added anything yet!","Couldn't
save!",MB_OK|MB_ICONEXCLAMATION);
        }
    }
}
//-----
void __fastcall TfrmMain::btnSaveListClick(TObject *Sender)
{
    String URL;
    String temp;

    if(file_name != "") {

        if(count != 0) {

            WritePrivateProfileString(
"Properties","Count",IntToStr(count).c_str(),file_name.c_str());

            for(int i = 0 ; i < count ; i++ ) {
                temp = "Addr" + IntToStr(i);
                URL = lbURL->Items->Strings[i];

```

```

        //WritePrivateProfileString(
"List", "addr1", "http://", file_name.c_str());
        WritePrivateProfileString( "List", temp.c_str(), URL.c_str(),
file_name.c_str());
    }

    } else {
        MessageBox(Handle, "You haven't added anything yet!", "Couldn't
save!", MB_OK|MB_ICONEXCLAMATION);
    }

    } else {
        btnSaveListAsClick(btnSaveList);
    }
}
//-----
void __fastcall TfrmMain::btnNewListClick(TObject *Sender)
{
    if(count == 0) {

        lbURL->Clear();
        count = 0;
        file_name = "";
        txtURL->Clear();

    } else {
        if(MessageBox(Handle, "Current list isn't empty!\nProcess with creating new
list?", "Warning", MB_YESNO|MB_ICONEXCLAMATION) == ID_YES) {
            lbURL->Clear();
            count = 0;
            file_name = "";
            txtURL->Clear();
        }
    }
}
//-----

bool TfrmMain::FileExist(String filename)
{
    // Check for existing file.
    // if not exist, then return false.
    // if exist, then return true.
    bool result = false;
    HANDLE exist;

    exist =
CreateFile(filename.c_str(), 0, FILE_SHARE_READ|FILE_SHARE_WRITE, 0, OPEN_EXISTING, FILE_
E_ATTRIBUTE_NORMAL, 0);
    if(exist != INVALID_HANDLE_VALUE) {
        result = true;
    } else {
        result = false;
    }
    CloseHandle(exist);

    return result;
}
//-----

void __fastcall TfrmMain::FormClose(TObject *Sender, TCloseAction &Action)
{
    // Save this save file on configuration.ini
    WritePrivateProfileString( "Options", "AutoOpen", BoolToStr(chkAutoOpen-
>Checked).c_str(), iniPath.c_str());
    if(file_name != "") {
        WritePrivateProfileString( "Options", "File", file_name.c_str(),
iniPath.c_str());
    }
}

```

```

}
//-----

void __fastcall TfrmMain::FormShow(TObject *Sender)
{
    // Load configuration.ini

    char * retStr2;
    retStr2 = (char*) calloc(261,sizeof(char));
    memset(retStr2,0,261);

    GetPrivateProfileString( "Options", "AutoOpen", "0",retStr2,261,
iniPath.c_str());
    chkAutoOpen->Checked = StrToBool(String(retStr2));
    memset(retStr2,0,261);
    GetPrivateProfileString( "Options", "File", "",retStr2,261, iniPath.c_str());
    file_name = String(retStr2);

    free(retStr2);

    if((FileExist(file_name)) == true && (chkAutoOpen->Checked == true)) {
        String URL;
        String temp;

        lbURL->Clear();
        char * retStr;
        retStr = (char*) calloc( 3001, sizeof(char));
        memset(retStr,0,3000);

        // Load selected download list file
        // Get the count
        GetPrivateProfileString(
"Properties", "Count", "0",retStr,3000,file_name.c_str());
        count = StrToInt(String(retStr));

        // Get the list
        for(int i = 0 ; i < count ; i++) {
            memset(retStr,0,3000);
            temp = "Addr" + IntToStr(i);
            GetPrivateProfileString(
"List",temp.c_str(),"",retStr,3000,file_name.c_str());
            URL = String(retStr);
            lbURL->Items->Add(URL);
        }

        free(retStr);
    }
}
//-----

```