

VB*Build Version 1.1 For Windows

Shareware Version

(C) Copyright Marc Brewerton 1993-1994. All Rights Reserved

This program is Shareware, you may evaluate it, for a period of no more than 30 days. After this period, you must either register (read REGISTER.WRI) or remove it from your system. Failure to comply with this condition is a violation of the law. Please register if you use this program for more than the trial period.

There are benefits from registering : - Program Update. Plus New Features.

PLEASE NOTE : The VBRUN300.DLL runtime library is required for this program. The DLL's THREED.VBX, CMDIALOG.VBX and COMMDLG.DLL are provided with this program.

****You Should Place All The Above Files Into Your Windows System Directory. If You Do Not Wish To Do This Then Place Them Somewhere On Your Path. ****

Program Overview.

As I have spent time programming in VB, I have often wished I had spent a much greater period of time designing the program before I started. I was traditionally taught how to program, this included correct use of Flowcharts and Algorithms. Anyhow, many people I have spoken to use the Visual part of VB to act as their flowchart, design the Visual component and attach the code to resulting control click event.

I believe, however that the majority of code should exist in sub-routines and functions separate to the visual part of the VB and should merely be called from the control events. It is much clearer this way and tends to be more portable to other applications. i.e. write one file opening routine and you will use it elsewhere (if it is easy to call up again / not attached to a control.) Without prior planning, code seldom has its own routines/functions and instead becomes attached to controls too easily.

VB*Build has been created so that you can think about a number of forms, sub-routines and functions before programming, giving them a name/description. This data can be saved as a template and converted into a bas module and forms for VB to load. The program does not presume to get it right first time, but give you a chance to think a bit more.

Installation Instructions.

PLEASE NOTE : The VBRUN300.DLL runtime library is required for this program.

The DLL's THREEED.VBX, CMDIALOG.VBX and COMMDLG.DLL are provided with this program. These VBX and DLL files should be placed in your Windows System directory. You should then create separate directory for VB*Build e.g C:\VBBUILD and copy into it, the remaining files.

Program Usage.

The program works very simply, you have 20 free form/code sections and you can make them either a function, sub routine or form. To each of these sections, you must give a name and a description. See below for a breakdown of actions.

Adding Forms/Routines/Functions.

Simply click on the corresponding command button on the top right of the window. And you will be presented with a window asking for the name and description. Use 'Cancel' to abandon the operation or 'Done' when complete. The new function/routine/form will be placed in the lowest available position.

Removing Forms/Routines/Functions.

Select the function/routine/form you wish to remove and drag it over the 'Remove' Button on the right. It will be removed and all above it will be shifted down.

Editing Forms/Routines/Functions.

Select the function/routine/form you wish to remove and drag it over the 'Edit' Button on the right. You will be presented with the same window you had earlier and asked for the changes. Click 'Cancel' for no change and 'Done' to make the changes.

Saving To A Template.

You can save your work in a template format which means you can design your ideas with no particular program in mind thus creating a general purpose module of empty (or with

attached text ~ Only with Registered Versions) subs and functions. This can then be used in all of our programs that deal with that subject but don't necessarily use the same code. e.g. File Saving/Opening/Handling Format or Graphing/Drawing/Interpreting/Evaluating.

Opening To A Template.

Retrieve your pre-saved work for distinct changes at a later date.

Converting To Project Files.

When converting to project format, you will be asked to choose a directory for the files, the program will name the code module the following way, CODE???.bas where ??? is a three digit number. VB*Build will look in your chosen directory and number the file at the next available number. The forms are automatically named after the first eight letters of the name you given them. All you have to do then is go into VB and use the 'Add File' option to merge them all together. The functions and routines you defined will have been made into real functions/routines with your descriptions as Remark statements.

Printing Your Template.

This is a basic listing of your template detailing names and descriptions.

Print Setup

Standard Windows Print Setup. Set your printer before printing your template.

Data Format.

The template data is stored in ASCII format and therefore can be read EASILY by other programs.

Worked Example :

You want to write a program to read a file, and display in a text box.

What Do We Need :-

FORMS

MainForm For the text box, menu bar, load and quit buttons.

SUBS

GetFileName To Get Filename To Open. Through CMDIALOG.

ReadFile To Open File, read until end and place data into text box.

ErrorReport Report Some Kind Of Error/File IO or if FUNCTION ChkFile Says Bad File.

FUNCTIONS

ChkFile To Look For File. Open It, If it Exists/Check That It Has Some Size.

Note : See Example.Tpl for the above worked example in code.

Registered Program Features.

- Ability to specify that a function/routine contains an error routine, these will be added automatically with dummy labels.
- Ability to add preset text code into your defined functions/routines.
- 40 Form/Code Sections rather than the 20 in the Shareware Version.

(See **REGISTER.WRI** for further information.)

	<u>Shareware Version.</u>	<u>Registered Version.</u>
<u>Program Limitations :</u>	20 Form/Code Sections.	40 Form/Code Sections.

Disclaimer.

This Software Is Provided AS IS And Any Express Or Implied Warranties, Including, But Not Limited To, The Implied Warranties Of Merchantability And Fitness For A Particular Purpose Are Disclaimed. In No Event Shall The Author Be Liable For Any Direct, Indirect, Incidental, Special, Exemplary, Or Consequential Damages (Including, But Not Limited To, Procurement Of Substitute Goods Or Services; Loss Of Use, Data, Or Profits; Or Business Interruption) However Caused And On Any Theory Of Liability, Whether In Contract, Strict Liability, Or Tort (Including Negligence Or Otherwise) Arising In Any Way Out Of The Use Of This Software, Even If Advised Of The Possibility Of Such Damage.

History.

Version 1.0 - Created 26-1-94. Written In Visual Basic Version 3.0

Version 1.1 - Fixed Bug With Carriage Returns In Descriptions.

Bugs & Suggestions.

Please report any bugs, problems or suggestions and I will be pleased to help :

Marc Brewerton

"VB*Build For Windows Version 1.1"

2, St.John's Meadow

Blindley Heath

Surrey. RH7 6JU

United Kingdom

email address : <me90mab@brunel.ac.uk>

Note

Windows and Visual Basic are Trademarks and Copyright To The Microsoft Corporation.