

Management

Projects Symbols Files FSymbols Resources

- Resources
 - wxSmithNotepad
 - wxFrame
 - wxSmithNotepadFrame
 - wxFrame
 - wxTextCtrl: TextCtrl1
 - wxButton: Button1
 - Tools
 - wxMenuBar: MenuBar1
 - wxStatusBar: StatusBar1
 - wxFileDialog: FileDialog1

wxSmithNotepadMain.cpp wxSmithNotepadframe.wxs

Text

FileOpen

Add TextCtrl

Text	Text
Max Length	0
Var name	TextCtrl1
Is member	<input checked="" type="checkbox"/>
Identifier	ID_TEXTCTRL1
Class name	wxTextCtrl
Default pos	<input type="checkbox"/>
X	0
Y	0
Pos in dialog units	<input type="checkbox"/>
Default size	<input type="checkbox"/>
Width	400
Height	264
Size in dialog units	<input type="checkbox"/>

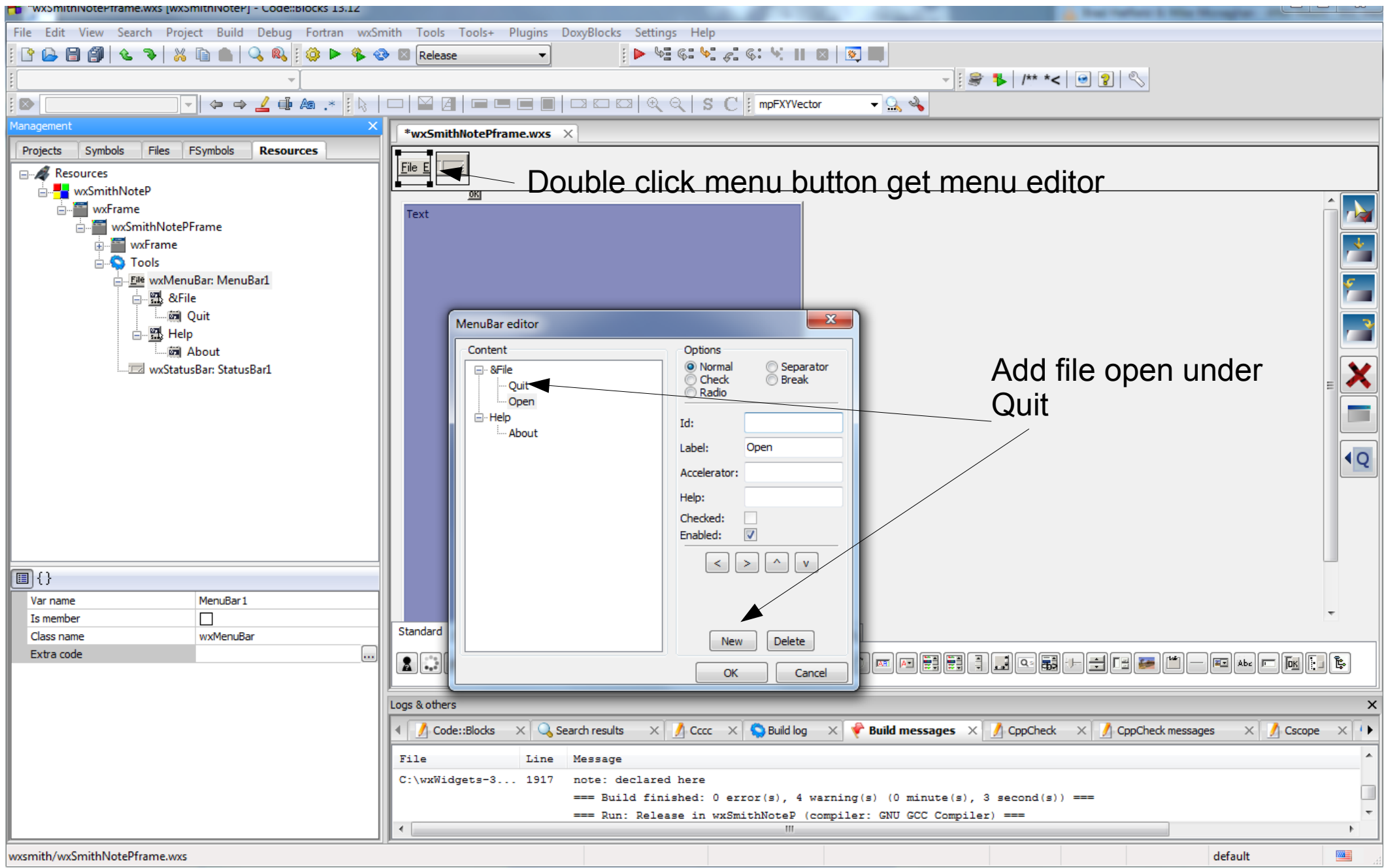
Standard Advanced Aui Contrib Dialogs KWIC Layout Led MathPlot Tools

Logs & others

- CppCheck messages
- Cscope
- Debugger
- DoxyBlocks
- Closed files list
- Thread search
- Fortran

File	Line	Scope	Text

Ready



Double click menu button get menu editor

Add file open under Quit

```
File | Line | Message
C:\wxWidgets-3... | 1917 | note: declared here
                    | | === Build finished: 0 error(s), 4 warning(s) (0 minute(s), 3 second(s)) ===
                    | | === Run: Release in wxSmithNoteP (compiler: GNU GCC Compiler) ===
```

The screenshot shows the Code::Blocks IDE with a project named 'wxSmithNoteP'. The 'Resources' tree on the left shows a menu structure under 'File wxMenuBar: MenuBar1' with items '&File', 'Quit', 'Open', and 'Help'. A 'New handler' dialog box is open, prompting 'Enter name for new handler:' with 'OnMenuItemOpen' entered. The dialog has 'OK' and 'Cancel' buttons. A 'Logs & others' window at the bottom shows build messages: 'note: declared here', '=== Build finished: 0 error(s), 4 warning(s) (0 minute(s), 3 second(s)) ===', and '=== Run: Release in wxSmithNoteP (compiler: GNU GCC Compiler) ==='. Three arrows labeled 1, 2, and 3 point to the 'E MENU' menu item, the 'New handler' dialog, and the text input field in the dialog, respectively.

File	Line	Message
C:\wxWidgets-3...	1917	note: declared here
=== Build finished: 0 error(s), 4 warning(s) (0 minute(s), 3 second(s)) ===		
=== Run: Release in wxSmithNoteP (compiler: GNU GCC Compiler) ===		

1

2

3

*wxSmithNotePframe.wx [wxSmithNoteP] - Code::Blocks 13.12

File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

wxSmithNotePFrame:: OnMenuItemOpen(wxCommandEvent& event) : void

mpFXYVector

Management

Projects Symbols Files FSymbols Resources

Resources

- Resources
 - wxSmithNoteP
 - wxFrame
 - wxSmithNotePFrame
 - wxFrame
 - Tools
 - File wxMenuBar: MenuBar1
 - &File
 - Quit
 - Open
 - Help
 - wxStatusBar: StatusBar1
 - wxFileDialog: FileDialog1

Message Select file

Default directory

Default file

Wildcard

Var name FileDialog1

Is member

Class name wxFileDialog

Extra code

Style wxFD_DEFAULT_STYLE

Extra style

Standard Advanced Aui Contrib Dialogs KWIC Layout Led MathPlot Tools

Logs & others

Code::Blocks Search results Cccc Build log Build messages CppCheck CppCheck messages Cscope

File	Line	Message
C:\wxWidgets-3...	1917	note: declared here
=== Build finished: 0 error(s), 4 warning(s) (0 minute(s), 3 second(s)) ===		
=== Run: Release in wxSmithNoteP (compiler: GNU GCC Compiler) ===		

wxsmith/wxSmithNotePframe.wx default

Add an instance of a File Dialog

added to open/load a selected file.

In the code for the OnOpen member function, a wxFileDialog is instantiated which pops up a file selector box.

See the following descriptions:

wxFileDialog overview:

http://docs.wxwidgets.org/stable/wx_commondialogoverview.html#wxfiledialogoverview

and wxFileDialog class description:

http://docs.wxwidgets.org/stable/wx_wxfiledialog.html#wxfiledialog

Specifically, in the wxEditorFrame::OnOpen function the following will be done:

1. Create an instance of the wxFileDialog
2. Cause the instance to 'pop-up' (ShowModal)
3. Check if the response is 'OK'
4. Load the selected file into the wxTextCtrl inside the wxFrame

This is the code which will do those 4 items. The intention is that you study these 4 lines of code and develop an understanding of what they are doing.

1. Create an instance of the wxFileDialog

```
wxFileDialog *openDialog = new wxFileDialog(this, wxT("Choose a file"), wxT(""), wxT(""),  
    wxT("Text Files (*.txt)|*.txt|C++ Files (*.cpp)|*.cpp|Header Files (*.h)|*.h"),  
    wxFD_OPEN );
```

2. Cause the instance to 'pop-up' (ShowModal)

```
int response = openDialog->ShowModal(); //get response from the dialog
```

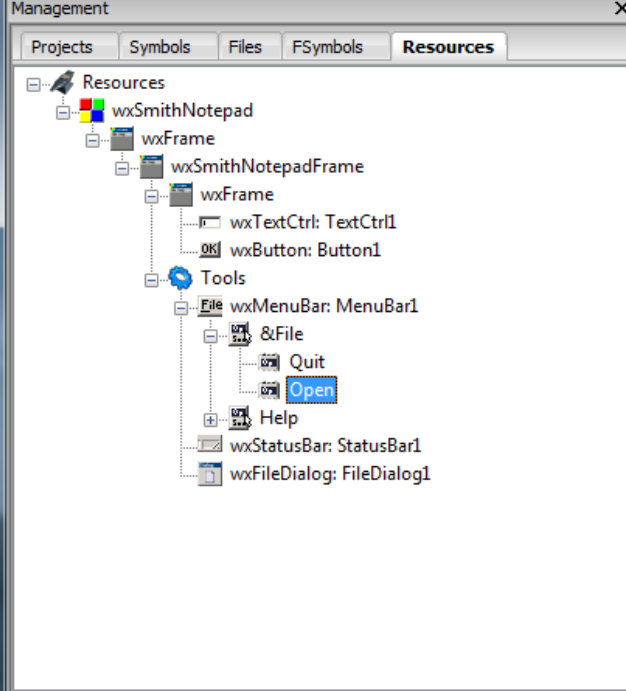
3. Check if the response is 'OK'

4. Load the selected file into the wxTextCtrl inside the wxFrame

```
if(response == wxID_OK)  
{ //if response ok, then load contents into textControl  
    this->textControl->LoadFile(openDialog->GetPath());  
}
```

Build and run the finished program. Turn in a screen copy **with the file open dialog activated.**

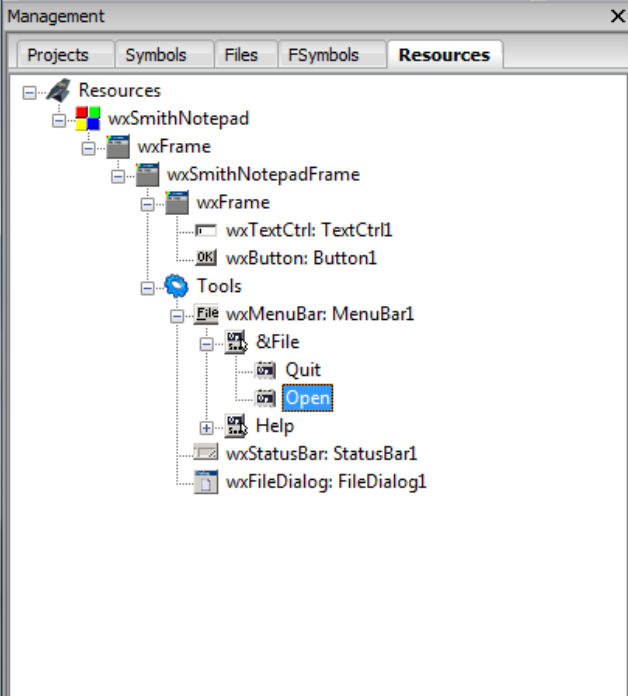
Put this code from PW4 into the Open event handler



EVT_MENU OnMenuOpen

Menu File Open And Button File Open From HW4

```
wxSmithNotepadMain.cpp wxSmithNotepadframe.wxw
102 }
103
104 void wxSmithNotepadFrame::OnQuit (wxCommandEvent& event)
105 {
106     Close();
107 }
108
109 void wxSmithNotepadFrame::OnAbout (wxCommandEvent& event)
110 {
111     wxString msg = wxbuildinfo(long_f);
112     wxMessageBox(msg, _("Welcome to..."));
113 }
114
115 void wxSmithNotepadFrame::OnButton1Click (wxCommandEvent& event)
116 {
117     int response = FileDialog1->ShowModal(); //get response from the dialog
118     if(response == wxID_OK)
119     { //if response ok, then load contents into textControl
120         this->TextCtrl1->LoadFile(FileDialog1->GetPath());
121     }
122 }
123
124
125 void wxSmithNotepadFrame::OnMenuOpen (wxCommandEvent& event)
126 {
127     int response = FileDialog1->ShowModal(); //get response from the dialog
128     if(response == wxID_OK)
129     { //if response ok, then load contents into textControl
130         this->TextCtrl1->LoadFile(FileDialog1->GetPath());
131     }
132 }
133
```



EVT_MENU OnMenuOpen

```
wxSmithNotepadMain.cpp wxSmithNotepadframe.wxw  
102 }  
103  
104 void wxSmithNotepadFrame::OnQuit(wxCommandEvent& event)  
105 {  
106     Close();  
107 }  
108  
109 void wxSmithNotepadFrame::OnAbout(wxCommandEvent& event)  
110 {  
111     wxString msg = wxbuildinfo(long_f);  
112     wxMessageBox(msg, _("Welcome to..."));  
113 }  
114  
115 void wxSmithNotepadFrame::OnButton1Click(wxCommandEvent& event)  
116 {  
117     int response = FileDialog1->ShowModal(); //get response from the dialog  
118     if(response == wxID_OK)  
119     { //if response ok, then load contents into textControl  
120         this->TextCtrl1->LoadFile(FileDialog1->GetPath());  
121     }  
122 }  
123  
124  
125 void wxSmithNotepadFrame::OnMenuOpen(wxCommandEvent& event)  
126 {  
127     int response = FileDialog1->ShowModal(); //get response from the dialog  
128     if(response == wxID_OK)  
129     { //if response ok, then load contents into textControl  
130         this->TextCtrl1->LoadFile(FileDialog1->GetPath());  
131     }  
132 }  
133
```

Logs & others
Code::Blocks Search results Cccc Build log Build messages CppCheck CppCheck mess

Make the TextCtrl style MultiLine and Rich Text

The screenshot shows the Code::Blocks IDE with the following components:

- Management Panel:** Shows a tree view of resources for `wxSmithNoteP`, including `wxFrame`, `wxTextCtrl: TextCtrl1`, `wxMenuBar: MenuBar1`, `wxStatusBar: StatusBar1`, and `wxFileDialog: FileDialog1`.
- Code Editor:** Displays the `wxSmithNotePMain.cpp` file with the following code:

```
59 //(*Initialize(wxSmithNotePFrame)
60 wxMenuItem* MenuItem2;
61 wxMenuItem* MenuItem1;
62 wxMenu* Menu1;
63 wxMenuBar* MenuBar1;
64 wxMenu* Menu2;
65
66 Create(parent, wxID_ANY, wxEmptyString, wxDefaultPosition, wxDefaultSize, wxDEFAULT_FRAME_STYLE, T("wxID_
67 TextCtrl1 = new wxTextCtrl(this, ID_TEXTCTRL1, _("Text"), wxPoint(152,144), wxDefaultSize, wxTE_MULTILINE|
68 MenuBar1 = new wxMenuBar();
69 Menu1 = new wxMenu();
70 MenuItem1 = new wxMenuItem(Menu1, idMenuQuit, _("Quit\tAlt-F4"), _("Quit the application"), wxITEM_NORMAL);
71 Menu1->Append(MenuItem1);
72 MenuItem3 = new wxMenuItem(Menu1, idMenuOpen, _("Open"), wxEmptyString, wxITEM_NORMAL);
73 Menu1->Append(MenuItem3);
74 MenuBar1->Append(Menu1, _("&File"));
75 Menu2 = new wxMenu();
76 MenuItem2 = new wxMenuItem(Menu2, idMenuAbout, _("About\tF1"), _("Show info about this application"), wxITEM
77 Menu2->Append(MenuItem2);
78 MenuBar1->Append(Menu2, _("Help"));
79 SetMenuBar(MenuBar1);
80 StatusBar1 = new wxStatusBar(this, ID_STATUSBAR1, 0, T("ID_STATUSBAR1"));
81 int __vxStatusBarWidths_1[1] = { -10 };
82 int __vxStatusBarStyles_1[1] = { wxSB_NORMAL };
83 StatusBar1->SetFieldsCount(1, __vxStatusBarWidths_1);
84 StatusBar1->SetStatusStyles(1, __vxStatusBarStyles_1);
85 SetStatusBar(StatusBar1);
86 SetStatusText( "Welcome to wxWidgets!" );
87 FileDialog1 = new wxFileDialog(this, _("Select file"), wxEmptyString, wxEmptyString, wxFileSelectorDefault
88
89 Connect(idMenuQuit,wx.EVT_COMMAND_MENU_SELECTED,(wxObjectEventFunction)&wxSmithNotePFrame::OnQuit);
```
- Style Settings:** A table showing the configuration for the `wxTextCtrl` style:

Style	Value
<code>wxTE_NO_VSCROLL</code>	<input type="checkbox"/>
<code>wxTE_AUTO_SCROLL</code>	<input type="checkbox"/>
<code>wxTE_PROCESS_ENTER</code>	<input type="checkbox"/>
<code>wxTE_PROCESS_TAB</code>	<input type="checkbox"/>
<code>wxTE_MULTILINE</code>	<input checked="" type="checkbox"/>
<code>wxTE_PASSWORD</code>	<input type="checkbox"/>
<code>wxTE_READONLY</code>	<input type="checkbox"/>
<code>wxTE_RICH</code>	<input checked="" type="checkbox"/>
<code>wxTE_RICH2</code>	<input type="checkbox"/>
<code>wxTE_AUTO_URL</code>	<input type="checkbox"/>
<code>wxTE_NOHIDESEL</code>	<input type="checkbox"/>
<code>wxTE_LEFT</code>	<input type="checkbox"/>

- Logs & others:** Shows build messages and a note: `note: declared here` at line 1917.
- Status Bar:** Displays the current file path, window title, and cursor position: `C:\Users\watsonh\Documents\CodeBlocks\wxSmithNoteP\wxSmithNotePMain.cpp`, `Windows (CR+LF)`, `WINDOWS-1252`, `Line 117, Column 21`.

Add title to the wxFrame

The screenshot shows the Code::Blocks IDE with the following components:

- Management Panel:** Shows a tree view of the project structure. The selected item is `wxFrame` under `wxSmithNotePFrame`.
- Properties Window:** Located at the bottom left, it shows the following properties for the selected `wxFrame`:

Property	Value
Title	MyNotePad
Centered	<input type="checkbox"/>
Icon	Click to add
Default pos	<input checked="" type="checkbox"/>
X	-1
Y	-1
Pos in dialog units	<input type="checkbox"/>
Default size	<input checked="" type="checkbox"/>
Width	-1
Height	-1
Size in dialog units	<input type="checkbox"/>
Enabled	<input checked="" type="checkbox"/>
Focused	<input type="checkbox"/>
Hidden	<input type="checkbox"/>
- Code Editor:** Displays the `wxSmithNotePFrame.wxs` file with the following code:

```
59 //(*Initialize(wxSmithNotePFrame)
60 wxMenuItem* MenuItem2;
61 wxMenuItem* MenuItem1;
62 wxMenu* Menu1;
63 wxMenuBar* MenuBar1;
64 wxMenu* Menu2;
65
66 Create(parent, wxID_ANY, wxEmptyString, wxDefaultPosition, wxDefaultSize, wxDEFAULT_FRAME_STYLE, _T("wxID_
67 TextCtrl1 = new wxTextCtrl(this, ID_TEXTCTRL1, _("Text"), wxPoint(152,144), wxDefaultSize, wxTE_MULTILINE|
68 MenuBar1 = new wxMenuBar();
69 Menu1 = new wxMenu();
70 MenuItem1 = new wxMenuItem(Menu1, idMenuQuit, _("Quit\tAlt-F4"), _("Quit the application"), wxITEM_NORMAL);
71 Menu1->Append(MenuItem1);
72 MenuItem3 = new wxMenuItem(Menu1, idMenuOpen, _("Open"), wxEmptyString, wxITEM_NORMAL);
73 Menu1->Append(MenuItem3);
74 MenuBar1->Append(Menu1, _("&File"));
75 Menu2 = new wxMenu();
76 MenuItem2 = new wxMenuItem(Menu2, idMenuAbout, _("About\tF1"), _("Show info about this application"), wxIT
77 Menu2->Append(MenuItem2);
78 MenuBar1->Append(Menu2, _("Help"));
79 SetMenuBar(MenuBar1);
80 StatusBar1 = new wxStatusBar(this, ID_STATUSBAR1, 0, _T("ID_STATUSBAR1"));
81 int __wxStatusBarWidths_1[1] = { -10 };
82 int __wxStatusBarStyles_1[1] = { wxSB_NORMAL };
83 StatusBar1->SetFieldsCount(1, __wxStatusBarWidths_1);
84 StatusBar1->SetStatusStyles(1, __wxStatusBarStyles_1);
85 SetStatusBar(StatusBar1);
86 SetStatusText( "Welcome to wxWidgets!" );
87 FileDialog1 = new wxFileDialog(this, _("Select file"), wxEmptyString, wxEmptyString, wxFileSelectorDefault
88
89 Connect(idMenuQuit, wxEVT_COMMAND_MENU_SELECTED, (wxObjectEventFunction) &wxSmithNotePFrame::OnQuit);
```
- Build Messages Panel:** Shows the following output:

```
File Line Message
C:\wxWidgets-3... 1917 note: declared here
=== Build finished: 0 error(s), 4 warning(s) (0 minute(s), 3 second(s)) ===
=== Run: Release in wxSmithNoteP (compiler: GNU GCC Compiler) ===
```

C:\Users\watsonh\Documents\CodeBlocks\wxSmithNoteP\wxSmithNotePMain.cpp

Windows (CR+LF)

WINDOWS-1252

Line 117, Column 21

Insert

Read/Write

default

Same process – File Save

The screenshot shows the Code::Blocks IDE interface with the following components:

- Management Panel (Left):** Shows a tree view of resources. A red arrow labeled '1' points to the 'Save' menu item under the 'File' menu.
- Event List (Bottom Left):** Shows 'EVT_MENU' with a dropdown menu containing '-- Add new handler --'. A red arrow labeled '2' points to this dropdown.
- Main Canvas (Center):** Displays a graphical user interface with a large blue text area. A 'New handler' dialog box is open, with a red arrow labeled '3' pointing to it. The dialog contains the text 'Enter name for new handler:' and a text input field with 'OnMenuItemSave' entered.
- Bottom Panel:** Contains a toolbar with various icons and a 'Logs & others' window showing compiler output and warnings.

At the bottom of the IDE, the status bar shows the file path 'wxsmith/wxSmithNotePframe.wxw' and the theme 'default'.

From PW5

This is the code which will do those 4 items.

1. Create another instance of the wxFileDialog

```
wxFileDialog *openDialog = new wxFileDialog(this, wxT("Choose a file"), wxT(""), wxT(""),  
wxT("Text Files (*.txt)|*.txt|C++ Files (*.cpp)|*.cpp|Header Files (*.h)|*.h"),  
wxFD_OPEN );
```

PW3: in order to get the file save dialog, wxFD_OPEN should be replaced with wxFD_SAVE

2. Cause the instance to 'pop-up' (ShowModal)

```
int response = openDialog->ShowModal(); //get response from the dialog
```

4. Load the selected file into the wxTextCtrl inside the wxFrame

```
if(response == wxID_OK)  
{ //if response ok, then load contents into textControl  
this->textControl->LoadFile(openDialog->GetPath());  
}
```

Second file dialog, wxID_Save style

The screenshot displays the wxWidgets IDE interface. On the left, the 'Resources' tree shows a project structure with a 'Tools' folder containing a 'File' menu bar. The 'File' menu bar has several items: '&File', 'Quit', 'Open', 'Save', and 'Help'. The 'Save' item is highlighted with a mouse cursor. Below the Resources tree is the 'Properties' window, which shows the configuration for the selected 'wxFileDialog' object. The 'Style' section is expanded, and the 'wxFD_SAVE' checkbox is checked. The main editor area shows a menu bar with the following items: 'File', '&File', 'Quit', 'Open', 'Save', and 'Help'. The 'Save' item is highlighted with a mouse cursor. The 'Logs & others' window at the bottom shows the build output, including a warning about a deprecated function and a note about a declaration.

Message	Select file
Default directory	
Default file	
Wildcard	
Var name	FileDialog2
Is member	<input checked="" type="checkbox"/>
Class name	wxFileDialog
Extra code	
Style	wxFD_DEFAULT_STYLE, wxFD_SA
wxFD_DEFAULT_STYLE	<input checked="" type="checkbox"/>
wxFD_OPEN	<input type="checkbox"/>
wxFD_SAVE	<input checked="" type="checkbox"/>
wxFD_OVERWRITE_PROMPT	<input type="checkbox"/>
wxFD_FILE_MUST_EXIST	<input type="checkbox"/>

```
File Line Message
==== Build: Release in wxSmithNoteP (compiler: GNU GCC Compiler) ====
C:\Users\watso... 120 warning: 'virtual void wxWindowBase::SetInitialBestSize(const wxSize&)' is deprecated: use SetInitialSize() inste
C:\wxWidgets-3... 1917 note: declared here
```

Same process 'New' to Clear the TextCtrl

The screenshot displays the Code::Blocks IDE interface. The main editor window shows the `wxSmithNotePMain.cpp` file with the `OnMenuItemNew` function highlighted in green. The function is as follows:

```
140 void wxSmithNotePFrame::OnMenuItemNew(wxCommandEvent& event)
141 {
142     this->TextCtrl1->Clear();
143 }
144
```

The Resources panel on the left shows the project structure, with the `New` menu item selected under the `File` menu. The Properties panel at the bottom left shows the properties for the `New` menu item:

Label	New
Accelerator	
Help text	
Enabled	<input checked="" type="checkbox"/>
Bitmap	Click to add
Var name	MenuItem5
Is member	<input checked="" type="checkbox"/>
Identifier	idMenuNew
Extra code	

The Logs & others panel at the bottom right shows the following build messages:

```
=== Build: Release in wxSmithNoteP (compiler: GNU GCC Compiler) ===
C:\Users\watso... 33 warning: 'virtual void wxWindowBase::SetInitialBestSize(const wxSize&) is deprecated: use SetInitialSize() inste
C:\wxWidgets-3... 1917 note: declared here
```