

## Re: Design orientated 'C' question

---

*Source:* <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.language/2006-03/msg01137.html>

---

- *From:* David Wilkinson <[no-reply@xxxxxxxxxxxxx](mailto:no-reply@xxxxxxxxxxxxx)>
  - *Date:* Sun, 26 Mar 2006 18:37:57 -0500
- 

Robby wrote:

Hi,

Please consider the following code:

Globals.h

```
=====
//Declaration of structure
struct tagIdRoomIndex
{
int iRoomNumber;
TCHAR *szRoomName;
};
=====
```

WndProc\_CW1.cpp

```
=====
#include <windows.h>
#include "Globals.h"

//Function declaration
void L1_txtDISPLAY_ArSt2MEMB( HDC, struct tagIdRoomIndex RIndex[] );

//Declaration of an array of structures
tagIdRoomIndex RIndex[] =
{
1 ,TEXT("ABC"),
2 ,TEXT("DEF"),
3 ,TEXT("GHI"),
4 ,TEXT("JKL"),
};

LRESULT CALLBACK WndProc_CW1 (HWND hwnd, UINT message,
WPARAM wParam, LPARAM lParam) {
switch(message)
```

Re: Design orientated 'C' question

```
{
WM_PAINT:
F1(hdc,RIndex); //Calling function
break;
}
//Other code....
}
```

LIB1\_FUNCTIONS.cpp (Function implementations)

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

//include "Globals.h" //Already included in WndProc.cpp

F1(HDC hdc,struct tagIdRoomIndex RIndex[])
{
//Other code ...

//Errors point to the < RIndex[i].iRoomNumber); > part of line below
TextOut (hdc,0,0,szBuffer,
wsprintf(szBuffer,TEXT("%d"),RIndex[i].iRoomNumber));

}
=====
```

This code compiles OKAY if I include the following line of code in the function implementation file called LIB1\_FUNCTIONS.cpp

```
#include "Globals.h"
```

However if I don't include "Globals.h" I get several errors. Here are 2 of them:

```
c:\_DTS_PROGRAMMING\vc++\MY_APPS_LAB\APPLICATION_1\LIB1_FUNCTIONS.cpp(74)
: error C2036: 'tagIdRoomIndex []' : unknown size
```

```
c:\_DTS_PROGRAMMING\vc++\MY_APPS_LAB\APPLICATION_1\LIB1_FUNCTIONS.cpp(74)
: error C2027: use of undefined type 'tagIdRoomIndex'
```

My question is, when I am passing the array of structures by the calling function:

```
F1(hdc,RIndex);
```

Am I not passing the array and the structure. I have tried the following ways to pass the array of structures in the calling function:

```
F1( hdc,RIndex); //Calling function
```

OR

```
F1( hdc,tagIdRoomIndex RIndex); //Calling function
```

## Re: Design orientated 'C' question

And I always get errors.....

I find it redundant if I have to include the "Globals.h" once in WndProc\_CW1 and once again in LIB1\_FUNCTIONS.cpp

I don't mind, I can include it twice! \*I could live with this\* However Isn't this bad coding habits?

As far as where my declarations take place, I believe that declaring a structure in a header would not really be a bad idea, right...I mean, I have seen it in book samples!  
But then again, they were not books that showed serious examples under the context dealing with structuring of programming.

I would appreciate all feedback on this issue... Thankyou all.

All suggestions are appreciated

Robby:

A few things:

1. The reason you use a funny name like tagIdRoomIndex is so you can typedef it

```
typedef struct tagIdRoomIndex
{
int iRoomNumber;
TCHAR *szRoomName;
} IdRoomIndex;
```

Now you can use the identifier IdRoomIndex without the klutzy struct keyword. Actually, if you are compiling your programs as C++, this device is not necessary, because the struct keyword is not needed and you can do

```
struct IdRoomIndex
{
int iRoomNumber;
TCHAR *szRoomName;
};
```

and use IdRoomIndex without the struct keyword.

2. The above is actually called a structure definition (not declaration). The only things that should go in header files are structure (or class in C++) definitions and function declarations (prototypes). No variables!

3. Every .cpp file that uses structure/class definitions or function declarations from a header must include that header. Remember, every .cpp file must compile by itself. It is absolutely NOT bad to include a header in more than one .cpp file. In fact the purpose of the header is to ensure that every .cpp file that uses the things in it is using the same version.

Re: Design orientated 'C' question

Once again I advise you, when you get confused, to try compiling each .cpp file separately. If it doesn't compile, figure out why (the answer does not depend on anything in any other .cpp file).

David Wilkinson

.