Subject: MSC8 compile error "Button Example"
Posted by yoco on Wed, 12 Jul 2006 12:19:50 GMT
View Forum Message <> Reply to Message

Enviroment:
OS: Windows XP SP2
Compiler: MS VC++ 2005 Express (MSC8)
Library: Windows 32 Plateform SDK installed
UPP ver.: 606 dev4
File: Ctrl.iml
Error message:

Button.cpp
C:\upp\uppsrc\CtrlLib/Ctrl.iml : warning C4819: The file contains a character that cannot be represented in the current code page (950). Save the file in Unicode format to prevent data loss
C:\upp\uppsrc\CtrlLib/Ctrl.iml(2) : error C2001: newline in constant
C:\upp\uppsrc\CtrlLib/Ctrl.iml(3) : error C2001: newline in constant
(and the following 100 same lines...)

I had done and checked Setup->Automaitc setup,
All directory are correct,
and I had also installed Win32SDK.
What should I do now ?

Thank you very much.
(Chameleon is cool!)

Subject: Re: MSC8 compile error "Button Example"
Posted by mirek on Thu, 13 Jul 2006 16:14:28 GMT
View Forum Message <> Reply to Message

yoco wrote on Wed, 12 July 2006 08:19Enviroment:
OS: Windows XP SP2
Compiler: MS VC++ 2005 Express (MSC8)
Library: Windows 32 Plateform SDK installed
UPP ver.: 606 dev4
File: Ctrl.iml
Error message:

Button.cpp
C:\upp\uppsrc\CtrlLib/Ctrl.iml : warning C4819: The file contains a character that cannot be represented in the current code page (950). Save the file in Unicode format to prevent data loss
C:\upp\uppsrc\CtrlLib/Ctrl.iml(2) : error C2001: newline in constant
C:\upp\uppsrc\CtrlLib/Ctrl.iml(3) : error C2001: newline in constant
(and the following 100 same lines...)

I had done and checked Setup->Automaitc setup,
All directory are correct,
and I had also installed Win32SDK.
What should I do now ?

Thank you very much.
(Chameleon is cool!)

Unfortunately, this is a "feature" of MSC8 - it fails this way if you are using some codepage in XP. It has some related troubles with strings as well and frankly, I am not sure what to do with that.

Well, there is one thing - there will be new .iml format using straight numbers instead of strings...

Mirek

Subject: Re: MSC8 compile error "Button Example"
Posted by yoco on Fri, 14 Jul 2006 12:44:39 GMT
View Forum Message <> Reply to Message

I solved this problem.

The problem caused by the default charset in Taiwan is Big-5 (similar with iso-8895-1).

In far-east country, their words are more than 255,
far more than a byte can represent,
so sometimes they hava to combine two bytes to represent one words.

In this charset,
If the ASCII of a char is smaller than 127, it will treate as a normal ANSI char.
If the ASCII of a char is bigger than 128, it will be combined with the next char to represent one far-east word.

A segment of the Ctrl.iml file
In standerd ANSI, it looks like this

but in Big-5 charset, it is...

The second qoute which to close the string literal,
was been combined with the previous byte to present one fareast word,
cause of the prevouis byte is bigger than 128.
Since there are no qoute to close the string literal,
so it raised a C2001 compile error in MS VC++.

I had wrote a simple program to translate the original .iml file to a new one,
by replace byte bigger than 128 to OCT string literal form.
Recompile, and everything is fine.

I hope this may be helpful to you.

PS. the simple program.

```
int main( int args, char* argv[] )
{
 string fn = argv[1] ;
 ifstream fin ( fn.c_str() ,ios_base::binary ) ;
 if ( !fin ) { cout << fn << " not exist!" << endl ; return 0 ;}

 string cmd( "copy" ) ;
 cmd = cmd + " " + fn + " " + fn + ".bak" ;
 system ( cmd.c_str() ) ;

 fin.close() ;
 fin.open( (fn+".bak").c_str() ) ;

 ofstream fout(fn.c_str()) ;

 char buf [5] = "\\" ;
 unsigned char ch ;
 while ( fin.get((char&)ch) ) {
  if ( ch<128 ) { fout.put(ch) ; }
  else {
   buf[1] = '0' + ch/64 ;
   buf[2] = '0' + (ch%64)/8 ;
   buf[3] = '0' + ch%8 ;
   fout << buf ;
  }
 }
```

```
}
```

Subject: Re: MSC8 compile error "Button Example"
Posted by mirek on Fri, 14 Jul 2006 12:58:40 GMT
View Forum Message <> Reply to Message

Yes, I am actually aware of this trouble. The MSC8 problem is that there is

```
 #pragma setlocale("C")
```

in Core/defs.h (so that it gets included everywhere), which should be supported by MSC, but unfortunately something is wrong in the compiler....

In fact, there is little trouble with .iml files - next version can and will change format so that this does not matter.

However, the real troubles comes with translation files - it would be rather unfortunate if they would not be editable directly...

Mirek