


Εγκατάσταση της OpenGL και της FreeGLUT στο προγραμματιστικό περιβάλλον Dev-C++

(Windows XP / Vista / 7 / 8.x / 10 – 64bit)

Βήμα 1:

Μεταφορτώνουμε στον υπολογιστή μας το αρχείο **Embarcadero_Dev-Cpp_6.3_TDM-GCC_9.2_Setup.exe** εγκατάστασης του **Dev-C++** από τη διεύθυνση:

<https://github.com/Embarcadero/Dev-Cpp/releases>

 Embarcadero_Dev-Cpp_6.3_No_Compiler_Portable.7z	6.16 MB
 Embarcadero_Dev-Cpp_6.3_No_Compiler_Setup.exe	8.09 MB
 Embarcadero_Dev-Cpp_6.3_No_Compiler_Setup.zip	7.85 MB
 Embarcadero_Dev-Cpp_6.3_TDM-GCC_9.2_Portable.7z	72.1 MB
 Embarcadero_Dev-Cpp_6.3_TDM-GCC_9.2_Setup.exe	70.7 MB
 Embarcadero_Dev-Cpp_6.3_TDM-GCC_9.2_Setup.zip	70.4 MB
 Source code (zip)	
 Source code (tar.gz)	

Η έκδοση αυτή διαθέτει ενσωματωμένο τον μεταγλωττιστή **TDM-GCC / MinGW**. Εγκαθιστούμε το **Dev-C++**, επιλέγουμε **Next** σε όλες τις οθόνες του οδηγού που ακολουθούν. Στο τελευταίο βήμα, πριν το κλείσιμο του οδηγού, αποεπιλέγουμε το 'Εκτέλεση του Dev-C++'.

Βήμα 2:

Μεταφορτώνουμε το πακέτο **FreeGLUT** για τον μεταγλωττιστή **MinGW** από τη διεύθυνση:

<https://www.transmissionzero.co.uk/software/freelut-devel/>

freelut 3.0.0 MinGW Package

This package contains 32 and 64 bit Windows DLLs, import libraries, static libraries, and header files, allowing freelut applications to be compiled using the MinGW compiler. The package was built from source code using MinGW. The DLL is binary compatible with both my MSVC DLL, and the GLUT for Win32 DLL provided by Nate Robins. The DLL has been tested on Windows 98 SE, Windows ME, Windows 2000, Windows XP, Windows Vista, Windows 7 (64 bit), and Windows 8 (64 bit).

The package should also work with the various IDEs which offer frontends to MinGW / gcc, for example [Eclipse](#), [Dev-C++](#), and [Code::Blocks](#). However, you'll need to consult the manual for your IDE for instructions on how to do this.

[Download freelut 3.0.0 for MinGW](#) (with PGP signature and PGP key)

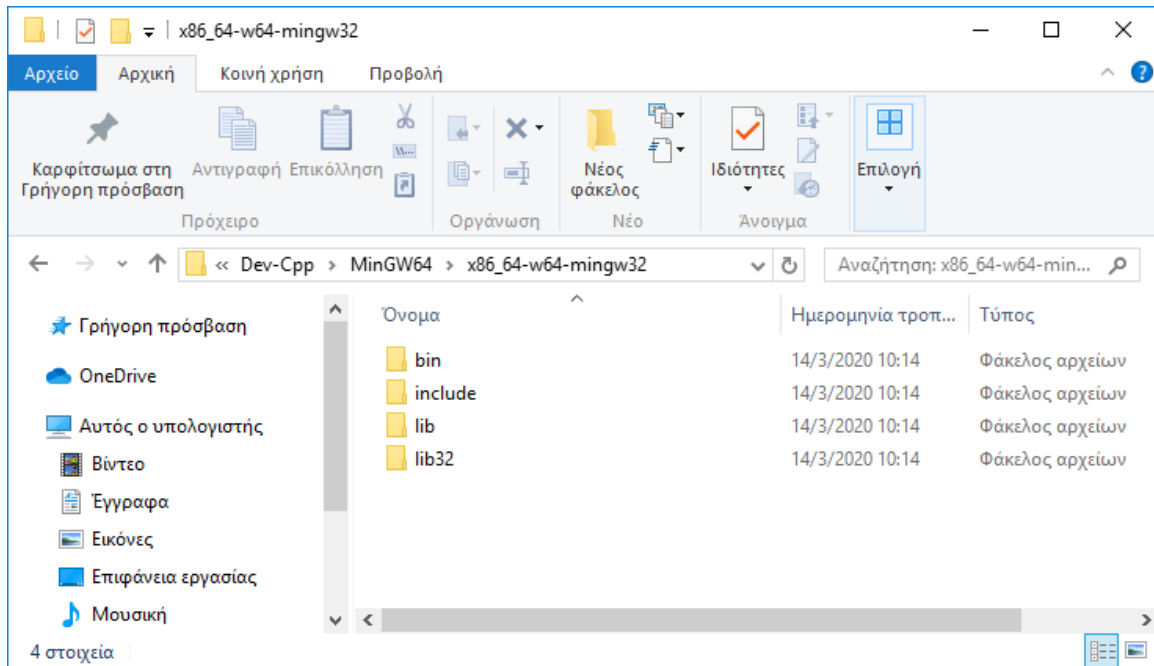
For further details on using freelut with MinGW, see my [GLUT MinGW article](#).

Το συμπιεσμένο αρχείο που μόλις κατεβάσαμε περιέχει έναν φάκελο με το όνομα **freelut**. Αποσυμπιέζουμε τον φάκελο αυτό σε έναν προσωρινό φάκελο, π.χ. στην **Επιφάνεια Εργασίας**.

Βήμα 3:

Μεταβαίνουμε στον φάκελο **x86_64-w64-mingw32** που βρίσκεται στον φάκελο εγκατάστασης του **Dev-C++**. Αν διαθέτουμε **Λ/Σ 64bit** τότε η διαδρομή είναι η:

C:\Αρχεία εφαρμογών (x86)\Embarcadero\Dev-Cpp\TDM-GCC-64\x86_64-w64-mingw32



Βήμα 4:

Αντιγράφουμε τα ακόλουθα αρχεία, από τους υποφακέλους του **freeglut** της **Επιφάνειας Εργασίας**, στους εξής προορισμούς:

Υποφάκελοι του φακέλου freeglut	Αντιγραφή των αρχείων	Προορισμός στον φάκελο x86_64-w64-mingw32
include\GL	freeglut.h freeglut_ext.h freeglut_std.h glut.h	include\GL
lib\x64	libfreeglut.a libfreeglut_static.a	lib
bin\x64	freeglut.dll	C:\Windows

Βήμα 5:

Ξεκινάμε το Dev-C++. Στον οδηγό ρυθμίσεων της πρώτης εκκίνησης, πατάμε σε όλα **Next**

Embarcadero Dev-C++ first time configuration

```
1 #include <iostream>
2
3 int main(int argc, char** argv)
4     std::cout << "Hello world!\n"
5     return 0;
6 }
```

Select your language:

- Bulgarian (Αλγύπρκθ)
- Catalan (Catal )
-         /Chinese
- Chinese (TW)
- Croatian
- Czech ( etina)
- Danish
- Dutch (Nederlands)
- English (Original)
- Estonian
- French
- Galego

You can later change the language at Tools >> Environment Options >> General.

> Next

Embarcadero Dev-C++ first time configuration

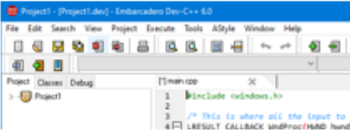
```
1 #include <iostream>
2
3 int main(int argc, char** argv)
4     std::cout << "Hello world!\n"
5     return 0;
6 }
```

Select your theme:

Font:

Color:

Theme:



You can later change themes at Tools >> Editor Options >> Fonts/Colors.

> Next

Embarcadero Dev-C++ first time configuration

```
1 #include <iostream>
2
3 int main(int argc, char** argv)
4     std::cout << "Hello world!\n"
5     return 0;
6 }
```

Embarcadero Dev-C++ has been configured successfully.

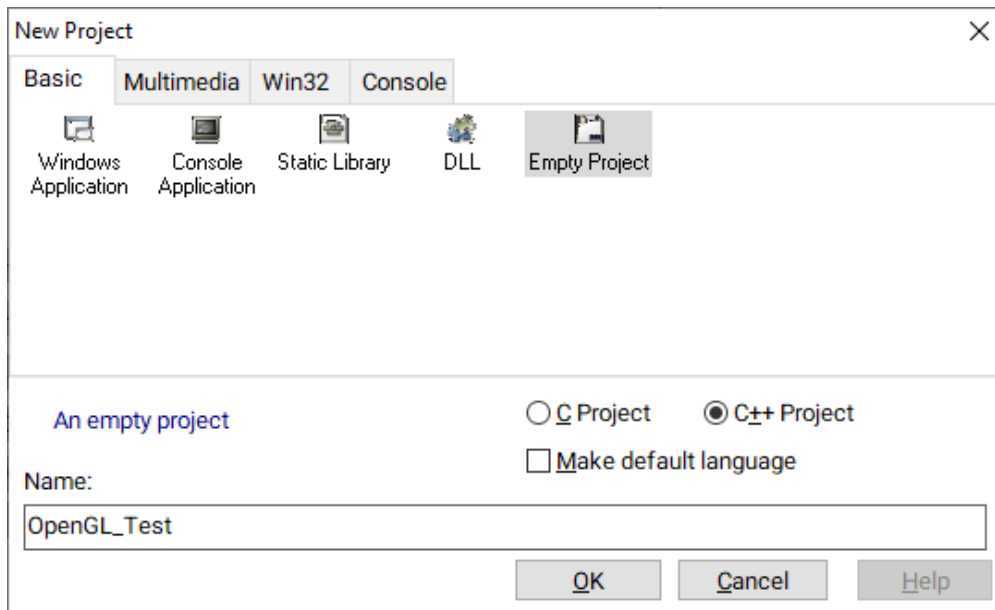
If you need help using Embarcadero Dev-C++, please refer to the Embarcadero Dev-C++ help file in the Help menu or send the developer a message (he doesn't mind!).

You can also download packages (like libraries or tools) to use with Embarcadero Dev-C++ using WebUpdate, which you will find in Tools menu >> Check for Packages.

> OK

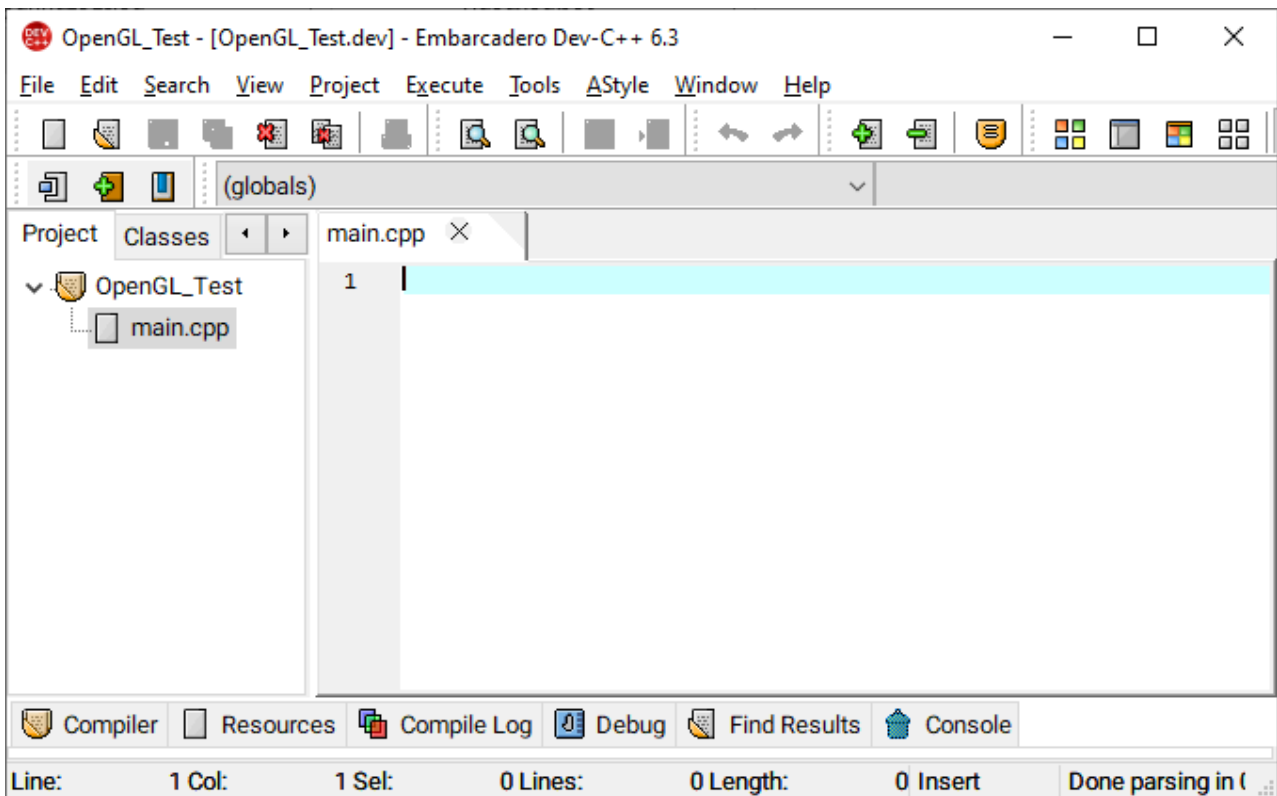
Βήμα 6:

Επιλέγουμε **File** → **New** → **Project...** Στον **οδηγό** που θα εμφανιστεί, επιλέγουμε **Empty Project**, **C++ Project**, και δίνουμε ένα όνομα, π.χ. **OpenGL_Test**



Στην ερώτηση για το που θα αποθηκεύσουμε το αρχείο με κατάληξη **.dev** (το project του Dev-C++), δημιουργούμε έναν φάκελο με όνομα **OpenGL_Test**, π.χ. στην **Επιφάνεια Εργασίας**, και το αποθηκεύουμε εκεί.

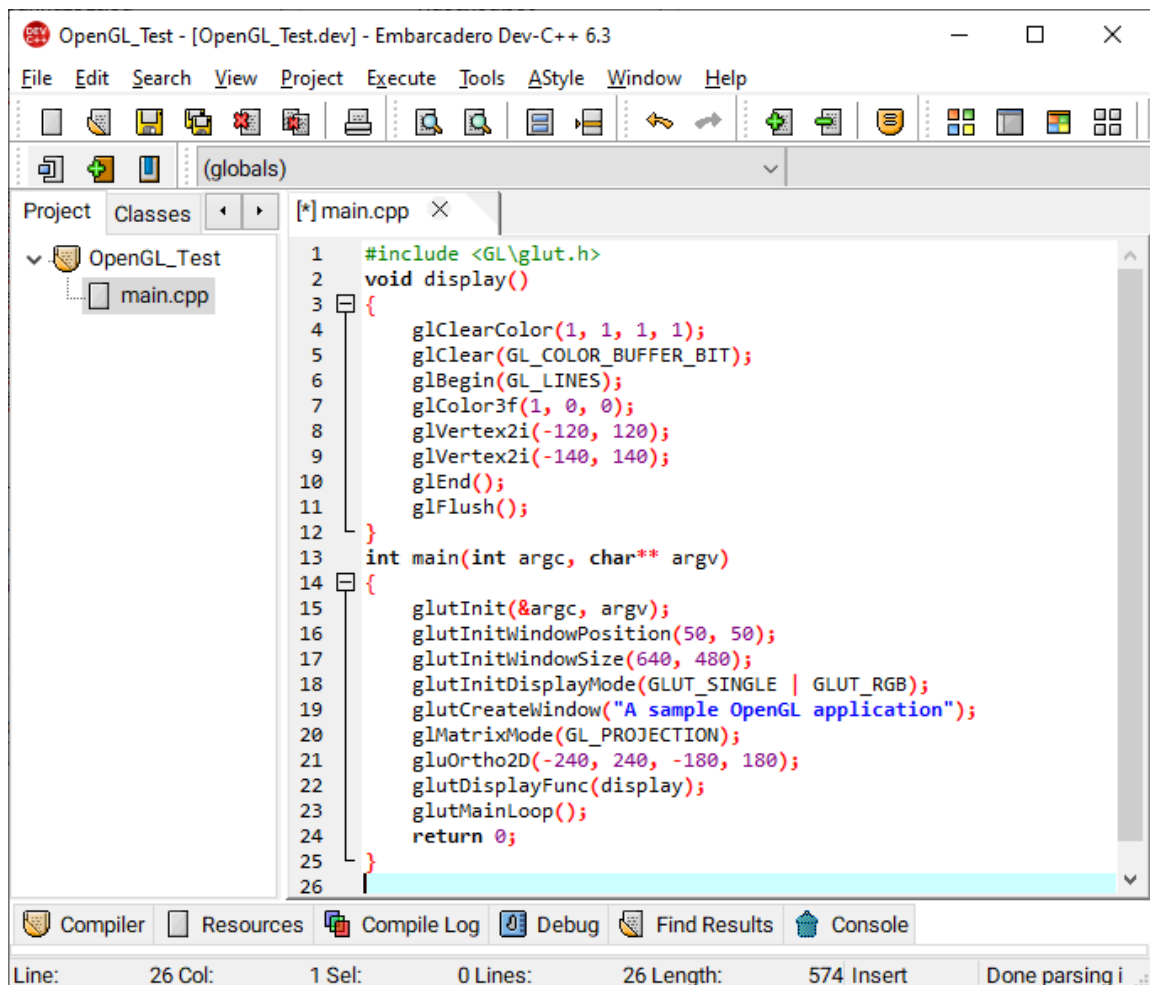
Επίσης επιλέγουμε αποθήκευση (**File** → **Save**) και δίνουμε ως όνομα αρχείου **main.cpp**. Το αρχείο αυτό το αποθηκεύουμε στον ίδιο φάκελο με το project του Dev-C++ που δημιουργήσαμε προηγουμένως.



Βήμα 7:

Επικολλούμε τον ακόλουθο κώδικα στο **main.cpp** αρχείο:

```
#include <GL\glut.h>
void display()
{
    glClearColor(1, 1, 1, 1);
    glClear(GL_COLOR_BUFFER_BIT);
    glBegin(GL_LINES);
    glColor3f(1, 0, 0);
    glVertex2i(-120, -120);
    glVertex2i(140, 140);
    glEnd();
    glFlush();
}
int main(int argc, char** argv)
{
    glutInit(&argc, argv);
    glutInitWindowPosition(50, 50);
    glutInitWindowSize(640, 480);
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
    glutCreateWindow("A sample OpenGL application");
    glMatrixMode(GL_PROJECTION);
    gluOrtho2D(-240, 240, -180, 180);
    glutDisplayFunc(display);
    glutMainLoop();
    return 0;
}
```



The screenshot shows the Embarcadero Dev-C++ IDE interface. The title bar reads "OpenGL_Test - [OpenGL_Test.dev] - Embarcadero Dev-C++ 6.3". The menu bar includes File, Edit, Search, View, Project, Execute, Tools, AStyle, Window, and Help. The toolbar contains various icons for file operations and development. The Project Explorer on the left shows a project named "OpenGL_Test" with a file named "main.cpp". The main editor window displays the following C++ code:

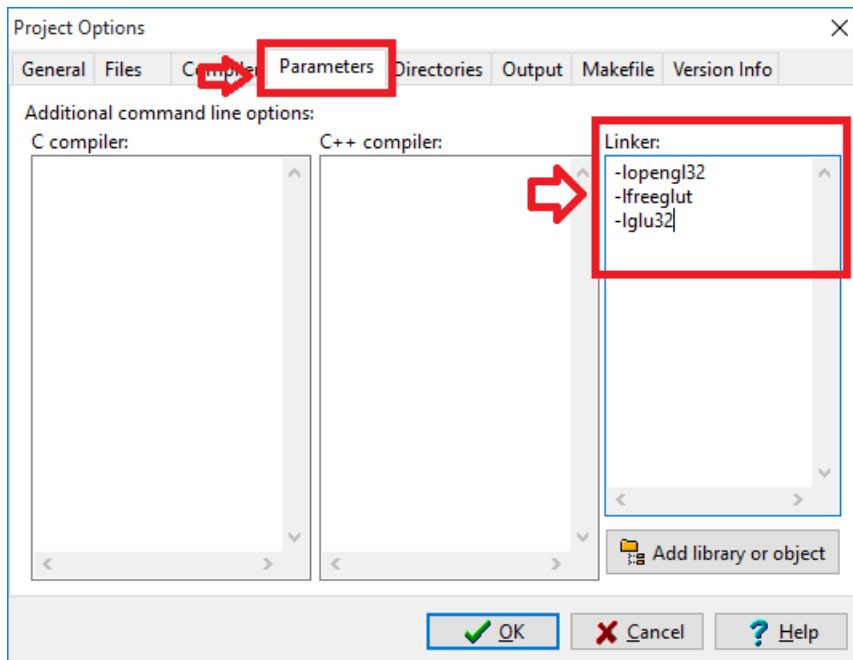
```
1  #include <GL\glut.h>
2  void display()
3  {
4      glClearColor(1, 1, 1, 1);
5      glClear(GL_COLOR_BUFFER_BIT);
6      glBegin(GL_LINES);
7      glColor3f(1, 0, 0);
8      glVertex2i(-120, 120);
9      glVertex2i(-140, 140);
10     glEnd();
11     glFlush();
12 }
13 int main(int argc, char** argv)
14 {
15     glutInit(&argc, argv);
16     glutInitWindowPosition(50, 50);
17     glutInitWindowSize(640, 480);
18     glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
19     glutCreateWindow("A sample OpenGL application");
20     glMatrixMode(GL_PROJECTION);
21     gluOrtho2D(-240, 240, -180, 180);
22     glutDisplayFunc(display);
23     glutMainLoop();
24     return 0;
25 }
26
```

The status bar at the bottom shows: Line: 26 Col: 26 Sel: 1 Len: 0 Lines: 26 Length: 574 Insert Done parsing i

Βήμα 8:

Από το μενού, επιλέγουμε το **Project → Project Options...** και στο παράθυρο που ανοίγει επιλέγουμε την καρτέλα **Parameters**. Εκεί, στο πεδίο **Linker**, πληκτρολογούμε τις επιλογές που θέλουμε να χρησιμοποιήσει ο συνδέτης (με τη σειρά που φαίνονται):

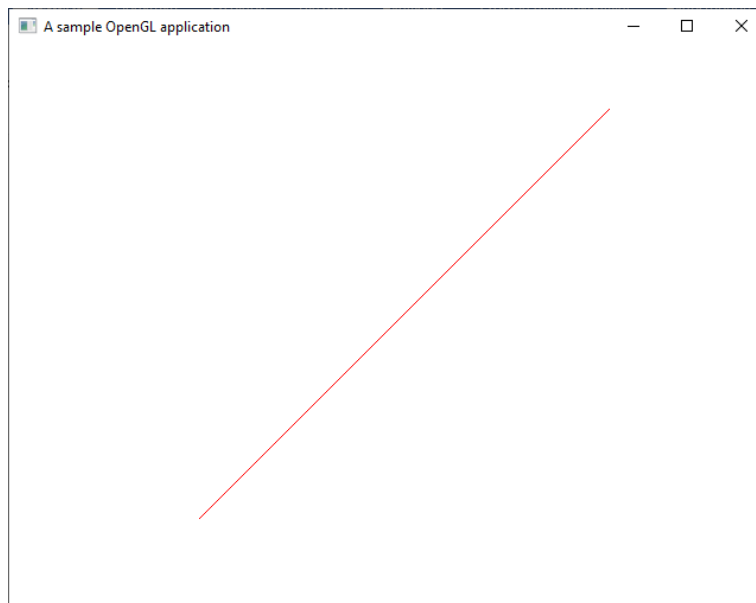
```
-lopengl32  
-lfreetglut  
-lglu32
```



Τις επιλογές αυτές θα πρέπει να τις πληκτρολογούμε κάθε φορά που δημιουργούμε ένα νέο project.

Βήμα 9:

Από το μενού, επιλέγουμε το **Execute → Compile & Run** ή απλώς πατάμε το πλήκτρο **F11**. Αν όλα πήγαν καλά θα πρέπει να εμφανιστεί ένα παράθυρο με μία κόκκινη γραμμή στο εσωτερικό του.



Προαιρετικά:

Κατά την εκτέλεση, μαζί με το παράθυρο του προγράμματος εμφανίζεται και ένα παράθυρο γραμμής εντολών (Command Line prompt) το οποίο συνήθως μας βοηθάει να βλέπουμε μηνύματα ή τιμές μεταβλητών. Αν δεν θέλουμε να εμφανίζεται, αλλάζουμε τον τύπο της εφαρμογής μας από **Win32 Console** σε **Win32 GUI** από τα **Project Options**.

