

Εγκατάσταση της OpenGL και της FreeGLUT στο Visual Studio Code



Windows 7 / 8.x / 10 / 11 (64 Bit)

Βήμα 1:

Μεταφορτώνουμε στον υπολογιστή μας το αρχείο εγκατάστασης του **Minimalist GNU for Windows** (εν συντομία **MinGW**) από την παρακάτω διεύθυνση, πηγαίνοντας στο σημείο **Download - MSVCRT runtime** και πατάμε τον σύνδεσμο **Zip archive** της επιλογής **GCC 11.2.0 + MinGW-w64 9.0.0 (MSVCRT) – release 6 (LATEST)** και **Win64**.

<https://winlibs.com/>

Download

The following downloads are available (for Windows only).
*You will need a decompressor like [7-Zip](#) (free) to unzip .7z archives, but they are a lot smaller than .zip files.

Each version comes in 2 flavors:

- Win32 - i686 - Windows 32-bit version, runs natively on and compiles for Windows 32-bit (also runs on Windows 64-bit)
- Win64 - x86_64 - Windows 64-bit version, runs natively on and compiles for Windows 64-bit (will not run on Windows 32-bit)

Release versions

UCRT runtime

- GCC 11.2.0 + MinGW-w64 9.0.0 (UCRT) - release 5 (**LATEST**)
 - Win32: [7-Zip archive*](#) | [Zip archive](#)
 - Win64: [7-Zip archive*](#) | [Zip archive](#)
- GCC 11.2.0 + LLVM/Clang/LLD/LLDB 13.0.0 + MinGW-w64 9.0.0 (UCRT) - release 2
 - Win32: [7-Zip archive*](#) | [Zip archive](#) - without LLVM/Clang/LLD/LLDB: [7-Zip archive*](#) | [Zip archive](#)
 - Win64: [7-Zip archive*](#) | [Zip archive](#) - without LLVM/Clang/LLD/LLDB: [7-Zip archive*](#) | [Zip archive](#)

MSVCRT runtime

- GCC 11.2.0 + MinGW-w64 9.0.0 (MSVCRT) - release 6 (**LATEST**)
 - Win32: [7-Zip archive*](#) | [Zip archive](#)
 - Win64: [7-Zip archive*](#) | [Zip archive](#)

Από το αρχείο που κατεβάσαμε, αντιγράφουμε τον φάκελο **mingw64** στον ριζικό κατάλογο του σκληρού μας δίσκου **C**. Ο φάκελος **C:\mingw64** έχει την ακόλουθη δομή αρχείων και φακέλων:

File Explorer window showing the contents of the **mingw64** directory on the C: drive.

Όνομα	Ημερομηνία τροποποι...	Τύπος	Μέγεθος
bin	10/2/2022 9:35 πμ	Φάκελος αρχείων	
include	10/2/2022 9:35 πμ	Φάκελος αρχείων	
lib	10/2/2022 9:35 πμ	Φάκελος αρχείων	
libexec	10/2/2022 9:35 πμ	Φάκελος αρχείων	
share	10/2/2022 9:35 πμ	Φάκελος αρχείων	
x86_64-w64-mingw32	10/2/2022 9:35 πμ	Φάκελος αρχείων	
version_info.txt	10/2/2022 9:35 πμ	Εγγραφο κειμένου	1 KB

Βήμα 2:

Μεταβαίνουμε στη διεύθυνση:

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

Από εκεί, μεταφορτώνουμε στον υπολογιστή μας και στην **Επιφάνεια Εργασίας** το αρχείο **freeglut-MinGW-3.0.0-1.mp.zip**, πατώντας τον σύνδεσμο **Download freeglut 3.0.0 for MinGW**:

freeglut 3.0.0 MinGW Package

This package contains 32 and 64 bit Windows DLLs, import libraries, static libraries, and header files, allowing freeglut 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 freeglut 3.0.0 for MinGW](#) (with PGP signature and PGP key)

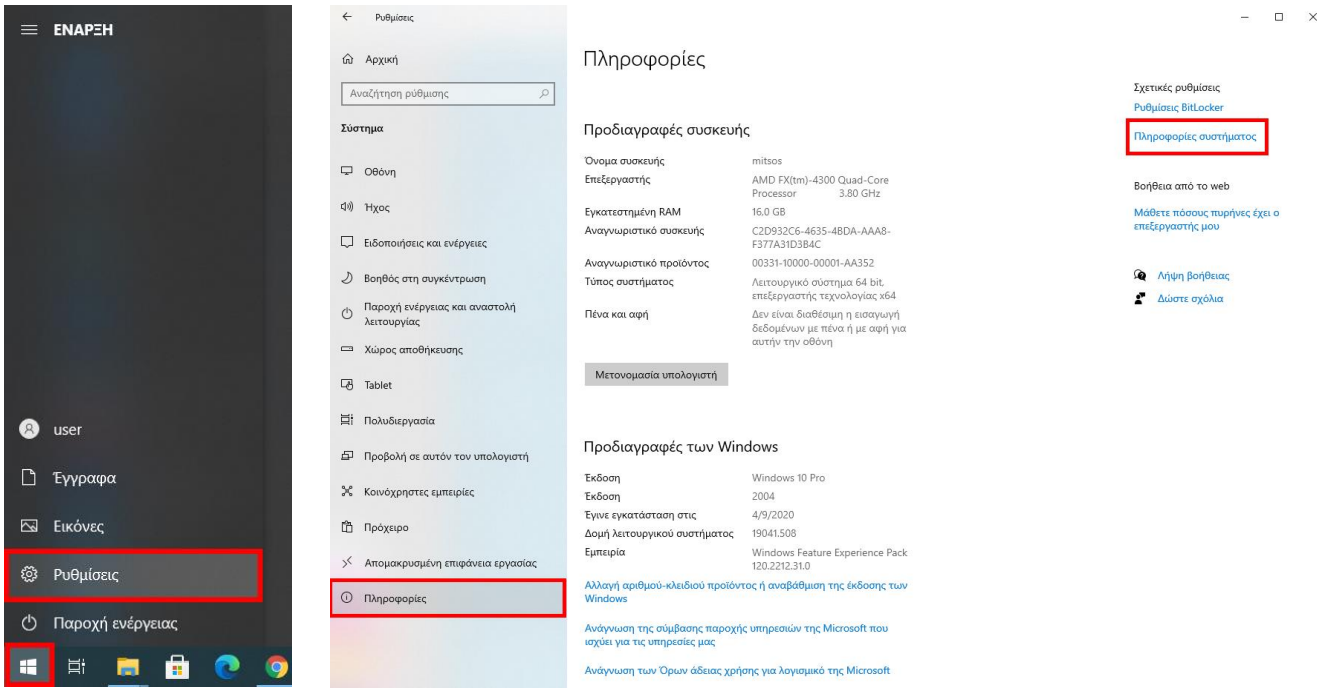
For further details on using freeglut with MinGW, see my [GLUT MinGW article](#).

Αποσυμπιέζουμε τα αρχεία από τους υποφακέλους που υπάρχουν στο αρχείο **freeglut-MinGW-3.0.0-1.mp.zip** που μόλις λάβαμε στην **Επιφάνεια Εργασίας**, στους παρακάτω υποφακέλους του **mingw64** και των **Windows**. Συγκεκριμένα:

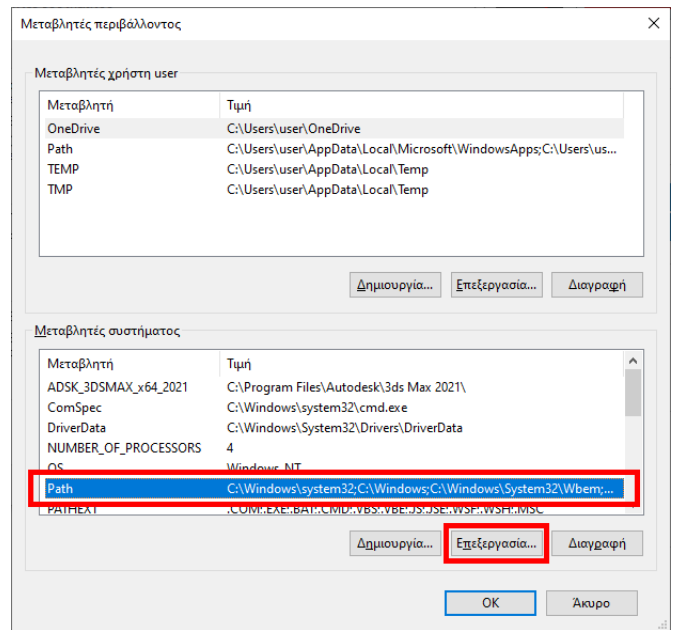
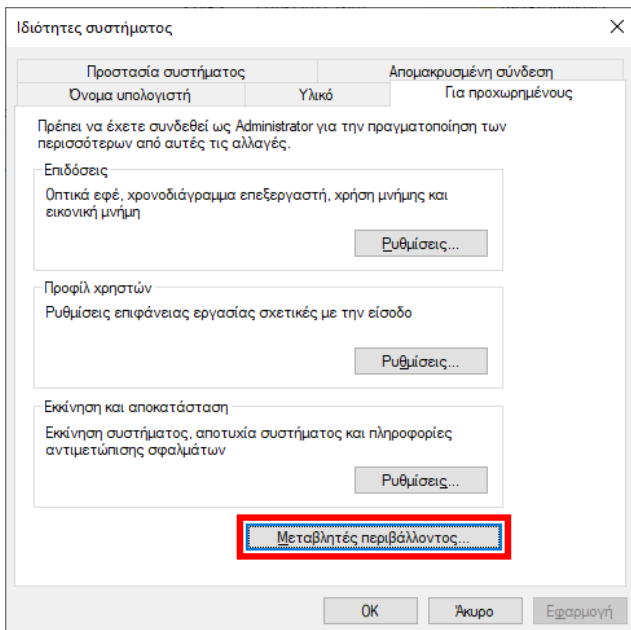
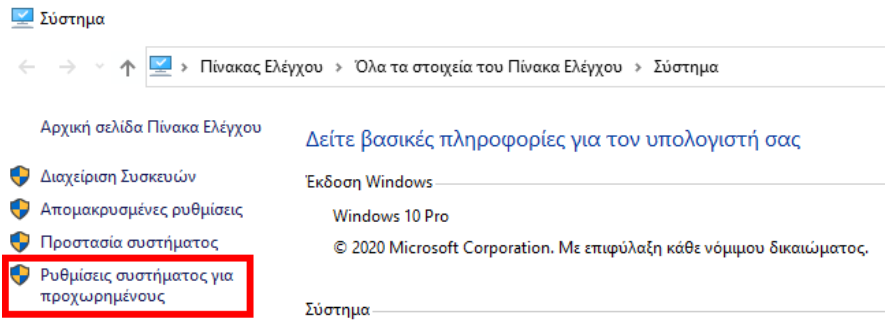
freeglut-MinGW-3.0.0-1.mp.zip	Αντιγραφή αρχείου(-ων)	Προορισμός(-οί)
freeglut\bin\x64	freeglut.dll	C:\mingw64\x86_64-w64-mingw32\bin C:\Windows
freeglut\include\GL	freeglut.h freeglut_ext.h freeglut_std.h glut.h	C:\ mingw64\x86_64-w64-mingw32\include\GL
freeglut\lib\x64	libfreeglut.a libfreeglut_static.a	C:\ mingw64\x86_64-w64-mingw32\lib

Βήμα 3:

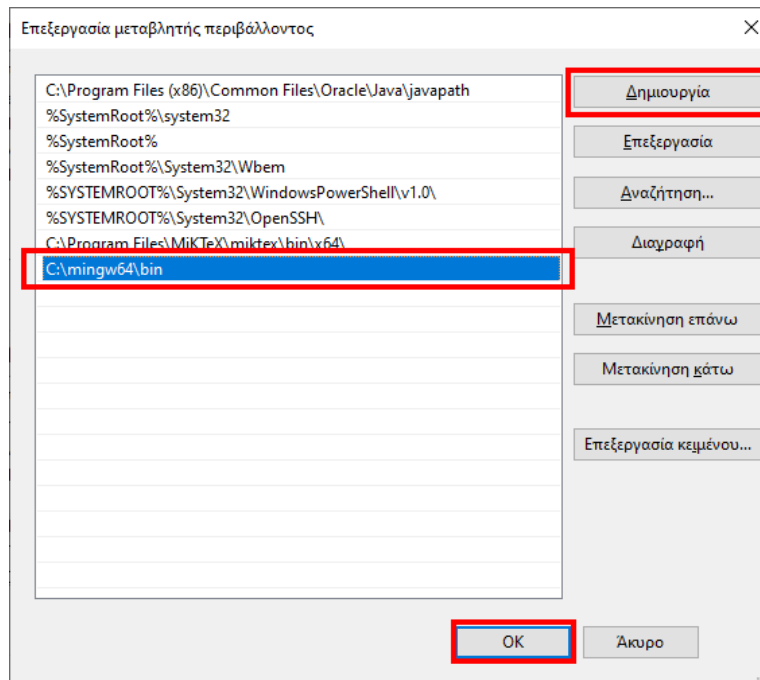
Πηγαίνουμε στις **Ρυθμίσεις** των Windows (**Εναρξη → Ρυθμίσεις**) και στη συνέχεια **Σύστημα → Πληροφορίες** και επιλέγουμε το **Πληροφορίες Συστήματος** που βρίσκεται στα δεξιά.



Από το παράθυρο που θα ανοίξει, επιλέγουμε **Ρυθμίσεις συστήματος για προχωρημένους** από τα δεξιά και μετά **Μεταβλητές περιβάλλοντος...** Στο **Μεταβλητές συστήματος** επιλέγουμε την **Path**, πατάμε το κουμπί **Επεξεργασία**.



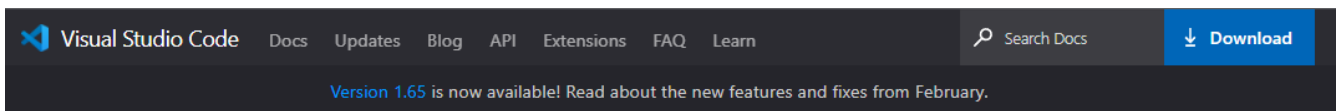
Επιλέγουμε **Δημιουργία** και πληκτρολογούμε ως νέα τιμή τη διαδρομή: **C:\mingw64\bin**. Πατάμε **OK** και κλείνουμε όλα τα παράθυρα.



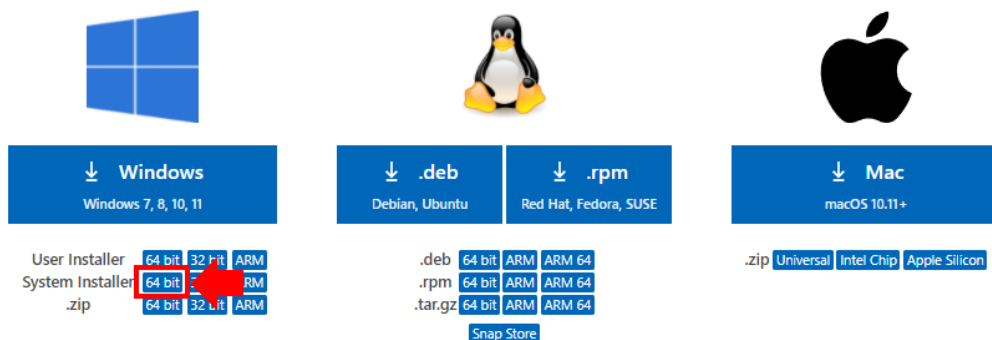
Βήμα 4:

Μεταφορτώνουμε στον υπολογιστή μας το αρχείο εγκατάστασης του **Visual Studio Code** από την παρακάτω διεύθυνση, πατώντας τον σύνδεσμο **64 bit** της επιλογής **System Installer** των **Windows**.

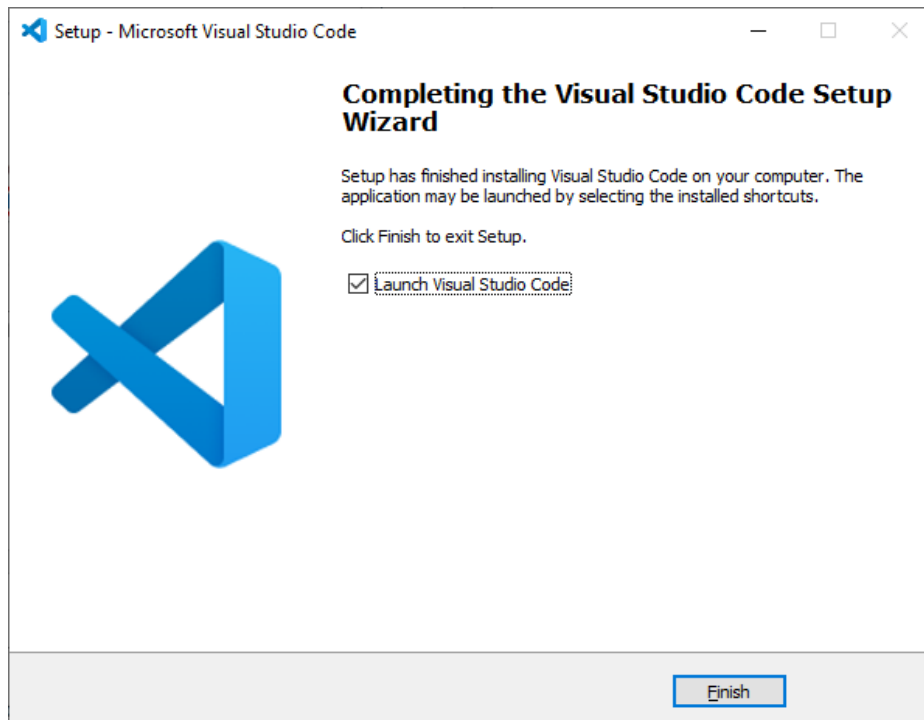
<https://code.visualstudio.com/download>



Download Visual Studio Code
Free and built on open source. Integrated Git, debugging and extensions.

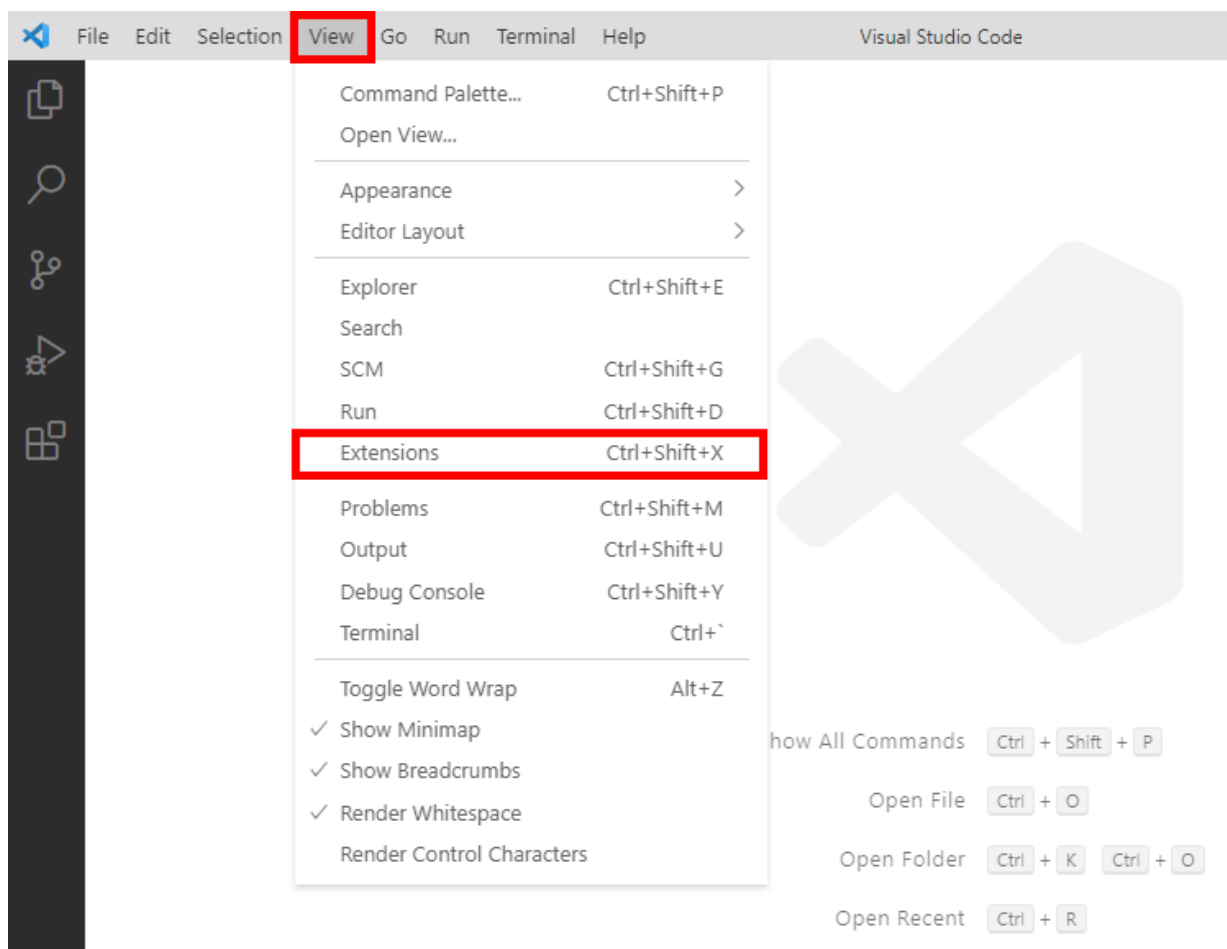


Εκκινούμε το αρχείο της εγκατάστασης, αφήνουμε ως έχουν τις προκαθορισμένες ρυθμίσεις σε κάθε βήμα και πατάμε **Next**. Στο τέλος κλείνουμε τον διάλογο της εγκατάστασης πατώντας **Finish** και εκτελούμε το **Visual Studio Code**.

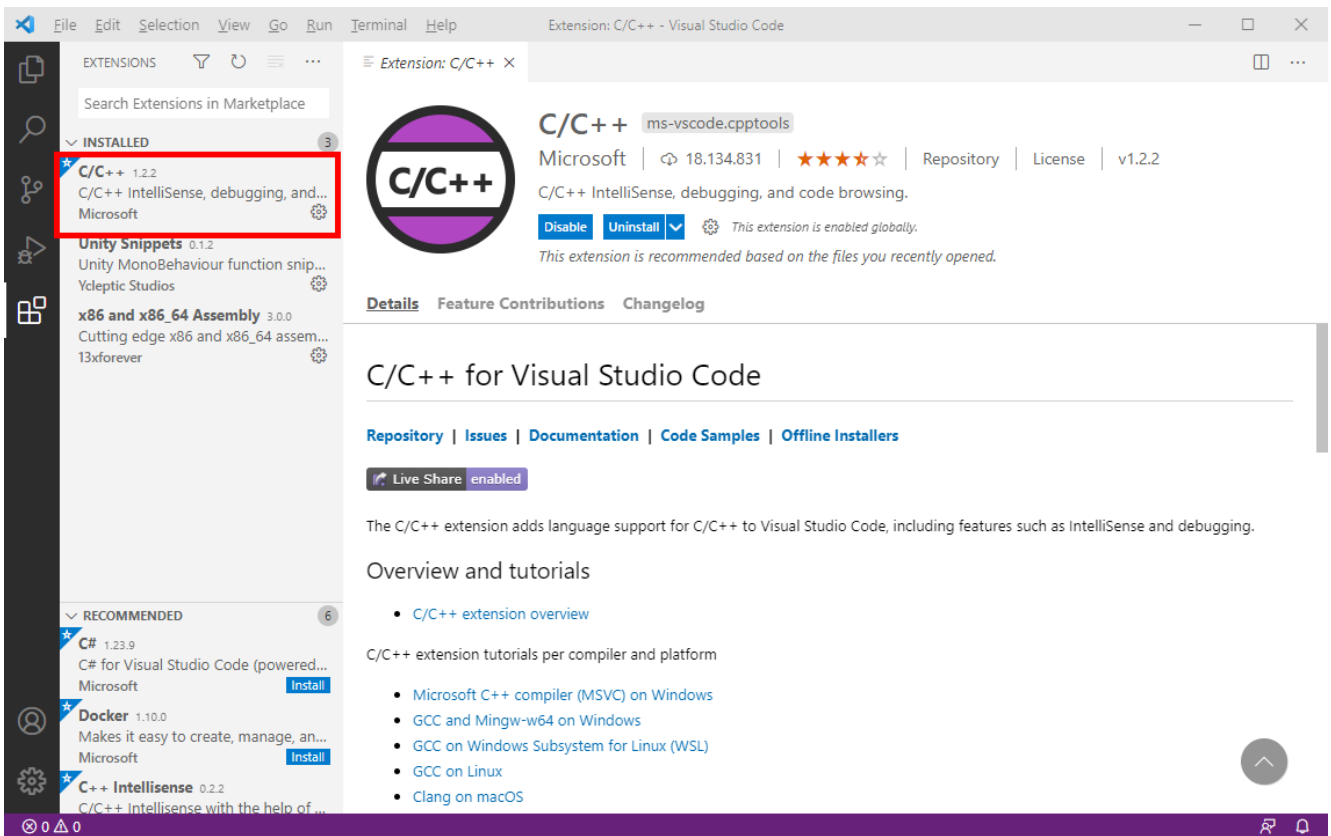


Βήμα 5:

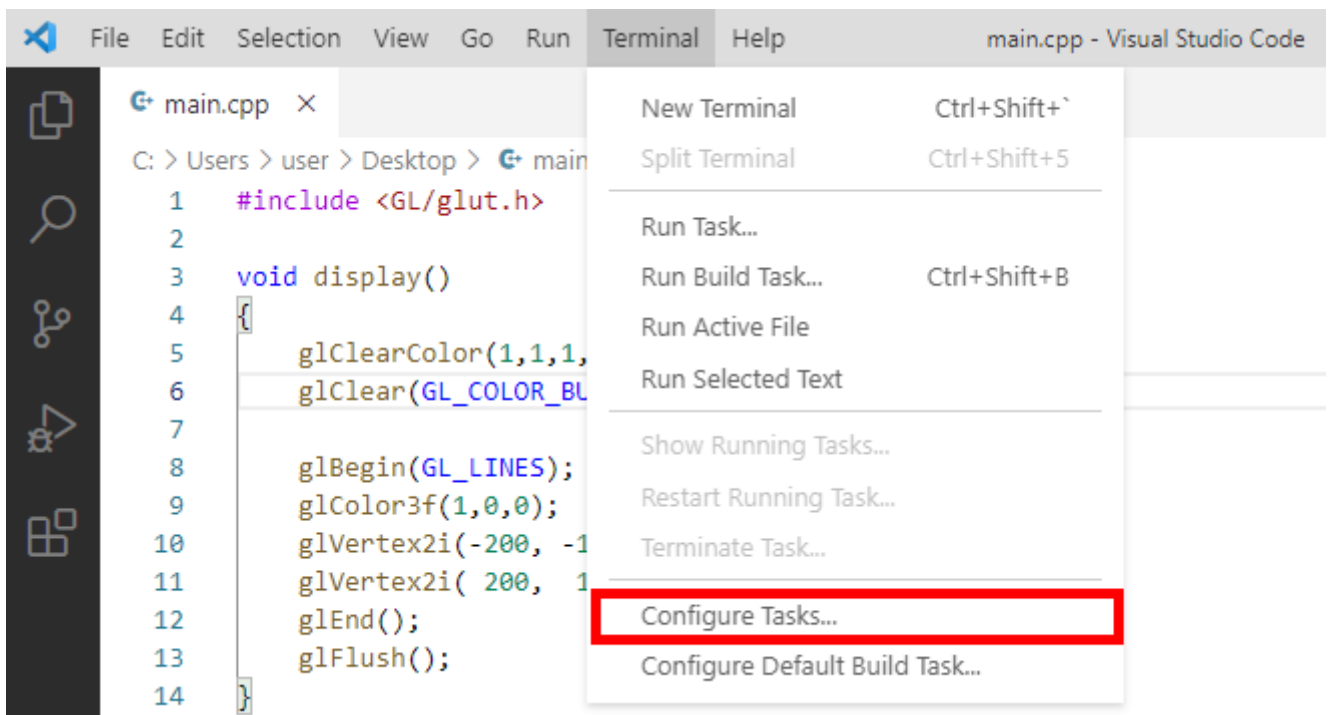
Από το μενού **View** επιλέγουμε το **Extensions** ή πατάμε **CTRL+Shift+X**



Εκεί πληκτρολογούμε **C/C++ for Visual Studio Code** και κάνουμε αναζήτηση. Εγκαθιστούμε την αντίστοιχη επέκταση της **Microsoft**.



Από το μενού **Terminal** επιλέγουμε **Configure Tasks...**



Επιλέγουμε διαδοχικά **Create tasks.json file from template** και μετά **Others (Example to run an arbitrary command)**. Κατόπιν, στο αρχείο **tasks.json** που εμφανίζεται, πληκτρολογούμε τον επόμενο κώδικα προκειμένου να προσθέσουμε τη δυνατότητα μεταγλώττισης προγραμμάτων σε **C++ / OpenGL** μέσω του μεταγλωττιστή **g++** του **mingw64**.

```

{
  "version": "2.0.0",
  "tasks":
  [
    {
      "type": "shell",
      "label" : "C/C++ - Build program",
      "command": "C:\\mingw64\\bin\\g++.exe",
      "args": [
        "-g", "${file}",
        "-o", "${fileDirname}\\${fileBasenameNoExtension}.exe",
        "-I", "C:\\mingw64\\include",
        "-L", "C:\\mingw64\\lib",
        "-lfreeglut",
        "-lopengl32",
        "-lglu32"
      ],
      "group":
      {
        "kind":"build",
        "isDefault":true
      },
      "problemMatcher": ["$gcc"],
    }
  ]
}

```

Από το μενού **File** → **New File** δημιουργούμε ένα νέο αρχείο κειμένου. Πληκτρολογούμε τον ακόλουθο κώδικα και τον αποθηκεύουμε σε έναν φάκελο (π.χ. στην **Επιφάνεια Εργασίας**) ως **main.cpp**.

```

#include <GL/glut.h>

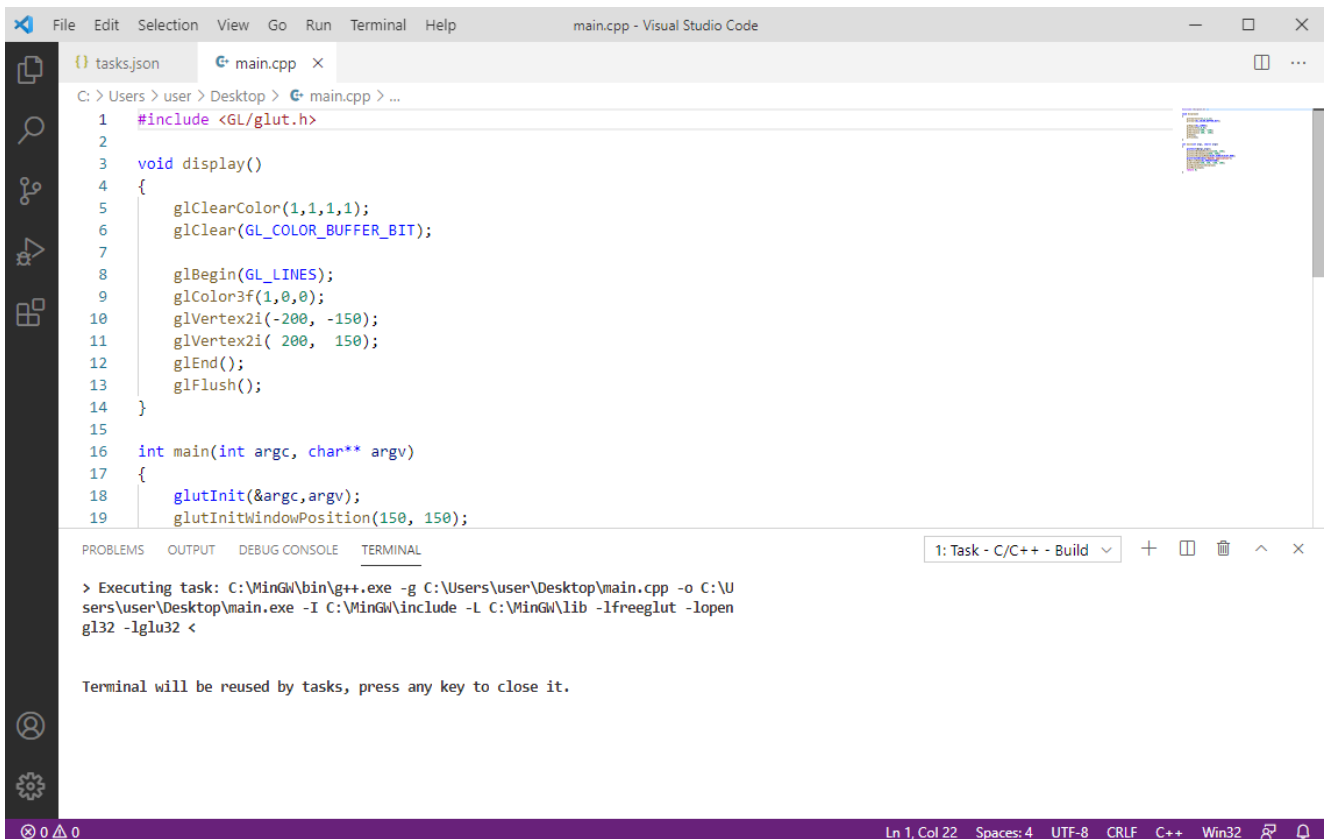
void display()
{
  glClearColor(1, 1, 1, 1);
  glClear(GL_COLOR_BUFFER_BIT);

  glBegin(GL_LINES);
  glColor3f(1, 0, 0);
  glVertex2i(-200, -150);
  glVertex2i( 200,  150);
  glEnd();
  glFlush();
}

int main(int argc, char** argv)
{
  glutInit(&argc,argv);
  glutInitWindowPosition(150, 150);
  glutInitWindowSize(640, 480);
  glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
  glutCreateWindow("OpenGL application");
  glMatrixMode(GL_PROJECTION);
  gluOrtho2D(-240, 240, -180, 180);
  glutDisplayFunc(display);
  glutMainLoop();
  return 0;
}

```

Από το μενού **Terminal** επιλέγουμε **Run Build Task...** (ή **CTRL+Shift+B**)



```
1 #include <GL/glut.h>
2
3 void display()
4 {
5     glClearColor(1,1,1,1);
6     glClear(GL_COLOR_BUFFER_BIT);
7
8     glBegin(GL_LINES);
9     glColor3f(1,0,0);
10    glVertex2i(-200, -150);
11    glVertex2i( 200, 150);
12    glEnd();
13    glFlush();
14 }
15
16 int main(int argc, char** argv)
17 {
18     glutInit(&argc, argv);
19     glutInitWindowPosition(150, 150);
```

Terminal output:

```
> Executing task: C:\MinGW\bin\g++.exe -g C:\Users\user\Desktop\main.cpp -o C:\Users\user\Desktop\main.exe -I C:\MinGW\include -L C:\MinGW\lib -lfreeglut -lopenGL32 -lglu32 <
Terminal will be reused by tasks, press any key to close it.
```

Αν η διαδικασία τερμάτισε χωρίς σφάλματα τότε στην **Επιφάνεια Εργασίας** έχει δημιουργηθεί το εκτελέσιμο αρχείο **main.exe** το οποίο αν τρέξουμε θα ανοίξει ένα παράθυρο με τίτλο **OpenGL application** και στο κέντρο του θα έχει μία κόκκινη γραμμή.

