

# OpenGL Configuration

## Mac User:

### Basic Setting

1. Google homebrew -> Open Terminal -> Input `/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"`
2. `brew install glfw3`
3. `brew install glew`

### Xcode Setting

1. Open Xcode -> Create OS X Command Line Tool (Language C++)
2. Click Project Layout -> Build Setting -> Search Paths -> Header Search Paths -> add `/usr/local/include`
3. Build Phases -> Link Binary With Libraries -> add `OpenGL.framework`
4. add -> add other -> press `Command+Shift+G` -> input `/usr/local -> Cellar/glew/1.12.0/lib/libGLEW.1.12.0.dylib *`
5. add -> add other -> press `Command+Shift+G` -> input `/usr/local -> Cellar/glfw/3.1.2/lib/libglfw.3.3.1.dylib *`  
(the number in step 4 and 5 may be changed)

Now jump to Coding Part at the bottom of this tutorial!

## Windows User:

### Basic Setting

1. google Visual Studio and download
2. google GLEW -> download Binaries Windows 32-bit and 64-bit
3. google GLFW -> click download on top right corner -> Windows pre-compiled binaries -> 32-bit (recommended)

### Visual Studio Setting

1. Create New Project -> Visual C++ Win32 Console Application (Empty project)
2. Add New Item -> C++ file (main.cpp)
3. Right Click the Project (Not the Solution) -> Property -> C/C++ -> General -> Additional Include Directories -> add ( then click ... button to browse) your `glew/include` (and) `glfw/include` folder
4. Linker -> General -> Additional Library Directories -> add ( then click ... button to browse) `glew/lib/release/win32` `glfw/lib-vc2015`
5. Input -> Additional Dependencies -> `opengl32.lib` `glew32s.lib` `glfw3.lib`
6. Copy the `glew32.dll` in your `glew/bin/release/win32` paste to the same folder of main.cpp file

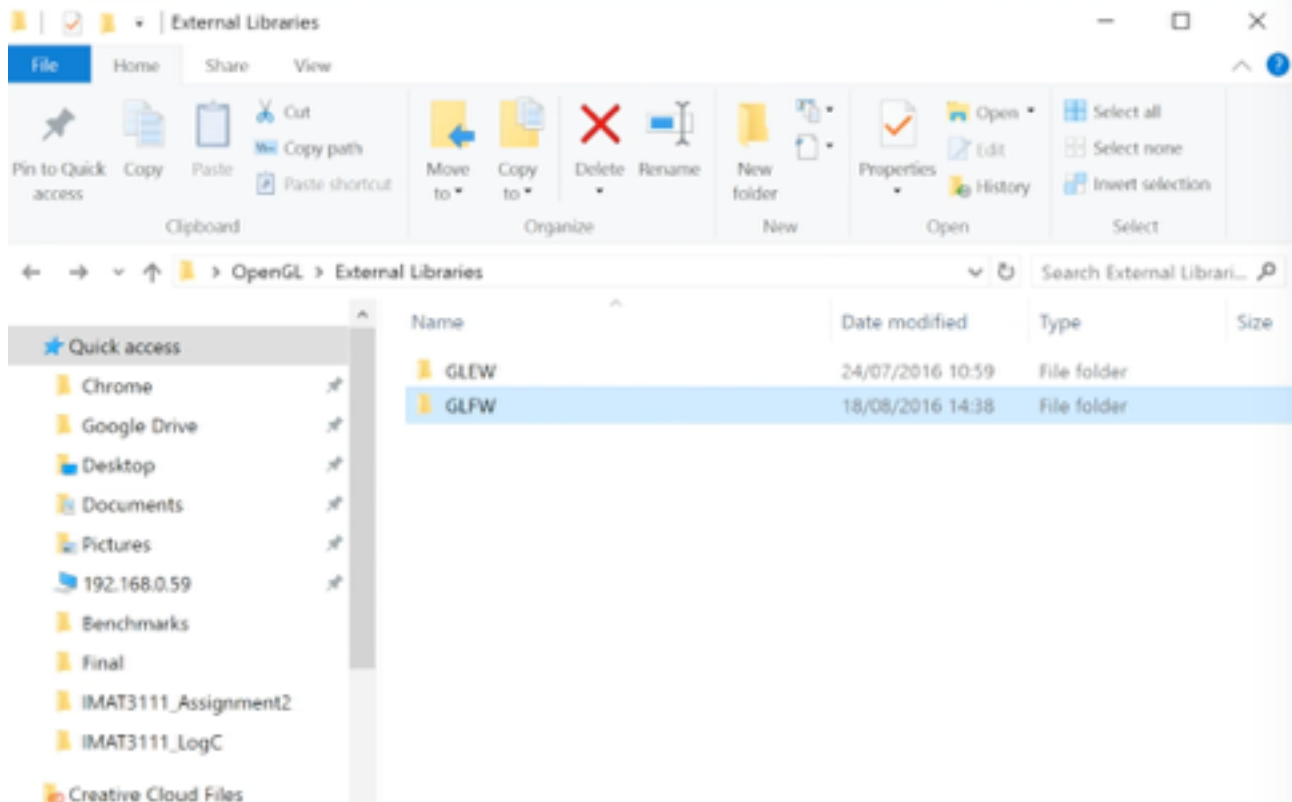
#### Optional Setting (Relative Linking)

There is a way for relative link setting which the GLEW and GLFW are in the same folder of the project. This tutorial only mentions the different part.

3. Right Click the Project (Not the Solution) -> Property -> C/C++ -> General -> Additional Include Directories -> add and input `$(SolutionDir)/../External Libraries/GLFW/include` (and) `$(SolutionDir)/../External Libraries/GLEW/include`
4. Linker -> General -> Additional Library Directories -> add ( and input) `$(SolutionDir)/../External Libraries/GLEW/lib/release/win32` (and) `$(SolutionDir)/../External Libraries/GLFW/lib-vc2015`

`$(SolutionDir)` refers to the folder which contains the project.sln file.

The file and folder structure is shown in the following figure.



## Coding

1. Delete the content in main.cpp
2. Type the following code

```
#include <iostream>
```

```
//GLEW
```

```
#define GLEW_STATIC
```

```
#include <GL/glew.h>
```

```
//GLFW
```

```
#include <GLFW/glfw3.h>
```

```
const GLint WIDTH = 800, HEIGHT = 600;
```

```
int main()
```

```
{
```

```
    glfwInit();
```

```
    glfwWindowHint(GLFW_CONTEXT_VERSION_MAJOR, 3);
```

```
    glfwWindowHint(GLFW_CONTEXT_VERSION_MINOR, 3);
```

```
    glfwWindowHint(GLFW_OPENGL_PROFILE, GLFW_OPENGL_CORE_PROFILE);
```

```
    glfwWindowHint(GLFW_OPENGL_FORWARD_COMPAT, GL_TRUE); // must for Mac
```

```
    glfwWindowHint(GLFW_RESIZABLE, GL_FALSE);
```

```
    GLFWwindow *window = glfwCreateWindow(WIDTH, HEIGHT, "Learn OpenGL", nullptr, nullptr);
```

```
    // next two lines are for mac retina display
```

```
    int screenWidth, screenHeight;
```

```
    glfwGetFramebufferSize( window, &screenWidth, &screenHeight);
```

```
    if (nullptr == window)
```

```
    {
```

```
        std::cout << "Failed to create GLFW window" << std::endl;
```

```
        glfwTerminate();
```

```
        return -1;
```

```
    }
```

```

glfwMakeContextCurrent( window );
glewExperimental = GL_TRUE;

if (GLEW_OK != glewInit() )
{
    std::cout << "Failed to initialise GLEW" << std::endl;
    return -1;
}
glViewport( 0, 0, screenWidth, screenHeight );

while (!glfwWindowShouldClose (window))
{
    glfwPollEvents();
    glClearColor(0.2f, 0.3f, 0.3f, 1.0f);
    glClear (GL_COLOR_BUFFER_BIT);

    glfwSwapBuffers (window);
}

glfwTerminate();
return 0;
}

```

Great!!! You should have a window and enjoy yourself!