# IMAT3904: Lab 3

Make sure you have completed Lab 2.

Clone Lab 3 from https://classroom.github.com/a/dLKrpkgV

Make sure the project is set to 32 bit (x86).

As it stands the project will not compile. As with last weeks lab, you need to move the relevant files into src, include and lib directories, alter the file locations in the vcxproj file so that the files are visible in visual studio and adjust the project properties so the code builds. Links to the extra libraries are already added under project properties-> linker->additional libraries but you will need to add the glad and khr input files and the glfw headers and library.

Although there is quite a lot of ‘boilerplate’ code, it doesn’t actually do anything other than draw a black screen. Take a look at the source code to see how the GLFW code has been separated from the ‘game’ code using the Game Class and the IEngineCore Interface.

Let’s make it do something. In the Interface class IEngineCore, create the following method:

virtual void renderColouredBackground(float r, float g, float b) = 0;

Implement the code in the GLFW\_EngineCore class, using OpenGL code to clear the screen used last week.

In the void Game::render() function use the m\_engineInterfacePtr member pointer variable to call the newly created function, getting the rgb values from m\_playerBackground

Test if the code now works by pressing the a and z keys on the keyboard to change the background colour.

Using the command pattern create suitable new InputCommands that change the Green and Blue variables of the background colour using other key presses (search online for Ascii values of the keys).

What ways can you think of to improve how this functions for the developer and also the end user (hint – press z 10 times)?

Commit the changes on your local git repo and then push the changes to the online repository.