```cpp
#include "Window.h"



Window::Window()
{
    width = 800;
    height = 600;
}

Window::Window(GLint windowWidth, GLint windowHeight)
{

    width = windowWidth;
    height = windowHeight;

}

int Window::initialise()
{
    // initialization

        if (!glfwInit())
        {
            printf("GLFW not working initializing ");
            glfwTerminate();
            return 1; //means fails


        }


    //setup GLFW window properties
    //OpemGL Version

    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);

    mainWindow = glfwCreateWindow(width, height, "Test Window", NULL, NULL);
    if (!mainWindow)

    {
        printf("GLFW window creation failed");
        glfwTerminate();
        return 1;

    }

    //int bufferWidth, bufferHeight;
    glfwGetFramebufferSize(mainWindow, &bufferWidth, &bufferHeight);
```

```
53
54      //set context for GLFW
55
56      glfwMakeContextCurrent(mainWindow);
57
58      //Allow modern extension features
59      glewExperimental = GL_TRUE;
60      //************  GLenum error = glewInit(); *********//
61      //************ if (error! = GLEW_OK) ***************//
62      if (glewInit() != GLEW_OK)
63      {
64          printf("GLEW initi... failed ");
65          glfwDestroyWindow(mainWindow);
66          glfwTerminate();
67
68          return 1;
69      }
70
71      //enable depth
72      glEnable(GL_DEPTH_TEST);
73
74      //setup Viewport Size
75      glViewport(0, 0, bufferWidth, bufferHeight);
76
77  }
78
79  Window::~Window()
80  {
81      //if window destroyed terminate glfw
82      glfwDestroyWindow(mainWindow);
83      glfwTerminate();
84  }
85
```