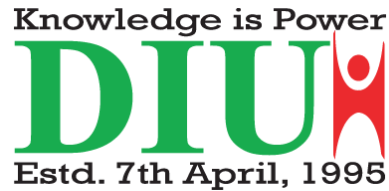# DHAKA INTERNATIONAL UNIVERSITY

Department of Computer Science & Engineering

LAB REPORT

COURSE NAME   **:** Computer Graphics and Multimedia Lab

COURSE CODE   **:** CSE-406

REPORT NO   **:** 01

REPORT NAME   **:** Introduction to OpenGL with GLFW

SUBMITTED BY	SUBMITTED TO
NAME : Md. Soumik Hasan **ROLL** : 34 **BATCH** : D-72	**Umme Niraj Mahi** **Lecturer** **Department of CSE** **Dhaka International University**

DATE OF SUBMISSION : 17 / 09 / 2025

Objectives

1. To initialize and set up a basic OpenGL environment using GLFW
2. To create and configure an OpenGL rendering window
3. To implement basic input handling for window management
4. To understand the basic structure of an OpenGL application loop
5. To become familiar with modern OpenGL context creation (version 3.3 Core Profile)

Introduction

OpenGL functions as a cross-platform application programming interface, which enables users to display both two-dimensional and three-dimensional vector graphics. This laboratory assignment teaches students how to create a basic OpenGL setup by using GLFW, which offers straightforward tools to build windows and contexts and manage user input. The code shows the essential procedures to start OpenGL initialization and window creation and rendering loop setup and keyboard input processing.

The implementation uses the core profile of OpenGL 3.3, which is a modern version that removes deprecated functionality. This approach ensures compatibility with current graphics hardware while maintaining a clean and efficient API.

Tools Required

1. Python
2. PyOpenGL package
3. GLFW library with Python bindings
4. Modern graphics card with OpenGL 3.3
5. Code editor or IDE (VS Code)

Code Explanation

The provided code contains multiple essential parts, which function as its core structure.

1. **GLFW Initialization:** The initialize_glfw() function runs the GLFW library startup process, which allows developers to launch new windows and contexts.
2. **Window Creation:** The create_window() function starts by defining window settings that enable OpenGL 3.3 Core Profile context creation before generating an 800x600 pixel window.
3. **Input Handling:** The key_callback() function processes keyboard events to detect when users press the ESC key, which triggers window closure.

4. **Main Loop:** The main() function serves as the central point that connects all components by starting GLFW and generating the window and setting the callback and executing the main rendering loop, which refreshes the screen with a blue-green background color every frame.

Output

When executed, the program:
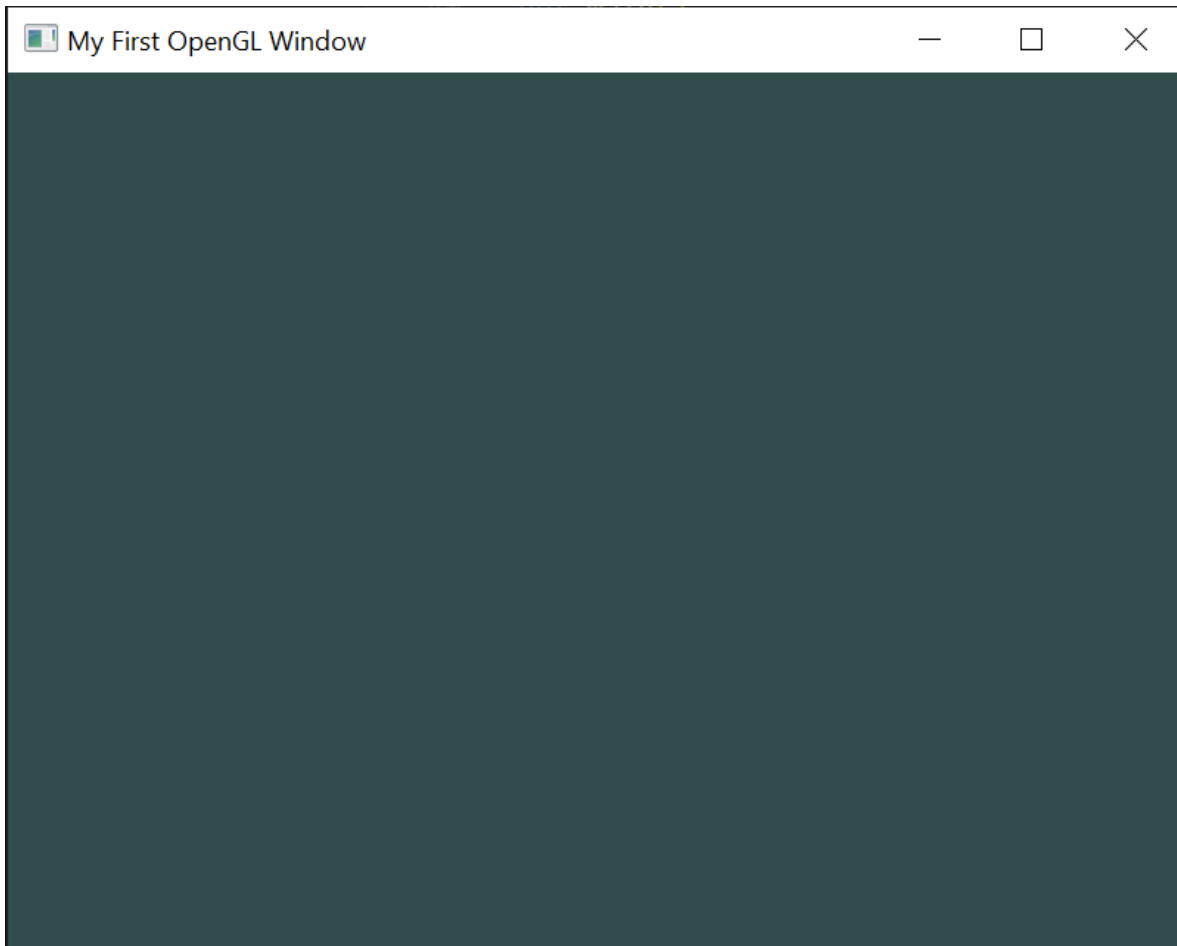
Console output example:

Glfw initialized successfully
Window created: 800x600
OpenGL Version: 3.3.0 - Build 31.0.101.2121
OpenGL Renderer: Intel(R) UHD Graphics 620
ESC pressed - closing window

Conclusion

During the lab session, I successfully implemented a basic OpenGL application using GLFW in Python. I initialized the GLFW library to open a window with OpenGL 3.3 Core Profile context and developed a keyboard callback function to process user input. The program showed a teal window, while users could exit by pressing the ESC key. The exercise gave me hands-on practice with a modern OpenGL setup and taught me how to build the basic framework for a graphics application main loop. The implementation worked as expected without any errors.