



CHITTAGONG UNIVERSITY OF ENGINEERING & TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Course No: CSE-458

Course Title: Computer Graphics (Sessional)

Lab Report -01

GLUT Toolkit Installation and Basic OpenGL Setup

Submitted By

Lailatul Husna Monika

ID: 1904009

Section: A

Level: 4

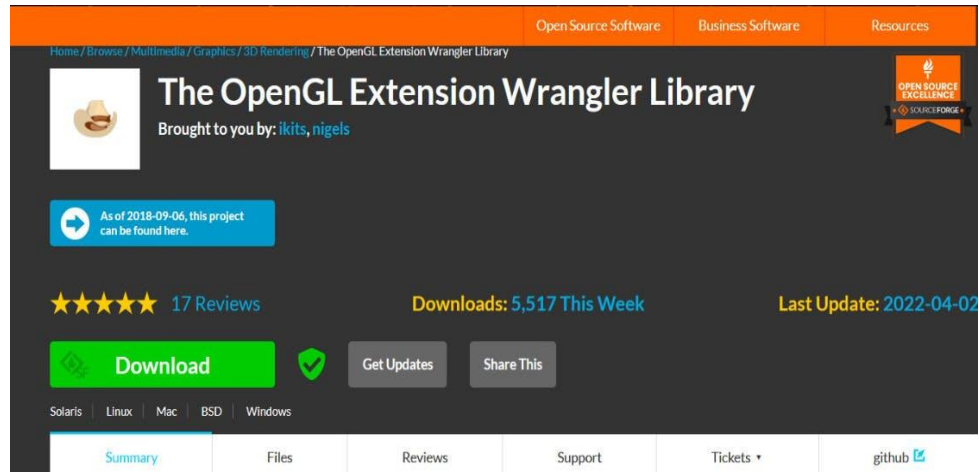
Term: 2

Remarks

--

Installation Steps:

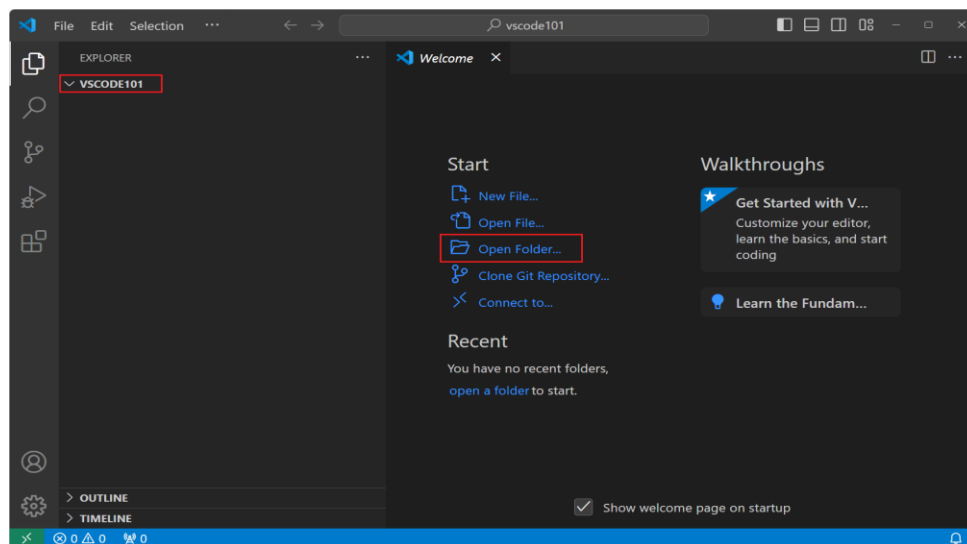
OpenGL ZIP file download from here:



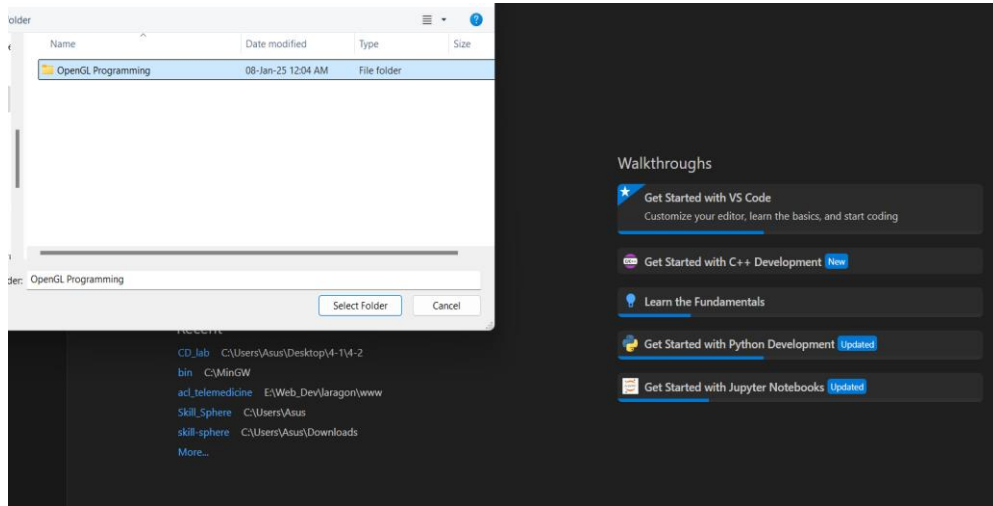
Extract the ZIP file:

Name	Date modified	Type	Size
Computer-Graphics-main	07-Jan-25 10:13 PM	File folder	
Computer-Graphics-main_Gluttoolkit_inst...	07-Jan-25 10:13 PM	Compressed (zipp...	190,395 KB

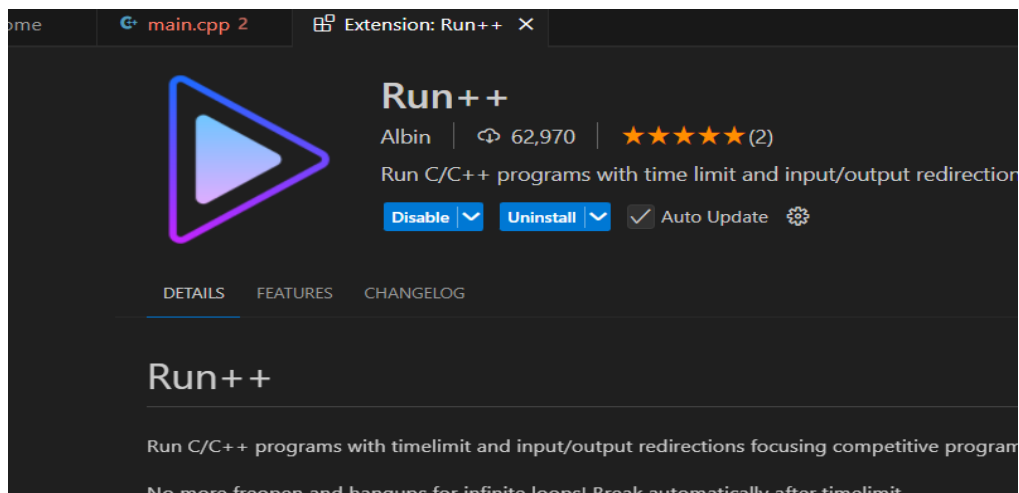
Open the VS Code:



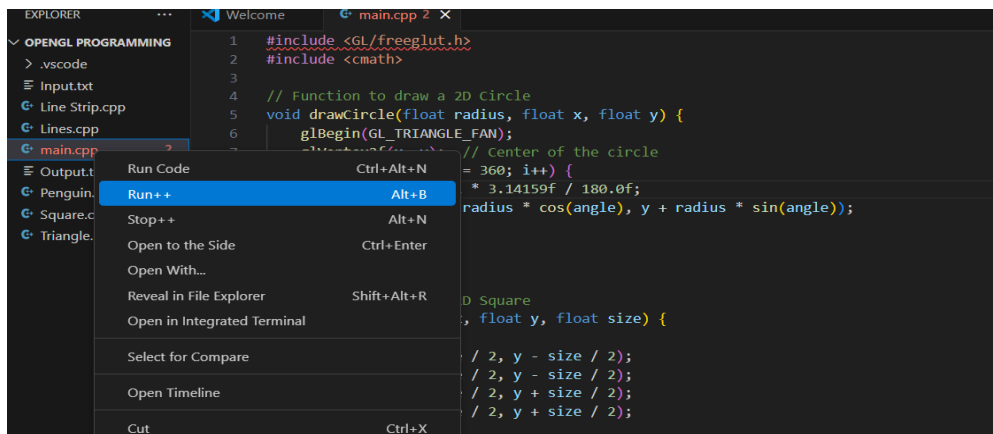
Open the OpenGL programming folder in VS Code:



Install the Run++ extension for running the sample program:



Run a main.cpp sample program:



GLUT Sample Code:

```
#include <GL/freeglut.h>
#include <cmath>

void drawCircle(float radius, float x, float y) {
    glBegin(GL_TRIANGLE_FAN);
    glVertex2f(x, y); // Center of the circle
    for (int i = 0; i <= 360; i++) {
        float angle = i * 3.14159f / 180.0f;
        glVertex2f(x + radius * cos(angle), y + radius * sin(angle));
    }glEnd();
}

void drawSquare(float x, float y, float size) {
    glBegin(GL_QUADS);
    glVertex2f(x - size / 2, y - size / 2);
    glVertex2f(x + size / 2, y - size / 2);
    glVertex2f(x + size / 2, y + size / 2);
    glVertex2f(x - size / 2, y + size / 2);
    glEnd();
}

void drawLine(float x1, float y1, float x2, float y2) {
    glBegin(GL_LINES);
    glVertex2f(x1, y1);
    glVertex2f(x2, y2);
    glEnd();
}

void draw3DTriangle() {
    glBegin(GL_TRIANGLES);
    glVertex3f(0.0f, 1.0f, 0.0f);
    glVertex3f(-1.0f, -1.0f, 0.0f);
    glVertex3f(1.0f, -1.0f, 0.0f);
    glEnd();
}

void draw3DCylinder() {
    glPushMatrix();
    glRotatef(-90.0f, 1.0f, 0.0f, 0.0f);
    glutSolidCylinder(0.5, 1.0, 32, 32);
    glPopMatrix();
}

void setupOpenGL() {
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glLoadIdentity();
}

void display() {
    setupOpenGL();
    glColor3f(1.0f, 0.0f, 0.0f);
```

```

drawCircle(0.5f, -1.5f, 0.0f);
glColor3f(0.0f, 1.0f, 0.0f);
drawSquare(1.0f, 0.0f, 1.0f);
glColor3f(0.0f, 0.0f, 1.0f);
drawLine(-1.0f, -1.0f, 1.0f, 1.0f);
glColor3f(1.0f, 1.0f, 0.0f);
glPushMatrix();
glTranslatef(-1.0f, 1.5f, 0.0f);
draw3DTriangle();
glPopMatrix();
glColor3f(0.0f, 1.0f, 1.0f);
glTranslatef(1.5f, 1.5f, 0.0f);
draw3DCylinder();
glPopMatrix();
glutSwapBuffers();
}

void reshape(int w, int h) {
    glViewport(0, 0, w, h);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(-3.0, 3.0, -3.0, 3.0, -3.0, 3.0);
    glMatrixMode(GL_MODELVIEW);
}

int main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB | GLUT_DEPTH);
    glutInitWindowSize(800, 600);
    glutCreateWindow("Basic 2D & 3D Shapes with GLUT");
    glEnable(GL_DEPTH_TEST);
    glutDisplayFunc(display);
    glutReshapeFunc(reshape);
    glutMainLoop();
    return 0;
}

```

Output of the Code:

