

```
1  #include "Mesh.h"
2
3  Mesh::Mesh()
4  {
5      VAO = 0;
6      VBO = 0;
7      IBO = 0;
8      indexCount = 0;
9
10 }
11
12
13 void Mesh::createMesh(GLfloat *vertices, unsigned int *indices, unsigned int    ↗
    numOfVertices, unsigned int numOfIndices)
14
15 {
16     indexCount = numOfIndices;
17
18     glGenVertexArrays(1, &VAO);
19     glBindVertexArray(VAO);
20
21
22     glGenBuffers(1, &IBO); //buffer drwst akain
23     glBindBuffer(GL_ELEMENT_ARRAY_BUFFER, IBO); //bindy buffer akain
24     glBufferData(GL_ELEMENT_ARRAY_BUFFER, sizeof(indices[0]) * numOfIndices,    ↗
        indices, GL_STATIC_DRAW); // zanyary akaina nawi
25
26     glGenBuffers(1, &VBO); //buffer drwst akain
27     glBindBuffer(GL_ARRAY_BUFFER, VBO); //bindy buffer akain
28     glBufferData(GL_ARRAY_BUFFER, sizeof(vertices[0]) * numOfVertices, vertices,    ↗
        GL_STATIC_DRAW); // zanyary akaina nawi
29
30     glVertexAttribPointer(0, 3, GL_FLOAT, GL_FALSE, 0, 0); // sifaatakany anwsin
31     glEnableVertexAttribArray(0); //eshy pe akain
32
33     glBindBuffer(GL_ARRAY_BUFFER, 0); //UNBIND
34     glBindBuffer(GL_ELEMENT_ARRAY_BUFFER, 0); //IMPORTANT:NOTE YOU SHOULD UNBIND    ↗
        VAO AFTER YOU UNBIND IBO
35     glBindVertexArray(0); //UNBIND
36 }
37
38 void Mesh::RenderMesh() { //called glDrawElements
39
40     glBindVertexArray(VAO); // bakary aheninawa Bindy akain hamw jare ka bakary    ↗
        ahenin
41
42     glBindBuffer(GL_ELEMENT_ARRAY_BUFFER, IBO); //some times doesn't need to bind    ↗
        this IBO, this for some graphic card bugs
43     glDrawElements(GL_TRIANGLES, indexCount, GL_UNSIGNED_INT, 0); // wena akeshin
44     //When glDrawElements is called, it uses count sequential elements from an    ↗
        enabled array, starting at indices to construct a sequence of geometric    ↗
        primitives.
```

```
45
46     glBindBuffer(GL_ELEMENT_ARRAY_BUFFER, 0); //unbind IBO ka eshman penama
47     glBindVertexArray(0); //unbind VAO ka eshman penama
48
49 }
50
51
52 void Mesh::clearMesh()
53 {
54     if (IBO != 0)
55     {
56         glDeleteBuffers(1, &IBO);
57         IBO = 0;
58     }
59
60     if (VBO != 0)
61     {
62         glDeleteBuffers(1, &VBO);
63         VBO = 0;
64     }
65
66     if (VAO != 0)
67     {
68         glDeleteVertexArrays(1, &VAO);
69         VAO = 0;
70     }
71     indexCount = 0;
72 }
73
74
75
76 Mesh::~Mesh()
77 {
78
79     clearMesh();
80 }
81
```