Ex. No. : 6 **Date:** 25/09/2025

Register No.: 231701059 **Name: Surya R**

Create and Render 3D Objects Using Polygons and Apply Basic Color and Shading Techniques

AIM:

To create and render 3D objects (such as **cube** or **sphere**) using polygonal modeling and apply:

- **Flat shading**
- **Basic color techniques**

Procedure:

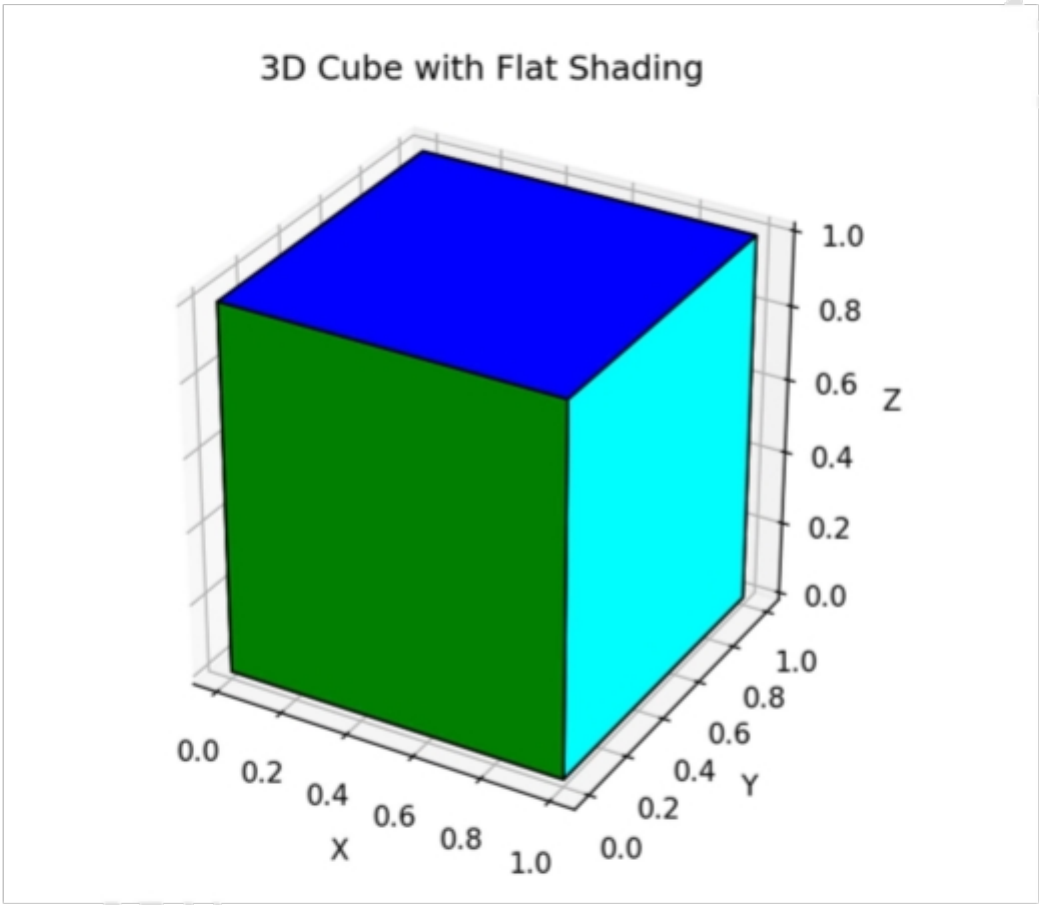
1. Define the 3D object using polygon faces (e.g., triangles or quadrilaterals).
2. Assign color values to faces or vertices.
3. Apply simple shading models:
    - **Flat shading** assigns one color per face.
    - **Vertex coloring** blends colors between corners.
4. Use 3D plotting to render the object with shading.
5. Visualize the effect of light using normal approximation (optional).

Program:

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d.art3d import Poly3DCollection
import numpy as np
vertices = np.array([[0,0,0], [1,0,0], [1,1,0], [0,1,0],
                     [0,0,1], [1,0,1], [1,1,1], [0,1,1]])
faces = [[vertices[j] for j in [0,1,2,3]],
         [vertices[j] for j in [4,5,6,7]],
         [vertices[j] for j in [0,1,5,4]],
         [vertices[j] for j in [2,3,7,6]],
         [vertices[j] for j in [1,2,6,5]],
         [vertices[j] for j in [4,7,3,0]]]
colors = ['red', 'blue', 'green', 'yellow', 'cyan', 'orange']
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
poly3d = Poly3DCollection(faces, facecolors=colors, edgecolors='black', linewidths=1)
ax.add_collection3d(poly3d)
```

IT3511 - LAB MANUAL

```
# Set view
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')
ax.set_title('3D Cube with Flat Shading')
ax.set_box_aspect([1,1,1])
plt.show()
```



Result:

3D polygonal objects were successfully created and rendered. Basic color and flat shading techniques were applied to enhance visual representation.