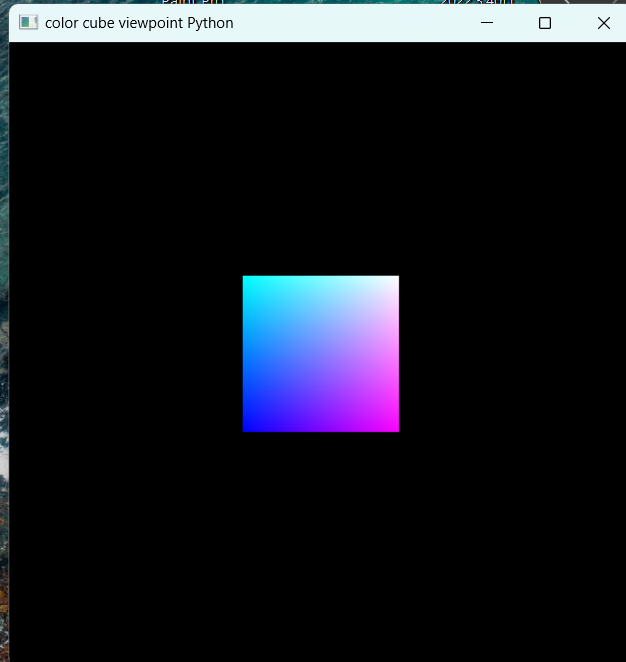
**컴퓨터그래픽스 Lab05 보고서**

|  |  |  |
| --- | --- | --- |
| **학번** | **이름** | **분반** |
| 2312282 | 임다희 | 003 |

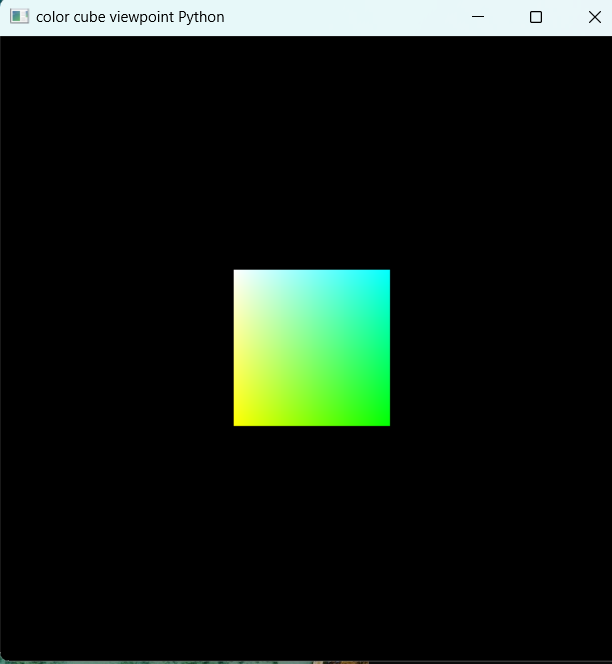
**[과제] 정의된 정육면체의 Front View, Top View, Side View, Perspective View를 보여주는 코드 작성**

**결과**

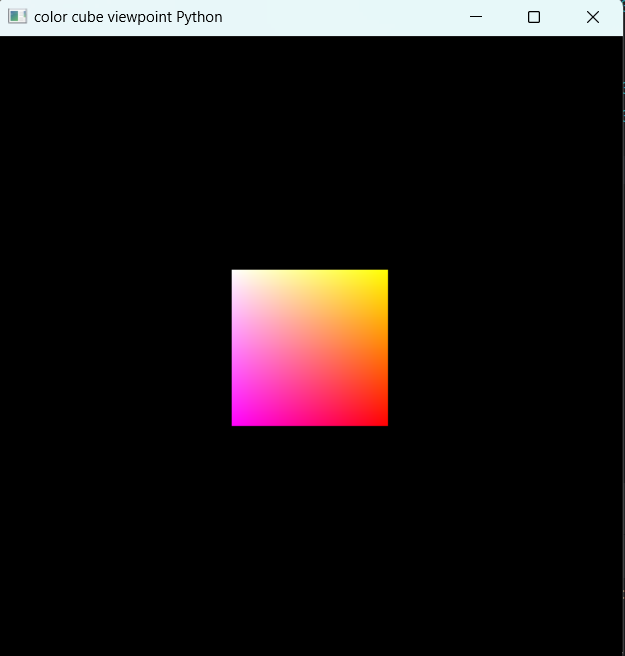


< Front view

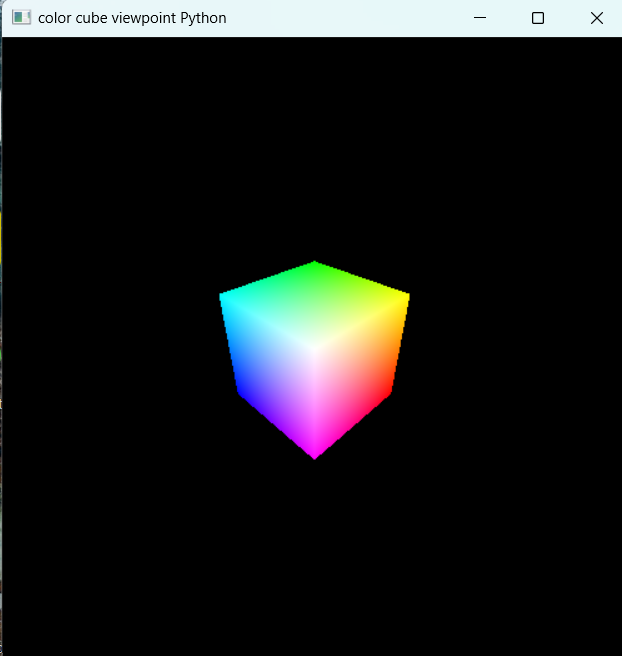
**코드**

****

< Top view

****

< Side view

****

< Perspective view

**코드**

from OpenGL.GL import \*

from OpenGL.GLUT import \*

from OpenGL.GLU import \*

vertices = ((-1.0,-1.0,-1.0),(1.0,-1.0,-1.0),

(1.0,1.0,-1.0), (-1.0,1.0,-1.0), (-1.0,-1.0,1.0),

(1.0,-1.0,1.0), (1.0,1.0,1.0), (-1.0,1.0,1.0))

colors = ((0.0,0.0,0.0),(1.0,0.0,0.0),

(1.0,1.0,0.0), (0.0,1.0,0.0), (0.0,0.0,1.0),

(1.0,0.0,1.0), (1.0,1.0,1.0), (0.0,1.0,1.0))

<--- 변수 myview 값에 따라 카메라가 큐브를 보여주는 시점이 달라진다.

myview =3

def polygon( a, b, c , d):

glBegin(GL\_POLYGON)

glColor3fv(colors[a])

glVertex3fv(vertices[a])

glColor3fv(colors[b])

glVertex3fv(vertices[b])

glColor3fv(colors[c])

glVertex3fv(vertices[c])

glColor3fv(colors[d])

glVertex3fv(vertices[d])

glEnd()

def colorcube():

polygon(0, 1, 2, 3)

polygon(0, 1, 5, 4)

polygon(4, 5, 6, 7)

polygon(5, 1, 2, 6)

polygon(6, 2, 3, 7)

polygon(4, 7, 3, 0)

def MyDisplay():

global myview

glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT)

glLoadIdentity()

<--- myview가 0일때 front view로, 카메라의 위치는 0,0,5 (front view는 z축 방향으로 물체를 보기에 z방향으로 5만큼의 위치를 가진다), 카메라가 보는 위치는 0,0,0 (큐브의 중심 위치) , up vector는 0,1,0이다. (z축에서 수직으로 위쪽 방향인 축을 y축으로 보고 y에 1 값을 준다)

myview가 1일 때는 top view로, 카메라의 위치는 0,5,0(top view는 y축 방향으로 물체를 보기에 y방향으로 5만큼의 위치를 가진다), 카메라가 보는 위치는 0,0,0(큐브의 중심 위치), up vector는 0,0,1이다. (y축에서 수직으로 위쪽 방향인 축을 z축으로 보고 z에 1 값을 준다)

myview가 2일 때는 side view로, 카메라의 위치는 5,0,0(side view는 x축 방향으로 물체를 보기에 x방향으로 5만큼의 위치를 가진다), 카메라가 보는 위치는 0,0,0(큐브의 중심 위치), up vector는 0,1,0이다. (x축에서 수직으로 위쪽 방향인 축을 y축으로 보고 y에 1 값을 준다)

myview가 3일 때는 perspective view로, 카메라의 위치는 3,3,3, (x,y,z방향으로 모두 적당히 떨어진 위치로 지정) 카메라가 보는 위치는 0,0,0(큐브의 중심 위치), up vector는 0,1,0이다. (물체가 y축 방향으로 똑바로 서있게 함)

if myview == 0: # front view

gluLookAt(0.0,0.0,5.0,0.0,0.0,0.0,0.0,1.0,0.0)

elif myview == 1: # top view

gluLookAt(0.0,5.0,0.0,0.0,0.0,0.0,0.0,0.0,1.0)

elif myview == 2: # side view

gluLookAt(5.0,0.0,0.0,0.0,0.0 ,0.0 ,0.0 ,1.0 , 0.0)

elif myview == 3: # perspective view

gluLookAt(3,3,3,0,0,0,0,1,0)

colorcube()

glutSwapBuffers()

def myReshape(w, h):

glViewport(0, 0, w, h)

glMatrixMode(GL\_PROJECTION)

glLoadIdentity()

# glFrustum (left, right, bottom, top, near distance, far distance)

if w <= h:

glFrustum(-2.0, 2.0, -2.0 \* float(h)/ float(w), 2.0\* float(h) / float(w), 2.0, 20.0)

else:

glFrustum(-2.0, 2.0, -2.0 \* float(w)/ float(h), 2.0\* float(w) / float(h), 2.0, 20.0)

glMatrixMode(GL\_MODELVIEW)

def main():

glutInit(sys.argv)

glutInitDisplayMode(GLUT\_DOUBLE | GLUT\_RGB | GLUT\_DEPTH)

glutInitWindowSize(500, 500)

glutCreateWindow('color cube viewpoint Python')

glutReshapeFunc(myReshape)

glutDisplayFunc(MyDisplay)

glutAttachMenu(GLUT\_RIGHT\_BUTTON)

glEnable(GL\_DEPTH\_TEST)

glutMainLoop()

if \_\_name\_\_ == "\_\_main\_\_":

main()