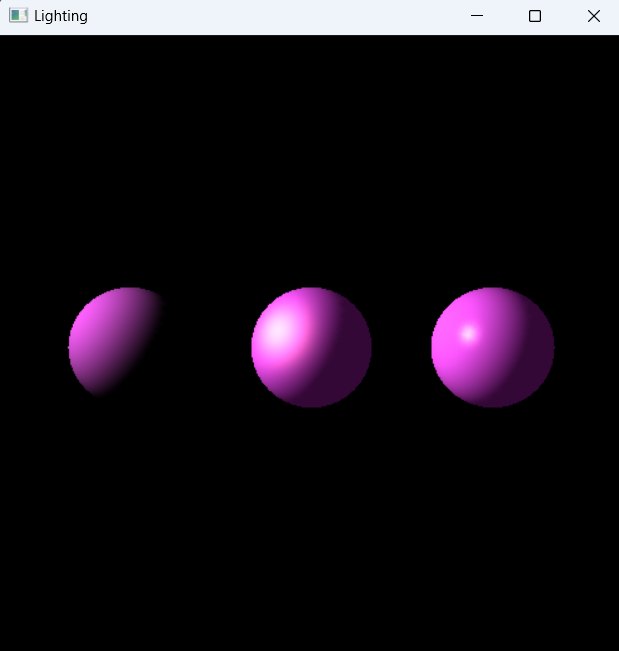
**컴퓨터그래픽스 Lab07 보고서**

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

**[과제]** 광원의 위치, 물체, 특성을 변화시켜 3개의 원 그리기

**결과**



**코드**

from OpenGL.GL import \*

from OpenGL.GLUT import \*

from OpenGL.GLU import \*

myview =0

light\_ambient = (0.2, 0.1, 0.1, 1.0)

# Ia : 광원의 주변광 세기를 조절한다. r,g,b,알파값이 각각 0.2,0.1,0.1,1.0

light\_diffuse = (1, 0.75, 1, 1.0)

# Id : 광원의 확산광 세기를 조절한다. r,g,b,알파값이 각각 1,0.75,1.0,1.0.

light\_specular = (1.0, 0.75, 0.5, 1.0)

# Is : 광원의 경면광 세기를 조절한다. r,g,b,알파값이 각각 1,0.75,0.5,1.0.

light\_position1 = (-10.0, 5.0, -8.0, 1.0)

# 광원의 위치 첫 번째. x,y,z, 값이 각각 -10.0,5.0,-8.0,1.0

light\_position2 = (15.0, 8.0, 8.0, 1.0)

# 광원의 위치 두 번째. x,y,z, 값이 각각 15.0,8.0,8.0,1.0

no\_mat = (0.0, 0.0, 0.0, 1.0)

mat\_ambient = (0.5, 0.1, 0.7, 1.0) #Ka

# Ka : 광원의 주변광 계수를 조절한다. r,g,b,알파값이 각각 0.5,0.1,0.7,1.0

mat\_diffuse = (1.0, 0.5, 1.0, 1.0) #kd

# Kd : 광원의 확산광 계수를 조절한다. r,g,b,알파값이 각각 1.0,0.5,1.0,1.0

mat\_specular = (1.0, 0.75, 1.0, 1.0) #Ks

# Ks : 광원의 경면광 계수를 조절한다. r,g,b,알파값이 각각 1.0, 0.75,1.0,1.0

no\_shininess = 0.0 # Shininess Coefficient

# 광원의 광택 계수를 조정한다. 광택 계수가 0인 경우

low\_shininess = 10.0 # Shininess Coefficient

# 광원의 광택 계수를 조정한다. 광택 계수가 10인 경우

high\_shininess = 100 # Shininess Coefficient

# 광원의 광택 계수를 조정한다. 광택 계수가 100인 경우

def MyDisplay():

global myview

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

glLoadIdentity()

# light 넣기

glLightfv(GL\_LIGHT0, GL\_AMBIENT, light\_ambient)

glLightfv(GL\_LIGHT0, GL\_DIFFUSE, light\_diffuse)

glLightfv(GL\_LIGHT0, GL\_SPECULAR, light\_specular)

glLightfv(GL\_LIGHT0, GL\_POSITION, light\_position1)

#주변광, 확산광, 경면광을 넣는다.

# 사전에 정의한 광원 세기 계수를 적용하고, 광원의 위치는 첫 번째 값으로 지정한다.

gluLookAt(0.0, 0.0, 13.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0)

#object1

glPushMatrix()

glTranslatef(-3, 0.0, 0.0)

# 첫 번째 구체 object1에 material을 입힌다.

glMaterialfv(GL\_FRONT, GL\_AMBIENT, no\_mat)

# 주변광 반사 계수를 no\_mat, 즉 (0,0,0,1)으로 지정한다.

glMaterialfv(GL\_FRONT, GL\_DIFFUSE, mat\_diffuse)

# 확산광 반사 계수를 mat\_diffuse ()으로 지정한다.

glMaterialfv(GL\_FRONT, GL\_SPECULAR, no\_mat)

# 격면광 반사 계수를 no\_mat, 즉 (0,0,0,1)으로 지정한다.

glMaterialf(GL\_FRONT, GL\_SHININESS, no\_shininess)

# 광원의 광택 계수를 no\_shininess (0.0)으로 지정한다.

glutSolidSphere(1.0, 40, 40)

glPopMatrix()

#object2

# 두 번째 구체 object2에 material을 입힌다.

glPushMatrix()

glMaterialfv(GL\_FRONT, GL\_AMBIENT, mat\_ambient)

glMaterialfv(GL\_FRONT, GL\_DIFFUSE, mat\_diffuse)

glMaterialfv(GL\_FRONT, GL\_SPECULAR, mat\_specular)

glMaterialf(GL\_FRONT, GL\_SHININESS, low\_shininess)

glutSolidSphere(1.0, 40, 40)

glPopMatrix()

#object3

# 두 번째 구체 object3에 material을 입힌다.

glPushMatrix()

glTranslatef(3, 0.0, 0.0)

glMaterialfv(GL\_FRONT, GL\_AMBIENT, mat\_ambient)

glMaterialfv(GL\_FRONT, GL\_DIFFUSE, mat\_diffuse)

glMaterialfv(GL\_FRONT, GL\_SPECULAR, mat\_specular)

glMaterialf(GL\_FRONT, GL\_SHININESS, high\_shininess)

glutSolidSphere(1.0, 40, 40)

glPopMatrix()

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), 5.0, 30.0)

glMatrixMode(GL\_MODELVIEW)

def main():

glutInit(sys.argv)

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

glutInitWindowSize(500, 500)

glutCreateWindow('Lighting')

glEnable(GL\_DEPTH\_TEST)

glEnable(GL\_LIGHTING)

glEnable(GL\_LIGHT0)

glEnable(GL\_NORMALIZE)

glShadeModel(GL\_SMOOTH)

glutReshapeFunc(myReshape)

glutDisplayFunc(MyDisplay)

glutMainLoop()

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

main()