

## LAB SET 7

Write a program to demonstrate the OpenGL transformation functions with any of the suitable primitive for the following transformations

- a. Translate
- b. Scale
- c. Rotate

```
//Program
#include <stdio.h>
#include <stdlib.h>
#include <GL/glut.h>
#include <math.h>
/* Demonstration of Transformations using APIs with Keyboard interfacing */

void display()
{
    glClear(GL_COLOR_BUFFER_BIT);
    glBegin(GL_POLYGON);
    glVertex2f(-2.0, -2.0);
    glVertex2f(2.0, -2.0);
    glVertex2f(2.0, 2.0);
    glVertex2f(-2.0, 2.0);
    glEnd();
    glFlush();
}

void init()
{
    glViewport(0,0,500,500);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluOrtho2D(-30.0,30.0,-30.0,30.0);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
    glColor3f(0.5,0.5,1.0);
    glClearColor(1.0,1.0,1.0,0.0);
}

void mykeys(unsigned char key,int x, int y)
{
    switch(key)
    {
        case 'l':          //left
            glTranslatef(-2.0,0.0f,0.0f);
            break;
    }
}
```

```

        case 'r':           //right
                            glTranslatef(2.0,0.0f,0.0f);
                            break;

        case 'u':           //up
                            glTranslatef(0.0f,2.0,0.0f);
                            break;

        case 'd':           //down
                            glTranslatef(0.0,-2.0,0.0);
                            break;

        case 'i':           //increase-size
                            glScalef(1.5,1.5,1.5);
                            break;

        case 'D':           //decrease-size
                            glScalef(-0.5,-0.5,-0.5);
                            break;

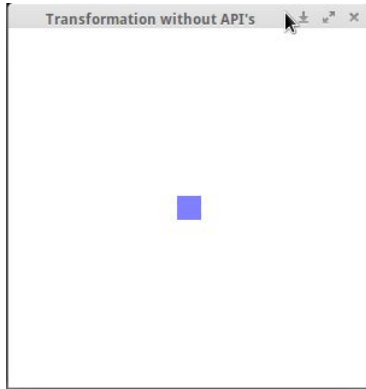
        case 'R':           //rotate-right
                            glRotatef(10,1.0,1.0,0.0);
                            break;

        case 'L':           //rotate-left
                            glRotatef(-10,1.0,0.0,0.0);
                            break;
    }
    glutPostRedisplay();
}

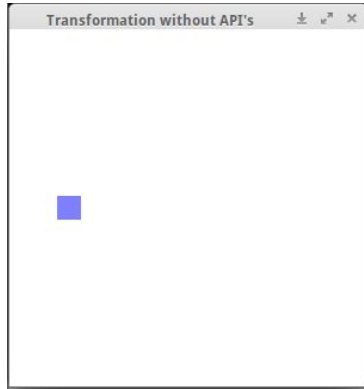
int main(int argc,char **argv)
{
    glutInit(&argc,argv);
    glutInitDisplayMode(GLUT_RGB|GLUT_SINGLE);
    glutCreateWindow("Transformation without API's");
    glutDisplayFunc(display);
    glutKeyboardFunc(mykeys);
    init();
    glutMainLoop();
    return 0;
}

```

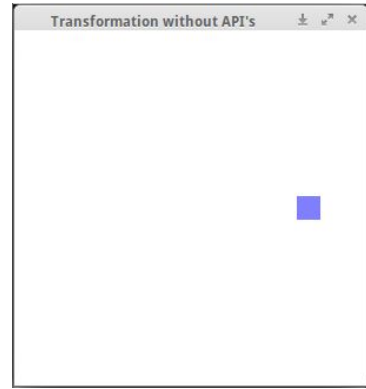
## OUTPUT



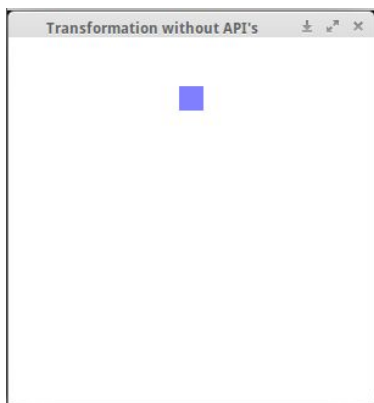
SCREENSHOT 1



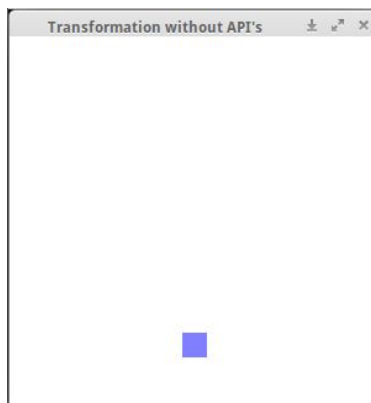
SCREENSHOT 2



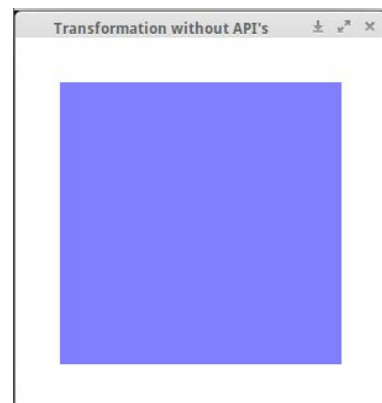
SCREENSHOT 3



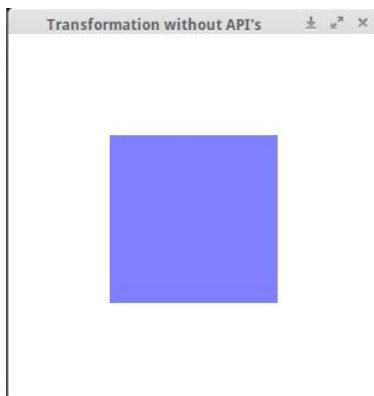
SCREENSHOT 4



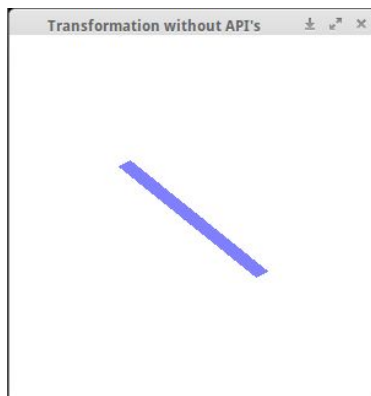
SCREENSHOT 5



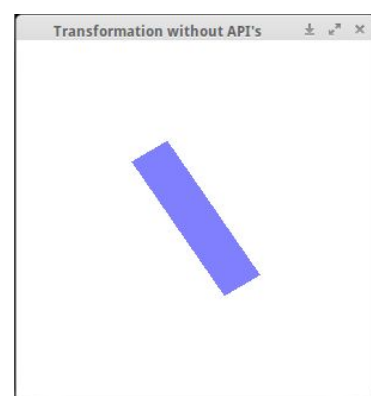
SCREENSHOT 6



SCREENSHOT 7



SCREENSHOT 8



SCREENSHOT 9