Instituto Tecnológico de Culiacán

UNIDAD 4

MACBETH ADOLFO CARRILLO IBARRA

Graficación

GRUPO: EB01

Portafolio.

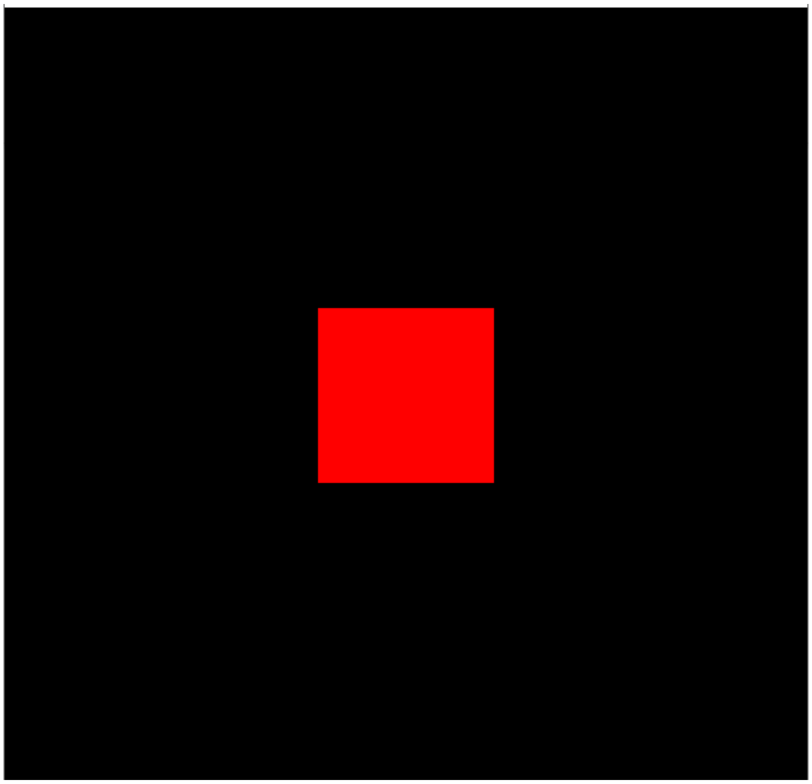
HORA: 04:00p.m - 05:00p.m

MAESTRA: CESAR ALFREDO SANCHEZ BELTRAN

Mi primera aplicación 3D: Un cubo.

```
import java.awt.BorderLayout;
import java.awt.GraphicsConfiguration;

import javax.media.j3d.Alpha;
import javax.media.j3d.BoundingSphere;
import javax.media.j3d.BranchGroup;
import javax.media.j3d.Canvas3D;
import javax.media.j3d.RotationInterpolator;
import javax.media.j3d.TransformGroup;
import javax.swing.JFrame;
import javax.swing.JPanel;

import com.sun.j3d.utils.geometry.ColorCube;
import com.sun.j3d.utils.universe.SimpleUniverse;

public class App3D extends JPanel {
    public App3D(){
        GraphicsConfiguration config = 
SimpleUniverse.getPreferredConfiguration();
                Canvas3D canvas3D = new Canvas3D(config);
        setLayout(new BorderLayout());
        add(canvas3D);
        SimpleUniverse universo = new SimpleUniverse(canvas3D);
        universo.getViewingPlatform().setNominalViewingTransform();

    BranchGroup escena = crearGrafoEscena();
    escena.compile();
    universo.addBranchGraph(escena);
    }
    public BranchGroup crearGrafoEscena(){
        BranchGroup objetoRaiz = new BranchGroup();

        TransformGroup objetoGiro = new TransformGroup();
        objetoGiro.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
        objetoGiro.addChild(objetoGiro);

        ColorCube cubo = new ColorCube(0.2);
        objetoGiro.addChild(cubo);

        Alpha rotacionAlpha = new Alpha(-1, 4000);
        RotationInterpolator rotacion = new RotationInterpolator(rotacionAlpha, 
objetoGiro);
```

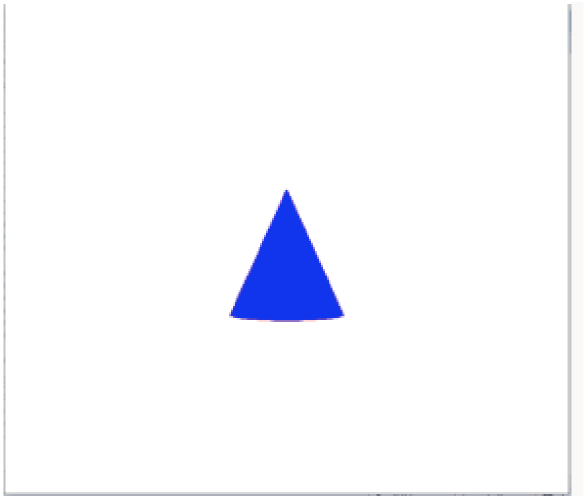
```
            rotacion.setSchedulingBounds ( new BoundingSphere());
            objetoRaiz.addChild(rotacion);

            return objetoRaiz;
        }
        public static void main(String [] args){
            System.setProperty("sun.awt.noerasebackground", "true");
            JFrame ventana = new JFrame ("TransformGroup java 3D");
            App3D panel = new App3D ();
            ventana.add(panel);
            ventana.setSize(700,700);
            ventana.setVisible(true);
            ventana.setLocationRelativeTo(null);
            ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        }

}
```

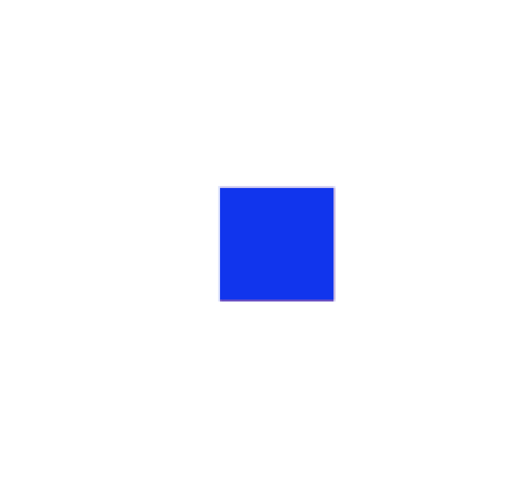


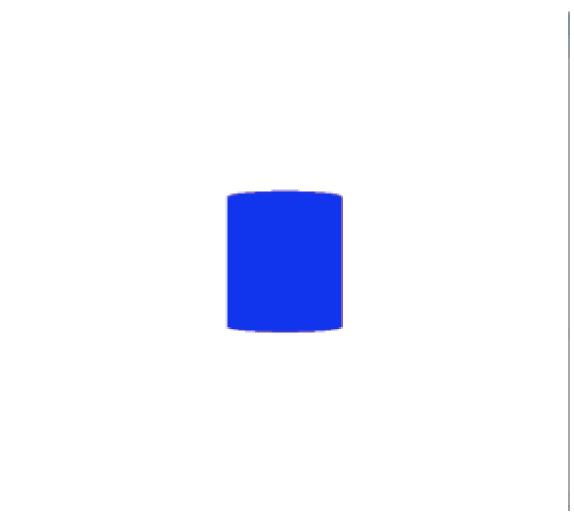
Figuras 3D: Cubo, cono, cilindro y esfera.

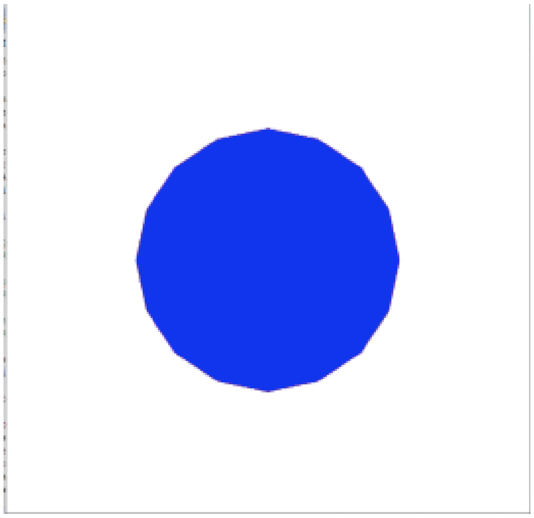
```
import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.GraphicsConfiguration;

import javax.media.j3d.Alpha;
import javax.media.j3d.Appearance;
import javax.media.j3d.Background;
import javax.media.j3d.BoundingSphere;
import javax.media.j3d.BranchGroup;
import javax.media.j3d.Canvas3D;
import javax.media.j3d.ColoringAttributes;
import javax.media.j3d.RotationInterpolator;
import javax.media.j3d.TransformGroup;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.vecmath.Color3f;

import com.sun.j3d.utils.geometry.Box;
import com.sun.j3d.utils.geometry.ColorCube;
import com.sun.j3d.utils.geometry.Cone;
import com.sun.j3d.utils.geometry.Cylinder;
import com.sun.j3d.utils.geometry.Sphere;
import com.sun.j3d.utils.universe.SimpleUniverse;

public class App3D_2 extends JPanel {
    public App3D_2(){
        GraphicsConfiguration config = SimpleUniverse.getPreferredConfiguration();
```

```
            Canvas3D canvas3D = new Canvas3D(config);
    setLayout(new BorderLayout());
    add(canvas3D);
    SimpleUniverse universo = new SimpleUniverse(canvas3D);
    universo.getViewingPlatform().setNominalViewingTransform();

BranchGroup escena = crearGrafoEscena();
escena.compile();
universo.addBranchGraph(escena);
}
public BranchGroup crearGrafoEscena(){
    BranchGroup objetoRaiz = new BranchGroup();

    Background fondo = new Background (new Color3f(Color.white));
     fondo.setApplicationBounds(new BoundingSphere());
     objetoRaiz.addChild(fondo);

    Appearance aparienciaRoja = new Appearance();
    Color3f azul = new Color3f(Color.azul);
    ColoringAttributes ca = new ColoringAttributes();
    ca.setColor(azul);

    aparienciaRoja.setColoringAttributes(ca);

    Cone cono = new Cone (0.2f, 0.5f,  aparienciaRoja);
    objetoRaiz.addChild(cono);



    return objetoRaiz;
    }

    public static void main(String [] args){

        System.setProperty("sun.awt.noerasebackground", "true");

        JFrame ventana = new JFrame ("TransformGroup java 3D_2");

        App3D_2 panel = new App3D_2 ();

        ventana.add(panel);

        ventana.setSize(700,700);

        ventana.setVisible(true);

        ventana.setLocationRelativeTo(null);

        ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    }

}
```





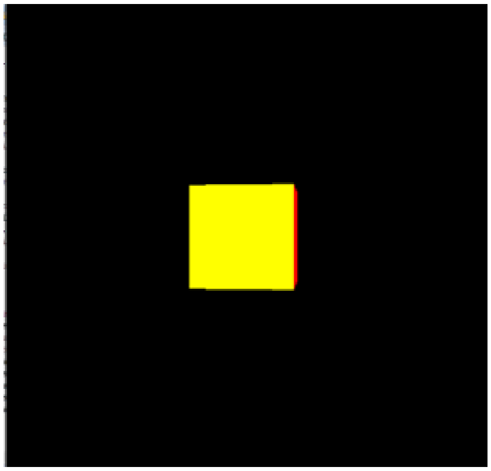


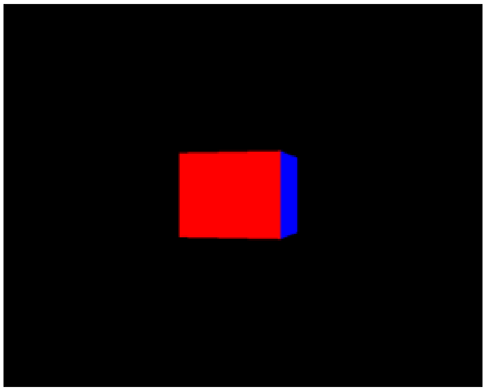


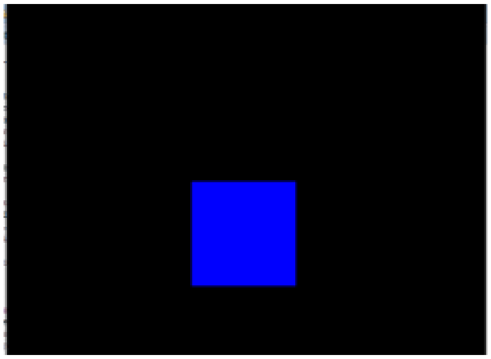
TransformGroup java 3D: Giro del cubo.

```
import java.awt.BorderLayout;

import java.awt.Color;

import java.awt.GraphicsConfiguration;

import javax.media.j3d.BranchGroup;

import javax.media.j3d.Canvas3D;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.vecmath.Color3f;
import com.sun.j3d.utils.geometry.ColorCube;

import com.sun.j3d.utils.universe.SimpleUniverse;

import javax.media.j3d.Alpha;

import javax.media.j3d.Background;

import javax.media.j3d.BoundingSphere;

import javax.media.j3d.RotationInterpolator;

import javax.media.j3d.TransformGroup;


public class App3D_3 extends JPanel {

        public App3D_3() {

                GraphicsConfiguration config = SimpleUniverse.getPreferredConfiguration();

                Canvas3D canvas3D = new Canvas3D(config);


                setLayout(new BorderLayout());

                add(canvas3D);

                SimpleUniverse universo = new SimpleUniverse(canvas3D);

                universo.getViewingPlatform().setNominalViewingTransform();


                BranchGroup escena = CrearGrafoEscena();

                escena.compile();


                universo.addBranchGraph(escena);

        }


        public BranchGroup CrearGrafoEscena() {

                BranchGroup objetoRaiz = new BranchGroup();

                TransformGroup objetoGiro = new TransformGroup();

                objetoGiro.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
                objetoRaiz.addChild(objetoGiro);
                Background fondo = new Background (new Color3f(Color.white));
                 fondo.setApplicationBounds(new BoundingSphere());
                 objetoRaiz.addChild(fondo);

                ColorCube cubo = new ColorCube(0.2);
                objetoGiro.addChild(cubo);

                Alpha rotacionAlpha = new Alpha(-1, 4000);
                RotationInterpolator rotacion = new RotationInterpolator(rotacionAlpha, objetoGiro);
                rotacion.setSchedulingBounds(new BoundingSphere());
                objetoRaiz.addChild(rotacion);

                return objetoRaiz;

        }

        public static void main(String[] args) {
                System.setProperty("sun.awt.noerasebackground", "true");
                JFrame ventana = new JFrame("TransformGroup java 3D_3");
                App3D_3 panel = new App3D_3();
                ventana.add(panel);
                ventana.setSize(700, 700);
                ventana.setVisible(true);
                ventana.setLocationRelativeTo(null);
                ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        }
```

```
}
```

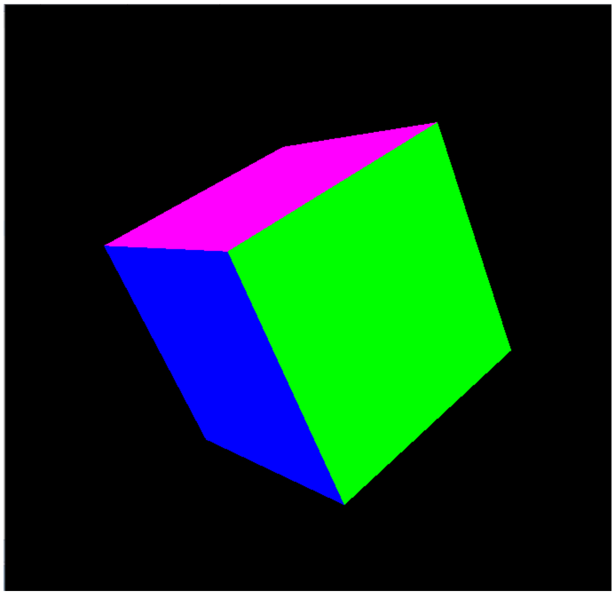


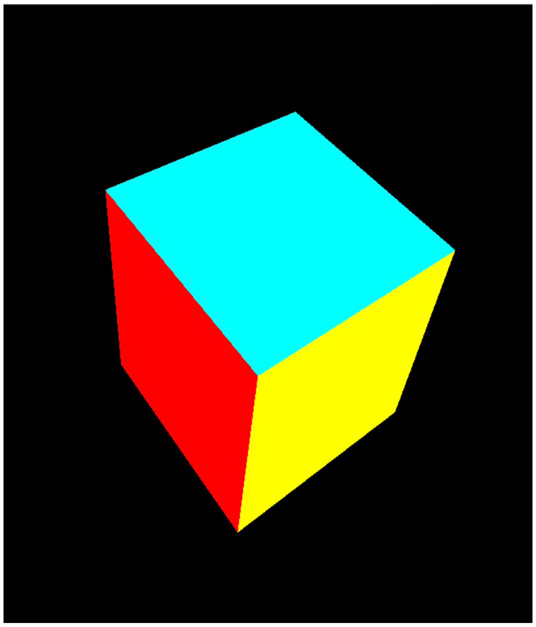




Mover un Cubo con el mouse.

```
import java.awt.BorderLayout;

import java.awt.GraphicsConfiguration;

import javax.media.j3d.BranchGroup;

import javax.media.j3d.Canvas3D;

import javax.swing.JFrame;

import javax.swing.JPanel;


import com.sun.j3d.utils.geometry.ColorCube;

import com.sun.j3d.utils.universe.SimpleUniverse;

import javax.media.j3d.Alpha;
import javax.media.j3d.BoundingSphere;

import javax.media.j3d.RotationInterpolator;

import javax.media.j3d.TransformGroup;


public class App3D_4 extends JPanel {

        SimpleUniverse universo;


        public App3D_4() {

                GraphicsConfiguration config = SimpleUniverse.getPreferredConfiguration();

                Canvas3D canvas3D = new Canvas3D(config);


                setLayout(new BorderLayout());

                add(canvas3D);


                 universo = new SimpleUniverse(canvas3D);

                universo.getViewingPlatform().setNominalViewingTransform();


                BranchGroup escena = CrearGrafoEscena();

                escena.compile();


                universo.addBranchGraph(escena);

        }


        public BranchGroup CrearGrafoEscena() {

                BranchGroup objetoRaiz = new BranchGroup();


                TransformGroup mouseGroup = new TransformGroup();

                mouseGroup.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);

                mouseGroup.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
                objetoRaiz.addChild(mouseGroup);
                ColorCube cubo = new ColorCube(0.4f);
                mouseGroup.addChild(cubo);

                MouseRotate mr = new MouseRotate();
                mr.setTransformGroup(mouseGroup);
                mr.setSchedulingBounds(new BoundingSphere (new Point3d(), 1000f));
                objetoRaiz.addChild(mr);

                return objetoRaiz;

        }

        public static void main(String[] args) {
                System.setProperty("sun.awt.noerasebackground", "true");
                JFrame ventana = new JFrame("TransformGroup java 3D");
                App3D_4 panel = new App3D_4();
                ventana.add(panel);
                ventana.setSize(700, 700);
                ventana.setVisible(true);
                ventana.setLocationRelativeTo(null);
                ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        }
}
```

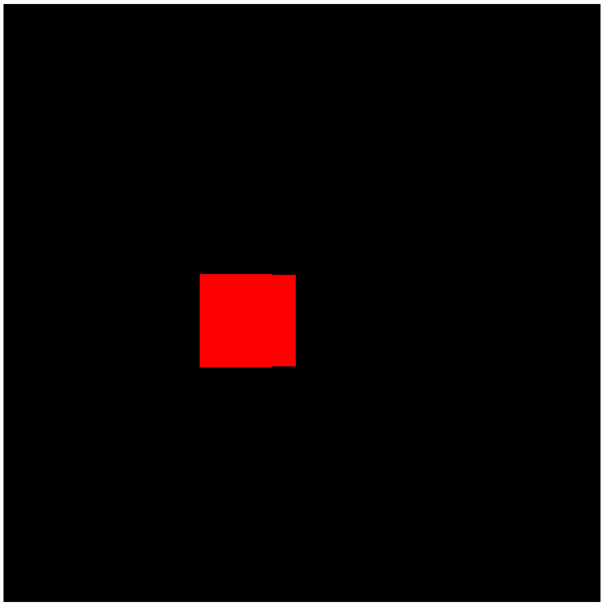


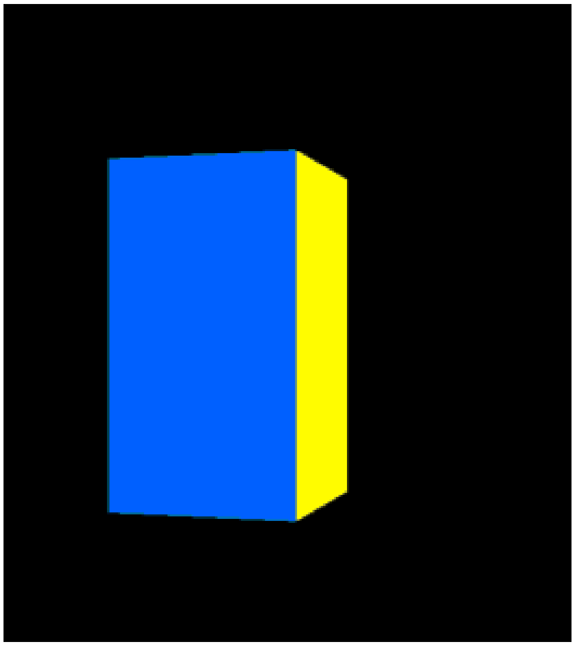


Mover un Cubo con el Teclado.

```
import java.awt.BorderLayout;

import java.awt.GraphicsConfiguration;

import javax.media.j3d.BranchGroup;

import javax.media.j3d.Canvas3D;

import javax.swing.JFrame;
```

```
}

import javax.swing.JPanel;
import javax.vecmath.Point3d;
importcom.sun.j3d.utils.behaviors.keyboard.KeyNavigatorBehavior;
import com.sun.j3d.utils.behaviors.mouse.MouseRotate;
import com.sun.j3d.utils.geometry.ColorCube;
import com.sun.j3d.utils.universe.SimpleUniverse;
import javax.media.j3d.Alpha;
import javax.media.j3d.BoundingSphere;
import javax.media.j3d.RotationInterpolator;
import javax.media.j3d.TransformGroup;

public class App3D_5 extends JPanel {
SimpleUniverse universo;

public App3D_5() {
        GraphicsConfiguration config = SimpleUniverse.getPreferredConfiguration();
        Canvas3D canvas3D = new Canvas3D(config);
   universo= new SimpleUniverse(canvas3D);

        setLayout(new BorderLayout());
        add(canvas3D);

        BranchGroup escena = CrearGrafoEscena();
        escena.compile();

         universo.getViewingPlatform().setNominalViewingTransform();
         universo.addBranchGraph(escena);

}
public BranchGroup CrearGrafoEscena() {
        BranchGroup objetoRaiz = new BranchGroup();

        TransformGroup tecladoGroup = new TransformGroup();
        objetoRaiz.addChild(tecladoGroup);

        ColorCube cubo = new ColorCube(0.3f);
        tecladoGroup.addChild(cubo);

        KeyNavigatorBehavior knb= new     KeyNavigatorBehavior
(universo.getViewingPlatform().getViewPlatformTransform());
BoundingSphere bs = new BoundingSphere (new Point3d(), 1000f);
knb.setSchedulingBounds(bs);
tecladoGroup.addChild(knb);

        return objetoRaiz;

}

public static void main(String[] args) {
        System.setProperty("sun.awt.noerasebackground", "true");
        JFrame ventana = new JFrame("TransformGroup java 3D");
        App3D_5 panel = new App3D_5();
        ventana.add(panel);
        ventana.setSize(700, 700);
        ventana.setVisible(true);
        ventana.setLocationRelativeTo(null);
        ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        }
}
```

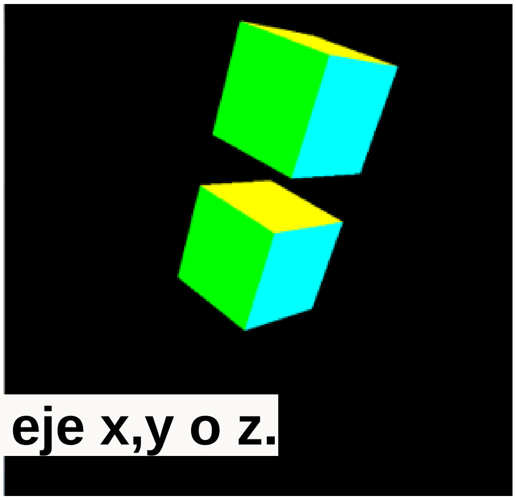




Mover dos Cubo con el mouse.

```
import java.awt.BorderLayout;
import java.awt.GraphicsConfiguration;

import javax.media.j3d.BoundingSphere;
import javax.media.j3d.BranchGroup;
import javax.media.j3d.Canvas3D;
import javax.media.j3d.Transform3D;
import javax.media.j3d.TransformGroup;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.vecmath.Point3d;
import javax.vecmath.Vector3f;

import com.sun.j3d.utils.behaviors.mouse.MouseRotate;
import com.sun.j3d.utils.geometry.ColorCube;
import com.sun.j3d.utils.universe.SimpleUniverse;

public class App3D_6 extends JPanel {
        public App3D_6() {
                GraphicsConfiguration config =
SimpleUniverse.getPreferredConfiguration();
                Canvas3D canvas3D = new Canvas3D(config);

                setLayout(new BorderLayout());
                add(canvas3D);
                SimpleUniverse universo = new SimpleUniverse(canvas3D);
                universo.getViewingPlatform().setNominalViewingTransform();

                BranchGroup escena = CrearGrafoEscena();
                escena.compile();

                universo.addBranchGraph(escena);
        }
        public BranchGroup CrearGrafoEscena() {
```

```
        BranchGroup objetoRaiz = new BranchGroup();

        TransformGroup mouseGroup = new TransformGroup();
    mouseGroup.setCapability(TransformGroup.ALLOW_TRANSFORM_READ);
    mouseGroup.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
        objetoRaiz.addChild(mouseGroup);

        //cubo 1
        Transform3D traslacion1 = new Transform3D();
        traslacion1.set(new Vector3f(-0.6f,0,0));

        TransformGroup tg1 = new TransformGroup(traslacion1);

        ColorCube cubo1 = new ColorCube(0.2f);
        tg1.addChild(cubo1);

        //cubo 2
        Transform3D traslacion2 = new Transform3D();
        traslacion1.set(new Vector3f(0.4f,0,0));

        TransformGroup tg2 = new TransformGroup(traslacion2);

        ColorCube cubo2 = new ColorCube(0.2f);
        tg2.addChild(cubo2);

        mouseGroup.addChild(tg1);
        mouseGroup.addChild(tg2);

        MouseRotate mr = new MouseRotate();
        mr.setTransformGroup(mouseGroup);
        mr.setSchedulingBounds(new BoundingSphere (new Point3d(), 1000f));
        objetoRaiz.addChild(mr);

        return objetoRaiz;

}
    public static void main(String[] args) {
        System.setProperty("sun.awt.noerasebackground", "true");
        JFrame ventana = new JFrame("TransformGroup java 3D");
        App3D_6 panel = new App3D_6();
        ventana.add(panel);
        ventana.setSize(700, 700);
        ventana.setVisible(true);
        ventana.setLocationRelativeTo(null);
        ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    }
}
```



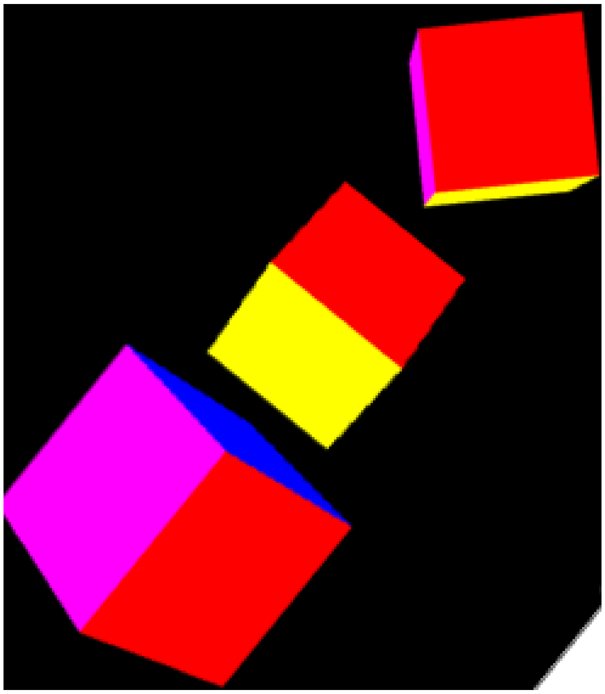
Rotación de un cubo por eje x,y o z.

```
import java.awt.BorderLayout;

import java.awt.GraphicsConfiguration;
import javax.media.j3d.BoundingSphere;
import javax.media.j3d.BranchGroup;
import javax.media.j3d.Canvas3D;
import javax.media.j3d.Transform3D;
import javax.media.j3d.TransformGroup;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.vecmath.Point3d;
import javax.vecmath.Vector3f;

import com.sun.j3d.utils.behaviors.mouse.MouseRotate;
import com.sun.j3d.utils.geometry.ColorCube;
import com.sun.j3d.utils.universe.SimpleUniverse;

public class App3D_7 extends JPanel {
    public App3D_7() {
        GraphicsConfiguration config = SimpleUniverse.getPreferredConfiguration();
        Canvas3D canvas3D = new Canvas3D(config);

        setLayout(new BorderLayout());
        add(canvas3D);
        SimpleUniverse universo = new SimpleUniverse(canvas3D);
        universo.getViewingPlatform().setNominalViewingTransform();

        BranchGroup escena = CrearGrafoEscena();
        escena.compile();

        universo.addBranchGraph(escena);
    }
```

```
public BranchGroup CrearGrafoEscena() {
        BranchGroup objetoRaiz = new BranchGroup();
        int grados =45;

        //rotar en el eje x
        Transform3D traslacion1 = new Transform3D();
        traslacion1.set(new Vector3f(-0.5f,0,0));
        Transform3D rotacion1 = new Transform3D();
        rotacion1.rotX(Math.toRadians(grados));

        traslacion1.mul(rotacion1);
        TransformGroup tg1 = new TransformGroup(traslacion1);

        ColorCube cubo1 = new ColorCube(0.2f);
        tg1.addChild(cubo1);
        // rotacion en el eje Y

        Transform3D traslacion2 = new Transform3D();
        traslacion2.set(new Vector3f(0,0,0));
        Transform3D rotacion2 = new Transform3D();
        rotacion2.rotY(Math.toRadians(grados));

        traslacion2.mul(rotacion2);
        TransformGroup tg2 = new TransformGroup(traslacion2);

        ColorCube cubo2 = new ColorCube(0.15f);
        tg2.addChild(cubo2);
        //rotacion en el eje Z
```

```
    Transform3D traslacion3 = new Transform3D();
    traslacion3.set(new Vector3f(0.5f,0,0));
    Transform3D rotacion3 = new Transform3D();
    rotacion3.rotZ(Math.toRadians(grados));

    traslacion3.mul(rotacion3);
    TransformGroup tg3 = new TransformGroup(traslacion3);

    ColorCube cubo3 = new ColorCube(0.15f);
    tg3.addChild(cubo3);

    objetoRaiz.addChild(tg1);
    objetoRaiz.addChild(tg2);
    objetoRaiz.addChild(tg3);


    return objetoRaiz;

}
public static void main(String[] args) {
    System.setProperty("sun.awt.noerasebackground", "true");
    JFrame ventana = new JFrame("TransformGroup java 3D");
    App3D_7 panel = new App3D_7();
    ventana.add(panel);
    ventana.setSize(700, 700);
    ventana.setVisible(true);
    ventana.setLocationRelativeTo(null);
    ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
        }

    }
```

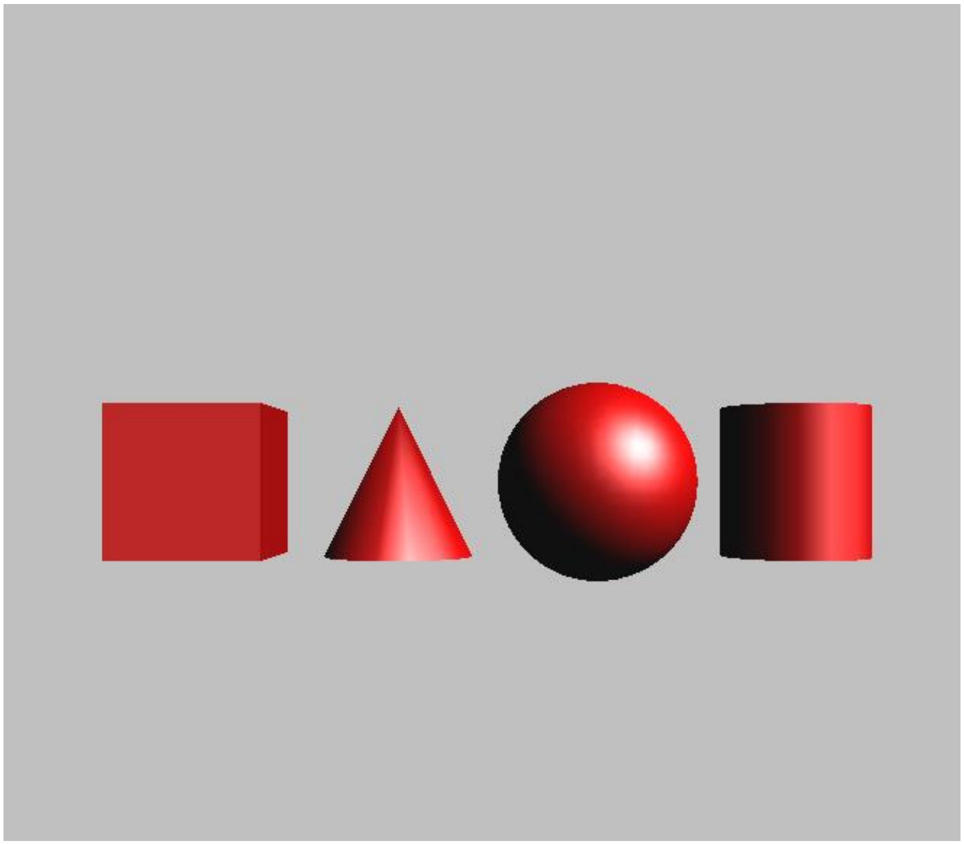


Figuras geométricas 3D sombreadas e iluminadas.

```
import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.GraphicsConfiguration;

import javax.media.j3d.AmbientLight;
import javax.media.j3d.Appearance;
import javax.media.j3d.Background;
import javax.media.j3d.BoundingSphere;
import javax.media.j3d.BranchGroup;
import javax.media.j3d.Canvas3D;
import javax.media.j3d.DirectionalLight;
import javax.media.j3d.Material;
import javax.media.j3d.Transform3D;
import javax.media.j3d.TransformGroup;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.vecmath.Color3f;
import javax.vecmath.Point3d;
import javax.vecmath.Vector3f;

import com.sun.j3d.utils.behaviors.mouse.MouseRotate;
import com.sun.j3d.utils.geometry.Box;
import com.sun.j3d.utils.geometry.ColorCube;
import com.sun.j3d.utils.geometry.Cone;
import com.sun.j3d.utils.geometry.Cylinder;
import com.sun.j3d.utils.geometry.Sphere;
import com.sun.j3d.utils.universe.SimpleUniverse;
```

```
public class App3D_8 extends JPanel {
    public App3D_8() {
        GraphicsConfiguration config = SimpleUniverse.getPreferredConfiguration();
        Canvas3D canvas3D = new Canvas3D(config);

        setLayout(new BorderLayout());
        add(canvas3D);
        SimpleUniverse universo = new SimpleUniverse(canvas3D);
        universo.getViewingPlatform().setNominalViewingTransform();

        BranchGroup escena = CrearGrafoEscena();
        escena.compile();

        universo.addBranchGraph(escena);
    }
    public BranchGroup CrearGrafoEscena() {
        BranchGroup objetoRaiz = new BranchGroup();

         Background fondo = new Background (new Color3f(Color.LIGHT_GRAY));
         fondo.setApplicationBounds(new BoundingSphere());
         objetoRaiz.addChild(fondo);

         Appearance apariencia = new Appearance();
         Material material = new Material();
         material.setAmbientColor(new Color3f(Color.DARK_GRAY));
         material.setDiffuseColor(new Color3f(Color.red));
         material.setSpecularColor(new Color3f(Color.white));
         material.setShininess(20.0f);
        apariencia.setMaterial(material);


        Transform3D traslacion1 = new Transform3D();
            traslacion1.set(new Vector3f(-0.6f,0,0));
            TransformGroup tg1 = new TransformGroup(traslacion1);
            Box cubo = new Box (0.15f, 0.15f, 0.15f, Box.GENERATE_NORMALS,
apariencia);
            tg1.addChild(cubo);


            Transform3D traslacion2 = new Transform3D();
            traslacion2.set(new Vector3f(-0.2f,0,0));
            TransformGroup tg2 = new TransformGroup(traslacion2);
            Cone cono = new Cone (0.15f, 0.3f,  Cone.GENERATE_NORMALS,
apariencia);
            tg2.addChild(cono);


            Transform3D traslacion3 = new Transform3D();
            traslacion3.set(new Vector3f(0.2f,0,0));
            TransformGroup tg3 = new TransformGroup(traslacion3);
            Sphere esfera = new Sphere (0.2f, Sphere.GENERATE_NORMALS, 50,
apariencia);
            tg3.addChild(esfera);


            Transform3D traslacion4 = new Transform3D();
            traslacion4.set(new Vector3f(0.6f,0,0));
            TransformGroup tg4 = new TransformGroup(traslacion4);
            Cylinder cilindro = new Cylinder (0.15f,0.3f,
Cylinder.GENERATE_NORMALS, apariencia);
            tg4.addChild(cilindro);
```

```
}


            Color3f colorAmbiente = new Color3f(Color.DARK_GRAY);
            AmbientLight luzAmbiente = new AmbientLight(colorAmbiente);
            luzAmbiente.setInfluencingBounds(new BoundingSphere(new Point3d(0,0,0),1000));


            Color3f colorLuz = new Color3f(Color.white);
            Vector3f dirLuz = new Vector3f(-1.0f,-1.0f,-1.0f );
            DirectionalLight luz  = new DirectionalLight(colorLuz, dirLuz);
            luz.setInfluencingBounds(new BoundingSphere(new Point3d(0,0,0),100));
            objetoRaiz.addChild(luzAmbiente);
            objetoRaiz.addChild(luz);
            objetoRaiz.addChild(tg1);
            objetoRaiz.addChild(tg2);
            objetoRaiz.addChild(tg3);
            objetoRaiz.addChild(tg4);


        return objetoRaiz;
}
    public static void main(String[] args) {
        System.setProperty("sun.awt.noerasebackground", "true");
        JFrame ventana = new JFrame("TransformGroup java 3D");
        App3D_8 panel = new App3D_8();
        ventana.add(panel);
        ventana.setSize(700, 700);
        ventana.setVisible(true);
        ventana.setLocationRelativeTo(null);
        ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}
```



Rotar automáticamentelas figuras del inciso 8.

```
import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.GraphicsConfiguration;
import javax.media.j3d.BranchGroup;
import javax.media.j3d.Canvas3D;
import javax.media.j3d.DirectionalLight;
import javax.media.j3d.Material;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.vecmath.Color3f;
import javax.vecmath.Point3d;
import javax.vecmath.Vector3d;
import javax.vecmath.Vector3f;

import com.sun.j3d.utils.behaviors.keyboard.KeyNavigator;
import com.sun.j3d.utils.behaviors.keyboard.KeyNavigatorBehavior;
import com.sun.j3d.utils.behaviors.mouse.MouseRotate;
import com.sun.j3d.utils.geometry.Box;
import com.sun.j3d.utils.geometry.ColorCube;
import com.sun.j3d.utils.geometry.Cone;
import com.sun.j3d.utils.geometry.Cylinder;
import com.sun.j3d.utils.geometry.Sphere;
import com.sun.j3d.utils.universe.SimpleUniverse;
import javax.media.j3d.Alpha;
import javax.media.j3d.AmbientLight;
import javax.media.j3d.Appearance;
import javax.media.j3d.Background;
import javax.media.j3d.BoundingSphere;
import javax.media.j3d.RotationInterpolator;
import javax.media.j3d.Transform3D;
import javax.media.j3d.TransformGroup;

public class App3D_9 extends JPanel {
        SimpleUniverse universo;

        public App3D_9() {
                GraphicsConfiguration config = SimpleUniverse.getPreferredConfiguration();
                Canvas3D canvas3D = new Canvas3D(config);
                universo = new SimpleUniverse(canvas3D);



                setLayout(new BorderLayout());
                add(canvas3D);

                universo.getViewingPlatform().setNominalViewingTransform();

                BranchGroup escena = CrearGrafoEscena();
                escena.compile();
                universo.addBranchGraph(escena);
        }

        public BranchGroup CrearGrafoEscena() {
                BranchGroup objetoRaiz = new BranchGroup();

                Background fondo = new Background(new Color3f(Color.WHITE));
                fondo.setApplicationBounds(new BoundingSphere());
```

```
objetoRaiz.addChild(fondo);

Appearance apariencia = new Appearance();
Material material  = new Material();
material.setAmbientColor(new Color3f(Color.DARK_GRAY));
material.setDiffuseColor(new Color3f(Color.RED));
material.setSpecularColor(new Color3f(Color.WHITE));
material.setShininess(30.0f);
apariencia.setMaterial(material);

//Cubo

Transform3D traslacion1 = new Transform3D();
traslacion1.set(new Vector3f(-0.6f,0,0));
TransformGroup tg1 = new TransformGroup(traslacion1);
Box cubo = new Box(0.15f,0.15f,0.15f, Box.GENERATE_NORMALS, apariencia);
tg1.addChild(cubo);

//Cono
Transform3D traslacion2 = new Transform3D();
traslacion2.set(new Vector3f(-0.2f,0,0));
TransformGroup tg2 = new TransformGroup(traslacion2);
Cone cono = new Cone(0.15f, 0.3f, Cone.GENERATE_NORMALS, apariencia);
tg2.addChild(cono);


//Esfera
Transform3D traslacion3 = new Transform3D();
traslacion3.set(new Vector3f(0.2f,0,0));
```

```
            TransformGroup tg3 = new TransformGroup(traslacion3);
            Sphere esfera = new Sphere(0.2f, Sphere.GENERATE_NORMALS,50,apariencia);
            tg3.addChild(esfera);

            Transform3D traslacion4 = new Transform3D();
            traslacion4.set(new Vector3f(0.6f,0,0));
            TransformGroup tg4 = new TransformGroup(traslacion4);
            Cylinder cilindro = new Cylinder(0.18f, 0.3f,
Cylinder.GENERATE_NORMALS,apariencia);
            tg4.addChild(cilindro);

            Color3f colorAmbiente = new Color3f(Color.DARK_GRAY);
            AmbientLight luzAmbiente = new AmbientLight(colorAmbiente);
            luzAmbiente.setInfluencingBounds(new BoundingSphere(new
Point3d(0,0,0),1000));
            Color3f colorLuz = new Color3f(Color.WHITE);
            Vector3f dirluz = new Vector3f(-1.0f, -1.0f, -1.0f);
            DirectionalLight luz = new DirectionalLight(colorLuz,dirluz);
            luz.setInfluencingBounds(new BoundingSphere(new Point3d(0,0,0),100));

            TransformGroup objetoGiro = new TransformGroup();
            objetoGiro.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
            objetoRaiz.addChild(objetoGiro);
            objetoGiro.addChild(tg1);
            objetoGiro.addChild(tg2);
            objetoGiro.addChild(tg3);
            objetoGiro.addChild(tg4);
            objetoGiro.addChild(luzAmbiente);
            objetoGiro.addChild(luz);
            Alpha rotacionAlpha = new Alpha(-1, 4000);
```

```
        RotationInterpolator rotacion = new RotationInterpolator(rotacionAlpha, objetoGiro);

        rotacion.setSchedulingBounds(new BoundingSphere());

        objetoRaiz.addChild(rotacion);


        return objetoRaiz;

    }

    public static void main(String[] args) {

        System.setProperty("sun.awt.noerasebackground", "true");

        JFrame ventana = new JFrame("TransformGroup java 3D");

        App3D_9 panel = new App3D_9();

        ventana.add(panel);

        ventana.setSize(700, 700);

        ventana.setVisible(true);

        ventana.setLocationRelativeTo(null);

        ventana.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);


    }

}
```